Cisco Content Switching Module (CSM) to Application Control Engine (ACE) Migration

December 4, 2008 this document was created upon test plan completion.

CSM to ACE migration testing was performed by the Cisco Safe Harbor testing team on behalf of the Application Delivery Business Unit (ADBU).

Executive Summary

Cisco Safe Harbor is an initiative that provides the global enterprise customer with a stable Cisco software version of choice. Safe Harbor focuses on satisfying customer quality requirements in key vertical markets. This program links and expands upon several Cisco testing projects, including development, regression testing, and systems testing critical global enterprise success. Safe Harbor is the successful completion of extensive validated testing for each release targeting enterprise markets.

Similar Cisco initiatives focusing on other platforms testing diverse vertical markets leverages Safe Harbor testing to provide a holistic approach to the general improvement of Cisco software.

This document describes the testing environment, test plans, and results.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. Safe Harbor is designed to cover critical path areas and augment ongoing regression and systems testing.

Note This documentation stipulates that Safe Harbor tests either pass, pass with exception, or fail. The underlying assumption for publishing a Safe Harbor release is that testing at large, passed, because all the individual tests passed, and that any exception to, or failure of, any test has been properly resolved, closed, or scrutinized by the Safe Harbor engineering team as a non show stopping defect, and noted. If a test fails, and the impact to our financial customer base is determined to be broad enough, the entire release fails (resulting from 1 or more unresolved DDTS's, notwithstanding unresolved cosmetic, minor, or test specific DDTS's, which are scrutinized by the Safe Harbor engineering team as being a non show stopping defect), so we therefore, do not certify, or publish, the release. If a test fails, and the impact to our financial customer base is determined to be minor, the release may still be certified, with DDTS's noted. Exceptions to any particular test are noted for disclosure purposes and incidental noteworthy clarification. Customers are advised to carefully review selected tests, by test suite and feature, peculiar to their environment.

Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

This document contains the following introductory sections:

This document contains the following test sections:

About the ADBU

The Cisco Application Delivery Business Unit (ADBU) develops next generation application optimization solutions for the Data Center and Branch Office / Remote User. These products provide customers with a unique end-to-end solution - enabling reliable, accelerated and secure application delivery to any user, using any device, anywhere in the world. With a unified architecture, ADBU delivers value by improving the way today's extended enterprises serve their customers, employees and partners.

1. Data Center Application Services (DCAS) products provide network and application operations management in Enterprise and Service Provider organizations with solutions to optimize and accelerate the delivery of critical application traffic while lowering overall server load and reducing network bandwidth requirements.
2. Wide Area Application Services (WAAS) enable seamless access over the WAN to centrally hosted applications, storage and rich media. These products allow enterprises to consolidate their distributed servers and storage into centrally managed data centers, while offering LAN-like access to their remote users.
3. Cisco Application and Content Networking System (ACNS) is a powerful solution for creating a video Content Delivery Network (CDN), delivering higher performance and more efficient Video on Demand (VoD) and live video streaming over the WAN.

   Cisco ACNS provides a scalable and reliable video streaming infrastructure to support the following applications:

   - Corporate communications
   - e-Learning (distance learning)
   - Digital signage

Video has proven itself as the best way for executives to communicate corporate strategies and achievements to employees, for departments to deliver sales and product training, and for organizations to ensure and track training that employees must receive for regulatory compliance.

About Cisco Safe Harbor

The goal of Cisco Safe Harbor is to provide improved network stability, reliability, and performance with respect to Cisco software. Safe Harbor involves testing the feature sets and protocols in a particular Cisco Release image on the Catalyst 6500 platform to provide high quality code for the enterprise sector. This combination of features, hardware, and image is tested in a laboratory environment that simulates the enterprise business network environment using regularly updated topologies and configurations provided by the enterprise customers. For information on the hardware tested and the network setup of the test environment, see the "Enterprise Lab Topology" section on page 3.

Functionality tests are conducted to verify feature functionality. Regression tests are conducted to validate existing features and verify that functionality is maintained. Negative tests are designed and conducted to stress the features and their interoperability. For information on each feature and its testing, see the "Test Cases" section on page 11.

During the testing, the network is placed under loads that are consistent with those in an network. A standard suite of tools (for example, Netcom Smartbits, IXIA packet generator, or Cisco Pagent) is used to generate network traffic. Network testing includes a combination of automated and manual tests. Simple Network Management Protocol (SNMP) is used to poll the network during the tests, and all tests are analyzed. For a summary of the test results, see the "Test Results Summary" section on page 5.

Note Safe Harbor testing does not address issues that might exist in the customer change control and operations processes.

Enterprise Lab Topology

This testing takes place on devices that are part of a larger enterprise topology. The topology is illustrated in Figure 1 and defined in the following sections.

The base lab topology is defined in the following sections:

- Testbed Diagrams, page 4
- Topology Configurations, page 208

The enterprise network environment configured in the lab includes the following hardware:

In this network upstream and downstream Layer 3 switches are running Native IOS 12.2(18)SXF13 while the ACE blades were running ACE Release A2(1.2) and the CSM blades were running CSM Release 4.2(6), configured as follows:

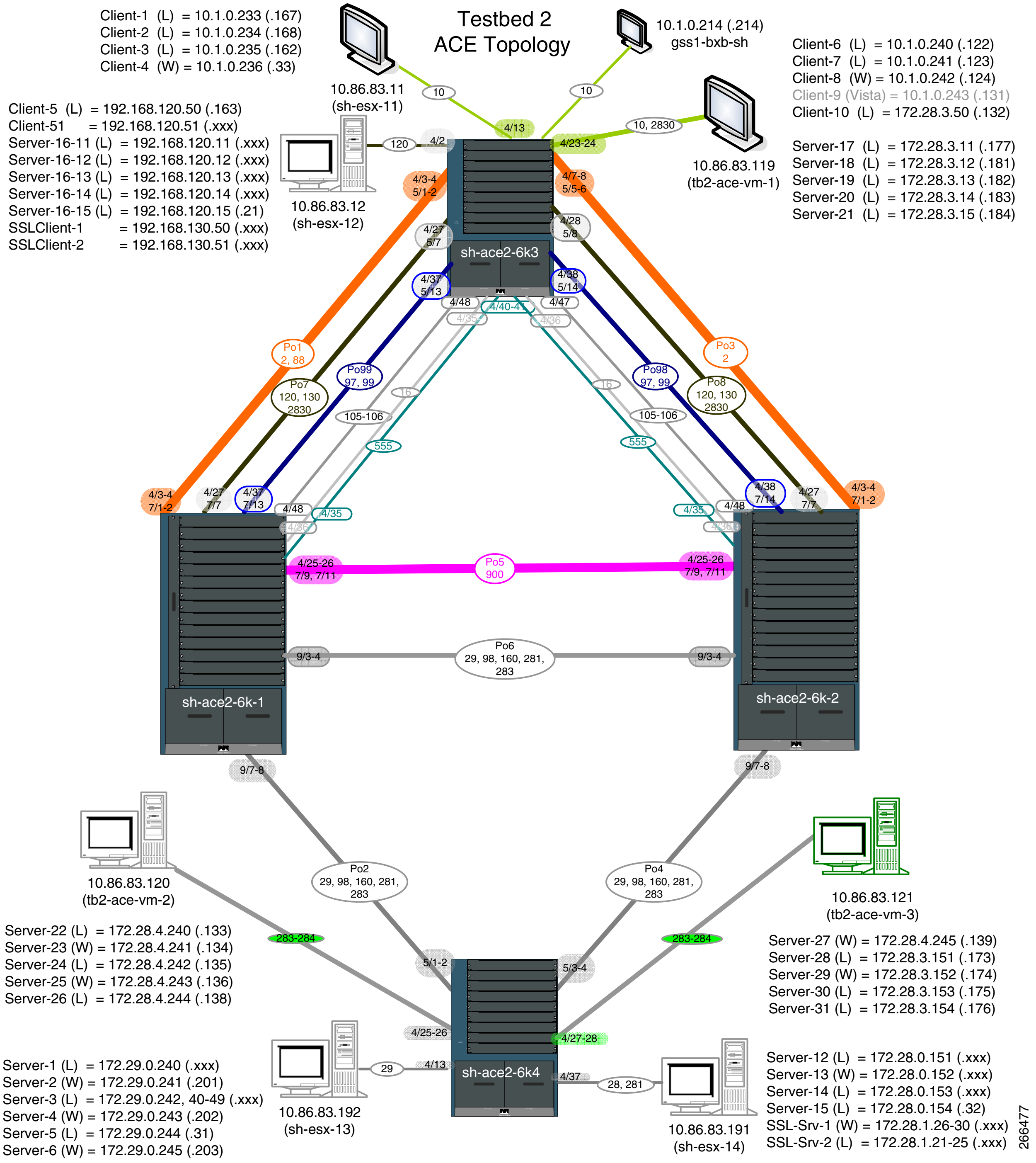
- Upstream and downstream switches (sh-ace-6k-3 and sh-ace-6k-4), are running Cisco IOS Native 12.2(18)SXF13
- CAT's with the ACE's and CSM's installed (sh-ace-6k-1 and sh-ace-6k-2), are running Native 12.2(18)SXF13 / ACE A2(1.2) / CSM 4.2(6).
- IXIA test devices to generate simulated customer traffic.

The hardware configuration in the test lab includes a combination of fabric-capable, and nonfabric modules.

Testbed Diagrams

- Figure 1 shows the Cisco CSM to ACE Migration topology configuration.

Figure 1 Cisco CSM to ACE Migration Topology


Testbed 2
ACE Topology
Client-1 (L) = 10.1.0.233 (.167)
Client-2 (L) = 10.1.0.234 (.168)
Client-3 (L) = 10.1.0.235 (.162)
Client-4 (W) = 10.1.0.236 (.33)
10.86.83.11
(sh-esx-11)
10.1.0.214 (.214)
gss1-bxb-sh
Client-6 (L) = 10.1.0.240 (.122)
Client-7 (L) = 10.1.0.241 (.123)
Client-8 (W) = 10.1.0.242 (.124)
Client-9 (Vista) = 10.1.0.243 (.131)
Client-10 (L) = 172.28.3.50 (.132)
Client-5 (L) = 192.168.120.50 (.163)
Client-51 = 192.168.120.51 (.xxx)
Server-16-11 (L) = 192.168.120.11 (.xxx)
Server-16-12 (L) = 192.168.120.12 (.xxx)
Server-16-13 (L) = 192.168.120.13 (.xxx)
Server-16-14 (L) = 192.168.120.14 (.xxx)
Server-16-15 (L) = 192.168.120.15 (.21)
SSLClient-1 = 192.168.130.50 (.xxx)
SSLClient-2 = 192.168.130.51 (.xxx)
10.86.83.12
(sh-esx-12)
10
10
10, 2830
120
4/2
4/13
4/23-24
10.86.83.119
(tb2-ace-vm-1)
Server-17 (L) = 172.28.3.11 (.177)
Server-18 (L) = 172.28.3.12 (.181)
Server-19 (L) = 172.28.3.13 (.182)
Server-20 (L) = 172.28.3.14 (.183)
Server-21 (L) = 172.28.3.15 (.184)
4/3-4
5/1-2
4/7-8
5/5-6
4/27
5/7
4/28
5/8
sh-ace2-6k3
4/37
5/13
4/38
5/14
4/48
4/40-41
4/47
4/35
4/36
Po1
2, 88
Po99
97, 99
Po7
120, 130
2830
16
105-106
555
Po3
2
Po98
97, 99
Po8
120, 130
2830
16
105-106
555
4/3-4
7/1-2
4/27
7/7
4/37
7/13
4/48
4/35
4/36
4/38
7/14
4/27
7/7
4/3-4
7/1-2
4/35
4/48
4/36
4/25-26
7/9, 7/11
Po5
900
4/25-26
7/9, 7/11
9/3-4
Po6
29, 98, 160, 281,
283
9/3-4
sh-ace2-6k-1
sh-ace2-6k-2
9/7-8
9/7-8
10.86.83.120
(tb2-ace-vm-2)
Po2
29, 98, 160, 281,
283
Po4
29, 98, 160, 281,
283
10.86.83.121
(tb2-ace-vm-3)
Server-22 (L) = 172.28.4.240 (.133)
Server-23 (W) = 172.28.4.241 (.134)
Server-24 (L) = 172.28.4.242 (.135)
Server-25 (W) = 172.28.4.243 (.136)
Server-26 (L) = 172.28.4.244 (.138)
283-284
283-284
Server-27 (W) = 172.28.4.245 (.139)
Server-28 (L) = 172.28.3.151 (.173)
Server-29 (W) = 172.28.3.152 (.174)
Server-30 (L) = 172.28.3.153 (.175)
Server-31 (L) = 172.28.3.154 (.176)
5/1-2
5/3-4
4/25-26
4/27-28
Server-1 (L) = 172.29.0.240 (.xxx)
Server-2 (W) = 172.29.0.241 (.201)
Server-3 (L) = 172.29.0.242, 40-49 (.xxx)
Server-4 (W) = 172.29.0.243 (.202)
Server-5 (L) = 172.29.0.244 (.31)
Server-6 (W) = 172.29.0.245 (.203)
29
4/13
sh-ace2-6k4
4/37
28, 281
10.86.83.192
(sh-esx-13)
10.86.83.191
(sh-esx-14)
Server-12 (L) = 172.28.0.151 (.xxx)
Server-13 (W) = 172.28.0.152 (.xxx)
Server-14 (L) = 172.28.0.153 (.xxx)
Server-15 (L) = 172.28.0.154 (.32)
SSL-Srv-1 (W) = 172.28.1.26-30 (.xxx)
SSL-Srv-2 (L) = 172.28.1.21-25 (.xxx)
266477

Test Results Summary

Table 1-1 summarizes results of all completed testing as part of the Cisco Safe Harbor initiative for this release. Table 1-1 includes the feature or function tested, the section or suite that describes the feature set to which the feature or function belongs, the component tests for each feature or function, and any related defects (DDTS Summary) found during Safe Harbor testing.

Note Test results are unique to technologies covered and actual scenarios in which they were tested. Safe Harbor is designed to cover critical path areas and augment ongoing regression and systems testing.

Note This documentation stipulates that Safe Harbor tests either pass, pass with exception, or fail. The underlying assumption for publishing a Safe Harbor release is that testing at large, passed, because all individual tests passed, and that any exception to, or failure of, any test, has been properly resolved, closed, or scrutinized by the Safe Harbor engineering team as a non show stopping defect, and noted. If a test fails, and the impact to our enterprise customer base is determined to be broad enough, the entire release fails (resulting from 1 or more unresolved DDTS's, notwithstanding unresolved cosmetic, minor, or test specific DDTS's, which are scrutinized by the Safe Harbor engineering team as being a non show stopping defect), so we therefore, do not certify, or publish, the release. If a test fails, and the impact to our enterprise customer base is determined to be minor, the release may still be certified, with DDTS's noted. Exceptions to any particular test are noted for disclosure purposes and incidental noteworthy clarification. Customers are advised to carefully review selected tests, by test suite and feature, peculiar to their environment.

Table 1-1 Safe Harbor Test Results Summary

Test Suites	Feature/Function	Tests	Results
Basic Functionality, page 11	Device Management, page 11	1. SNMP—ACE 2. SNMP—CSM 3. TACACS—ACE 4. TACACS—CSM	CSCsj80265
Basic Functionality, page 11	SPAN, page 19	1. Distributed Etherchannel SPAN—ACE 2. Distributed Etherchannel SPAN—CSM	CSCsm95456
Basic Functionality, page 11	Traffic Decisions, page 23	1. Failaction Purge—ACE 2. Failaction Purge—CSM 3. Idle Timeout—ACE 4. Idle Timeout—CSM 5. Route Health Injection—ACE 6. Route Health Injection—CSM	CSCsq63242
Health and Redundancy, page 43	Backup Serverfarm, page 43	1. Backup Serverfarm—ACE 2. Backup Serverfarm—CSM	

Table 1-1 Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Health and Redundancy, page 43	Configuration Syncronization, page 46	1. Config Sync Large—ACE 2. Config Sync Large—CSM 3. Configuration Sync—ACE 4. Configuration Sync—CSM	CSCsj68643 CSCsu79370 CSCso77725 CSCsq63242
Health and Redundancy, page 43	Probes, page 53	1. DNS probe—ACE 2. DNS probe—CSM 3. HTTP Probes—ACE 4. HTTP Probes—CSM 5. KALAP by Tag—ACE 6. KALAP by Tag—CSM 7. KALAP by VIP—ACE 8. KALAL by VIP—CSM 9. SSL Probe—ACE 10. SSL Probes—CSM	CSCsj26410 CSCsu54970 CSCsu55144
Health and Redundancy, page 43	Redundancy, page 91	1. Multiple Chassis Redundancy—ACE 2. Multiple Chassis Redundancy—CSM	CSCek51826
Health and Redundancy, page 43	Tracking, page 99	1. Fault Tolerant Tracking—ACE 2. Fault Tolerant Tracking—CSM	CSCek51743
Load Balancing, page 104	Predictors, page 104	1. Least Connection Predictor—ACE 2. Least Connection Predictor—CSM 3. Maxconn Connection Limiter—ACE 4. Maxconn Connection Limiter—CSM	CSCse15530 CSCsm12883 CSCsu65844
Traffic Handling, page 116	FTP, page 116	1. Passive FTP—ACE 2. Passive FTP—CSM	
Traffic Handling, page 116	Insert, page 119	1. Cookie Insert—ACE 2. Cookie Insert—CSM 3. Header Insert—ACE 4. Header Insert—CSM	
Traffic Handling, page 116	Maps, page 131	1. Cookie Map—ACE 2. Cookie Map—CSM 3. Header Map—ACE 4. Header Map—CSM 5. URL Map—ACE 6. URL Map—CSM	

Table 1-1 Safe Harbor Test Results Summary (continued)

Test Suites	Feature/Function	Tests	Results
Traffic Handling, page 116	Miscellaneous, page 142	1. Persistence Rebalance—ACE 2. Persistence Rebalance—CSM 3. Redirect Policy—ACE 4. Redirect Policy—CSM 5. Topology Baseline—ACE 6. Topology Baseline—CSM 7. UDP Load Balancing—ACE 8. UDP Load Balancing—CSM	CSCsu95887 CSCsg80625 CSCsu54652
Traffic Handling, page 116	SSL, page 157	1. Client Authentication—ACE 2. Client Authentication—CSM-SSLM 3. End to End SSL—ACE 4. End to End SSL—CSM-SSLM 5. Header Insert (SSL)—ACE 6. Header Insert (SSL)—CSM-SSLM 7. SSL Termination—ACE 8. SSL Termination—CSM-SSLM 9. SSL URL Rewrite—ACE 10. SSL URL Rewrite—CSM-SSLM	CSCsv01732 CSCsv02360 CSCsr41176
Traffic Handling, page 116	Sticky, page 189	1. Cookie Sticky—ACE 2. Cookie Sticky—CSM 3. Header Sticky—ACE 4. Header Sticky—CSM 5. IP Netmask Sticky—ACE 6. IP Netmask Sticky—CSM 7. SSL Session ID Sticky—ACE 8. SSL Session ID Sticky—CSM	

DDTS Summary

Table 2 lists Development Defect Tracking System (DDTS) software bugs with descriptions, filed by the Safe Harbor testing team during this test cycle, sorted by severity. Table 3 lists DDTS with descriptions encountered during this test cycle, sorted by severity. Table 4 lists DDTS of interest and not encountered during this test cycle, sorted by severity.

Table 2 DDTS Filed During This Cisco Safe Harbor Test Cycle

DDTS	Severity	Description
CSCsu95887	2	Core Dump on Primary ACE
CSCsv02360	2	low MTU on client and conn reuse breaks ace ssl termination
CSCsr41176	3	server hello responds with cipher not included in client hello
CSCsu54652	3	ACE Strips Checksum When 'inspect dns' Enabled
CSCsu54970	3	kalap reported load is 255 though actual load should be 2
CSCsu79370	3	CSM Hangs During Large Config Sync
CSCsu88760	3	scripted tftp probe does not fail when server is down
CSCsv01732	3	itasca ssl core
CSCsu65844	5	Syslog Reports Rserver OUTOFSERVICE When Its State Is MAXCONNS
CSCsu55144	6	serverfarm is down due to maxconn but vip is still inservice

Table 3 DDTS Encountered During This Cisco Safe Harbor Test Cycle

DDTS	Severity	Description
CSCsm95456	2	Duplicate L3 packets with 6708 and DEC
CSCek51743	3	csm ft detail shows tracked hsrp group as standby when it is active
CSCek51826	3	conn replication fails when reset master preempts
CSCsg80625	3	udp packets are dropped by ace
CSCsj26410	3	CSM returns load=128 for KAL-AP by VIP when vserver is OOS
CSCsj68643	3	can_wait_specific_msg: Aborting call (SAP 27, pid 959)
CSCsm12883	3	ACE Max Conn resetting connections
CSCsq63242	3	IP address change in interface produce unnecessary warning message
CSCse15530	4	hit count and total connections counters are incorrectly updated on s/o
CSCsj80265	4	SSHv1 with TACACS sessions into ace fail
CSCso77725	4	%ACE-3-440003: Deletion failed for Vserver Stats Table

Table 4 DDTS of Interest and Not Encountered During This Cisco Safe Harbor Test Cycle

DDTS	Severity	Description
CSCso74865	1	atlering from hash header predictor to hash cookie causes system crashed
CSCsk36611	2	ACE: ssl re-handshake fails with IE when cert chain bigger than 4k byte

Table 4 DDTS of Interest and Not Encountered During This Cisco Safe Harbor Test Cycle

DDTS	Severity	Description
CSCsl33851	2	action-list: stopped working completely for large number of header inser
CSCsl46334	2	Proxy Connection Leak with L4/L7 LB traffic in multiple contexts
CSCsm10702	2	ACE module hangs--unresponsive on network but reachable over EOBC
CSCsm43541	2	SSL - buffer corruption issue on the nitrox chip causing the crash ACE
CSCso91403	2	Modification to a large configuration can cause connection to be reset
CSCsq04822	2	conn-limit cause crash when max hit
CSCsj41909	3	ACE: NP 1 Failed : NP Process: QNX process io_net Crashed
CSCsj68643	3	can_wait_specific_msg: Aborting call (SAP 27, pid 959)
CSCsj74250	3	Key not encrypted when Configuring TACACS on ACE
CSCsk63774	3	"show serverfarm " display different stats from "show serverfarm"
CSCsl64911	3	ip address route in HTTPS probe is not working as expected in BAIKA
CSCsl68531	3	transparent rserver stops taking connections
CSCsl75662	3	ftp probe stuck in INIT after changing interval and delete/add to real
CSCsl80651	3	Scripted probes in INIT state
CSCsm37228	3	Standby ACE may reload after executing TACACS related command on master
CSCsm40004	3	ACE does not allow a space in the 'State' parameter of a CSR
CSCsm52480	3	ACE: Cannot Bridge Multicast IPv4 or IPv6 Traffic
CSCsm62263	3	ACE syslogd may core leading to ACE reload.
CSCsm64646	3	Probe stop firing and got stuck if large no of rserv became unreachable
CSCsm65534	3	ACE display of large byte counts in "show service-policy" is inaccurate
CSCsm67002	3	ME crash, ICM stuck - all ICM threads but one waiting for "send" signal
CSCsm71444	3	"no inservice standby" should not place the rserver in OPERATIONAL state
CSCsm72725	3	capture displayed on wrong context
CSCsm79292	3	vsh_conf_cmd scripted probe fails on standby
CSCsm90293	3	ACE: uses the underscore in the HELO domain of SMTP probe
CSCsm93110	3	SSL init fails with IIS 5.0 when Accpet/Required Client cert is enabled
CSCso00234	3	ACE can't handle non-minimally padded block ciphers
CSCso00356	3	L2-mode: syn cookie handled conn stalls if arp entry is not yet learned
CSCso02922	3	Half a context directory structure created if we run out of disk space
CSCso12722	3	A POST request might not fully pass to the Real
CSCso18391	3	Checkpoint Rollback can not do no nat dynamic
CSCso20415	3	ACE XML agent not translating the special char '&' for sh crypto cert
CSCso22472	3	ACE:wrong LB decision occurs if no command in class-map
CSCso25654	3	ACE: UDP probe not failing when interval set to 2 seconds
CSCso66799	3	ACE: All virtuals w/same IP must have ICMP Reply active configured

Table 4 DDTS of Interest and Not Encountered During This Cisco Safe Harbor Test Cycle

DDTS	Severity	Description
CSCso73385	3	ACE: EPSV command fails through ACE with inspect ftp
CSCso99599	3	Query on ring summary needs to take care of various ring types
CSCsq08329	3	Session cache not clearing after timeout value with cache full
CSCsq09414	3	not able to run xml-show command if aaa configured on the box
CSCsu79342	3	VLAN not up on the supervisor with autostate enabled
CSCsj80265	4	SSHv1 with TACACS sessions into ace fail
CSCsj64833	6	UDP/TCP traceroute does not work to configured ACE IP Interfaces
CSCsj94366	6	unable to change console settings

Test Cases

Functionality critical to global enterprises tested for this Cisco Safe Harbor release is described in the following sections. See Appendix: Topology Configurations for test device configurations.

- Basic Functionality, page 11
- Health and Redundancy, page 43
- Load Balancing, page 104
- Traffic Handling, page 116

Basic Functionality

Basic functionality covers the basic configurations needed to set up the load balancer.

This section contains the following topics:

- Device Management, page 11
- SPAN, page 19
- Traffic Decisions, page 23

Device Management

Testing related to administrative tasks on load balancer which range from device monitoring to access restrictions to remote configuration.

This section contains the following topics:

- SNMP—ACE, page 11
- SNMP—CSM, page 14
- TACACS—ACE, page 16
- TACACS—CSM, page 18

SNMP—ACE

SNMP is an application layer protocol that facilitates the exchange of management information between an NMS, SNMP agents, and managed devices such as the load balancer. load balancer may be configured to send traps (event notifications) to an NMS, or the NMS may be used to browse the Management Information Bases (MIB's) residing on the load balancer.

This test verified the current status of a particular rserver and vserver through SNMP and the CLI. This test also verified that SNMP traps were sent when a server or Virtual IP address (VIP) was not operational.

Relevant Load Balancer Configuration

```
serverfarm host MAX-CONN
  probe HTTP
  rserver RT-151
    conn-limit max 3 min 2
    inservice
```

```
serverfarm host MAX-CONN2
  probe HTTP
  rserver LOCAL-243
    conn-limit max 500 min 2
    inservice
  rserver RT-151
    conn-limit max 500 min 2
    inservice
  rserver RT-151 90
    conn-limit max 500 min 2
    inservice
  rserver RT-151 91
    conn-limit max 500 min 2
    inservice
  rserver RT-151 92
    conn-limit max 500 min 2
    inservice
  rserver RT-151 93
    conn-limit max 500 min 2
    inservice
  rserver RT-151 94
    conn-limit max 500 min 2
    inservice
  rserver RT-151 95
    conn-limit max 500 min 2
    inservice
  rserver RT-154
    inservice
class-map type http loadbalance match-all INDEX.HTML
  2 match http url /index.html
class-map match-all MAX-CONN-VIP_105
  2 match virtual-address 192.168.120.105 any
class-map type management match-any MGT
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
class-map type http loadbalance match-all URL*_L7
  2 match http url .*
policy-map type management first-match P-MGT
  class MGT
    permit
policy-map type loadbalance first-match MAX-CONN-LB-SF_MAX-CONN2
  class INDEX.HTML
    serverfarm MAX-CONN
  class URL*_L7
    serverfarm MAX-CONN2
policy-map multi-match SH-Gold-VIPs
  class MAX-CONN-VIP_105
    loadbalance vip inservice
    loadbalance policy MAX-CONN-LB-SF_MAX-CONN2
    loadbalance vip icmp-reply
    appl-parameter http advanced-options PERSIST-REBALANCE
service-policy input P-MGT
snmp-server community ACE-public group Network-Monitor
snmp-server community ACE-private group Network-Monitor
snmp-server host 10.1.0.236 traps version 2c ACE-public
snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
```

```
snmp-server enable traps snmp linkdown
```

Test Procedure

The procedure used to perform the SNMP—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that the Primary load balancer is configured for SNMP.

Step 3 Check the state of a server using the CLI and SNMP. Verify that the results of both commands match.

Step 4 Take a server out of service in a serverfarm. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 5 Only one server in the serverfarm was taken out of service. Verify that the global server remained operational.

Step 6 Bring the server back in service. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 7 Verify SNMP traps were sent when the that server went down and came back into an operational state.

Step 8 Cause a server to fail by adding a probe to it. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 9 Allow the failed server to become operational by changing the URL on the probe. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 10 Verify SNMP traps were sent when the server went down and came back into an operational state.

Step 11 Remove the probe from the serverfarm and change the probe’s URL.

Step 12 Bring the VIP out of service. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 13 Bring the VIP in service. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 14 Verify that SNMP traps were sent when the VIP transitioned to down and came back into an operational state.

Step 15 Return the load balancer back to its default Safe Harbor configuration.

Expected Results

The following test results are anticipated:

- We expect a client to retrieve information from the load balancer by using an SNMP browser.
- We expect SNMP traps to be sent to a trap host upon a server or VIP failure.
- We expect that the load balancer will not crash or become unresponsive.

Results

SNMP—ACE passed.

SNMP—CSM

SNMP is an application layer protocol that facilitates the exchange of management information between an NMS, SNMP agents, and managed devices such as the load balancer. load balancer may be configured to send traps (event notifications) to an NMS, or the NMS may be used to browse the Management Information Bases (MIB’s) residing on the load balancer.

This test verified the current status of a particular rserver and vserver through SNMP and the CLI. This test also verified that SNMP traps were sent when a server or Virtual IP address (VIP) was not operational.

Relevant Load Balancer Configuration

```
!
 real RT-LINUX-151
  address 172.28.0.151
  inservice
!
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 serverfarm MAX-CONN
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 3
   minconns 2
   inservice
  probe HTTP
!
 serverfarm MAX-CONN2
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 500
   inservice
  real name LOCAL-LINUX-240
   maxconns 500
   inservice
  real name LOCAL-IIS-241
   maxconns 500
   inservice
  real name RT-IIS-152
   maxconns 500
   inservice
  real name RT-LINUX-151 90
   maxconns 500
   inservice
  real name RT-LINUX-151 91
   maxconns 500
   inservice
  real name RT-LINUX-151 92
   maxconns 500
   inservice
  real name RT-LINUX-151 93
   maxconns 500
   inservice
  real name RT-LINUX-151 94
   maxconns 500
   inservice
  real name RT-LINUX-151 95
   maxconns 500
   inservice
  probe HTTP
!
 policy P-MAX-CONN
  url-map M-MAX-CONN
  sticky-group 31
  serverfarm MAX-CONN
!
 vserver MAX-CONN
  virtual 192.168.120.205 tcp www
  serverfarm MAX-CONN2
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  slb-policy P-MAX-CONN
  inservice
!
snmp-server community public RW
snmp-server trap-source Vlan83
snmp-server enable traps slb real virtual csrp
snmp-server host 10.86.83.124 public  casa slb
!
SNMP OID for real RT-LINUX-151
8.8.77.65.88.45.67.79.78.78.172.28.0.151.0
!
SNMP OID for vserver MAX-CONN
8.8.77.65.88.45.67.79.78.78
!
```

Test Procedure

The procedure used to perform the SNMP—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that the load balancer is configured for SNMP.

Step 3 Check the state of a server using the CLI and SNMP. Verify that the results of both commands match.

Step 4 Take a server out of service in a serverfarm. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 5 Only one server in the serverfarm was taken out of service. Verify that the global server remains operational.

Step 6 Bring the server back in service. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 7 Verify SNMP traps were sent when the that server went down and came back into an operational state.

Step 8 Cause a server to fail by adding a probe to it. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 9 Allow the failed server to become operational by changing the URL on the probe. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 10 Verify SNMP traps were sent when the server went down and came back into an operational state.

Step 11 Remove the probe from the serverfarm and change the probe's URL.

Step 12 Bring the VIP out of service. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 13 Bring the VIP in service. Verify the state change using the CLI and SNMP. The results of both commands should match.

Step 14 Verify SNMP traps were sent when the server transitioned to down and back into an operational state.

Step 15 Return the CSM to its default Safe Harbor configuration.

Expected Results

The following test results are anticipated:

- We expect a client to retrieve information from the load balancer by using an SNMP browser.
- We expect SNMP traps to be sent to a trap host upon a server or VIP failure.
- We expect no CPU or memory problems.

Results

SNMP—CSM passed.

TACACS—ACE

This test verified how the load balancer handled authentication requests using TACACS on a Cisco Secure ACS server. This test included how the load balancer handled multiple ACS servers. The server was configured with multiple users, each with different management rights. This test also verified how the load balancer handled the situation where the ACS servers were not reachable.

Relevant ACE Configuration

```
**SH-Gold context
tacacs-server key 7 "vwjjzamggu"
tacacs-server host 172.29.0.235 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-gold
  server 172.29.0.235
tacacs-server host 172.29.0.236 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-gold
  server 172.29.0.236
tacacs-server host 172.29.0.237 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-gold
  server 172.29.0.237
aaa authentication login default group sh-gold local
aaa authentication login error-enable
Admin domain SH-Gold-Domain
**Admin context
tacacs-server key 7 "vwjjzamggu"
tacacs-server host 10.86.83.215 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-admin
  server 10.86.83.215
aaa authentication login default group sh-admin local
aaa accounting default group sh-admin
aaa authentication login error-enable
```

Test Procedure

The procedure used to perform the TACACS—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Issue the **show tacacs-server** command to verify that all the TACACS servers are configured in both contexts. Issue the following commands:

```
changeto Admin
show tacacs-server
changeto SH-Gold
show tacacs-server
```

Step 3 From a Linux client, force authentication requests by trying to login to the SH-Gold context. First using SSH version 1, SSH version 2, and then telnet. Verify through the client CLI that the user was able to successfully login for all sessions.

Step 4 Verify on the ACE that the user (goldadmin) was authenticated and has the correct role and domain (Admin/SH-Gold) for all sessions. Issue the **show users** command.

Step 5 Disable all the tacacs servers. Try to open a telnet session to the SH-Gold context. First using a tacacs user account and then with a local user account. Verify through the client CLI that the tacacs account login failed, but the local user was successful.

Step 6 In the Admin context, change the tacacs server key to force a key mismatch. On a client, telnet to the management IP. Verify that the login is unsuccessful. Issue the following commands:

```
changeto Admin
configure
tacacs-server host 10.86.83.215 key cisco
end
```

Step 7 Change the tacacs server key, so that it matches. On a client, telnet to the Admin management IP. Verify that the login is successful. Issue the following commands:

```
configure
tacacs-server host 10.86.83.215 key Safe Harbor
end
changeto SH-Gold
```

Step 8 Login to the SH-Gold context with a user that only has user privileges. Issue the **configure** command, and verify that the client was not able to enter into configuration mode.

Step 9 Verify the user golduser cannot change to any other contexts.

Step 10 Login with a user that only has admin access to the context SH-Gold. Issue the **configure** command, and verify that the client was able to enter into configuration mode.

Step 11 Verify the user goldadmin cannot change to any other contexts.

Step 12 Login to the Admin context with a user that has full admin access. Verify that the user aceadmin can enter config mode and change to other contexts.

Step 13 In the SH-Gold context, change the tacacs configuration to not allow fallback to local user database. Issue the following commands:

```
changeto SH-Gold
configure
no aaa authentication login default group sh-gold local
aaa authentication login default group sh-gold
end
```

Step 14 Disable all of the tacacs servers configured for the SH-Gold context. From a Linux client try to login as a local user. Verify that the login attempt was unsuccessful.

Step 15 Replace the tacacs command to allow fallback to local user database. From a Linux client try to login again as a local user. Verify that the login attempt was successful. Issue the following commands:

```
configure
no aaa authentication login default group sh-gold
aaa authentication login default group sh-gold local
end
```

Expected Results

The following test results are anticipated:

- We expect users access to be authenticated by Cisco Secure ACS. We expect users not to access contexts that they should not be able to access.
- We expect the Cisco Secure ACS server to be unavailable if access to the ACE continues to exist.
- We expect that no core dumps or crashes will occur on the Primary or Standby ACE modules.
- We expect no CPU or memory problems.

Results

TACACS—ACE passed with exception. The following exceptions were noted: CSCsj80265.

TACACS—CSM

This test verified how the load balancer handled authentication requests using TACACS on a Cisco Secure ACS server. This test included how the load balancer handled multiple ACS servers. The server was configured with multiple users, each with different management rights. This test also verified how the load balancer handled the situation where the ACS servers were not reachable.

```
CSM/IOS Configurations
aaa new-model
tacacs-server key Safe Harbor
tacacs-server host 172.29.0.235
tacacs-server host 172.29.0.236
tacacs-server host 172.29.0.237
aaa group server tacacs+ csm-tacacs
        server 172.29.0.235
        server 172.29.0.236
        server 172.29.0.237
aaa authentication login default group csm-tacacs local
aaa authorization config-commands
aaa authorization commands 1 default group csm-tacacs local
aaa authorization commands 15 default group csm-tacacs local
```

Test Procedure

The procedure used to perform the TACACS—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On sh-ace2-6k-1, issue the **show tacacs** command to verify that all the TACACS servers are configured.

Step 3 From a client, attempt an authentication request. Verify on the client that the user was able to log in. Issue the **show users** command to verify on sh-ace2-6k-1 that the user was authenticated.

Step 4 Disable all the tacacs servers and try to log into sh-ace2-6k-1 with a tacacs user account. Then try to log in with a local user account.

Step 5 On sh-ace2-6k-1, change the tacacs server key to force a tacacs key mismatch. Try to authenticate a user and verify the client was unable to log in. Then restore the tacacs server key to the proper key. Try to authenticate a user and verify the client is able to log in.

Step 6 Log into sh-ace2-6k-1 with a user that does not have privileged EXEC command access. Issue the **configure terminal** command and verify that the client was not able to enter into configuration mode due to an authorization failure. Issue the command **show run** and verify the client can run the show command.

Step 7 Log into sh-ace2-6k-1 with a user that has privileged EXEC command access. Issue the **configure terminal** command, and verify that the client was able to enter into configuration mode.

Expected Results

The following test results are anticipated:

- We expect users access to be authenticated by Cisco Secure ACS.
- We expect to ba able to access the CSM if the Cisco Secure ACS server are unavailable.
- We expect no CPU or memory problems.

Results

TACACS—CSM passed.

SPAN

Testing related to the use of a SPAN port on the Catalyst 6000. Packets are replicated from the load balancer to the SPAN port. Packet captures are used to verify the traffic is successfully replicated.

This section contains the following topics:

- Distributed Etherchannel SPAN—ACE, page 20
- Distributed Etherchannel SPAN—CSM, page 21

Distributed Etherchannel SPAN—ACE

A SPAN session allows you to mirror packets from a port, vlan or etherchannel to another port for packet capture and analysis.

This test verified the functionality of the SPAN feature against different protocols traversing distributed etherchannels, captured on the load balancer's internal interface. The packet captured is used in conjunction with the test tool output to verify that the traffic was successfully replicated.

Relevant ACE Configuration

```
serverfarm host L3
  rserver LOCAL-244
    inservice
class-map match-all L3_139
  2 match virtual-address 192.168.120.139 any
policy-map type loadbalance first-match PLBSF_L3
  class class-default
    serverfarm L3
policy-map multi-match SH-Gold-VIPs
  class L3_139
    loadbalance vip inservice
    loadbalance policy PLBSF_L3
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
interface vlan 29
  ip address 172.29.0.3 255.255.255.0
  alias 172.29.0.1 255.255.255.0
  peer ip address 172.29.0.2 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.71 192.168.120.71 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  no shutdown
interface vlan 99
  ip address 192.168.99.3 255.255.255.0
  peer ip address 192.168.99.2 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 105
  ip address 192.168.105.3 255.255.255.0
  peer ip address 192.168.105.2 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
interface vlan 120
  description Upstream VLAN_120—Clients and VIPs
  ip address 192.168.120.3 255.255.255.0
  alias 192.168.120.1 255.255.255.0
  peer ip address 192.168.120.2 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.70 192.168.120.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  service-policy input NAT_POLICY
  no shutdown
```

Test Procedure

The procedure used to perform the Distributed Etherchannel SPAN—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Start a packet capture on the internal interface used by the ACE.

Step 3 Start a stream of UDP traffic to the load balancer. After the stream has completed, verify that verify that 5000 packets were transmitted and 5000 packets were received.

Step 4 Stop the packet capture and verify that 10000 packets were captured.

Step 5 On the aggregation Supervisor 720, start a packet capture on the NAM.

Step 6 Start a stream of TCP SYN's to IP address 192.168.120.139. After the stream has completed, verify that 20000 SYN's were sent.

Step 7 Stop the packet capture and verify that 20000 packets were captured.

Expected Results

The following test results are anticipated:

- We expect the same number of packets to be seen on the packet capture as was sent from the test tool.

Results

Distributed Etherchannel SPAN—ACE passed with exception. The following exceptions were noted: CSCsm95456.

Distributed Etherchannel SPAN—CSM

A SPAN session allows you to mirror packets from a port, vlan or etherchannel to another port for packet capture and analysis.

This test verified the functionality of the SPAN feature against different protocols traversing distributed etherchannels, captured on the load balancer's internal interface. The packet captured is used in conjunction with the test tool output to verify that the traffic was successfully replicated.

Relevant CSM Configuration

```
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 105 client
  ip address 192.168.105.12 255.255.255.0 alt 192.168.105.13 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
  ip address 10.86.83.13 255.255.255.0 alt 10.86.83.14 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 serverfarm ICMP
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  probe ICMP
 !
 vserver L3
  virtual 192.168.120.240 any
  serverfarm ICMP
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Distributed Etherchannel SPAN—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Start a packet capture on the internal Portchannel used by the CSM.

Step 3 Start a stream of UDP traffic to the load balancer. After the stream has completed, verify that verify that 5000 packets were transmitted and 5000 packets were received.

Step 4 Stop the packet capture and verify that 10000 packets were captured.

Step 5 On the aggregation Supervisor 720, start a packet capture on the NAM.

Step 6 Start a stream of TCP SYN’s to IP address 192.168.120.240. After the stream has completed, verify that 20000 SYN’s were sent.

Step 7 Stop the packet capture, started in an earlier step, and verify that 20000 packets were captured.

Expected Results

The following test results are anticipated:

- We expect the same number of packets to be seen on the packet capture as was sent from the test tool.

Results

Distributed Etherchannel SPAN—CSM passed.

Traffic Decisions

Basic Functionality tests verify that the load balancer module can perform the features constituting or serving as the basis or starting point for load balancer implementation. Such features include CLI, device management, SPAN, and traffic decisions.

The ACE uses several configurational elements to make decisions regarding traffic. Specific areas covered by this certification are failaction purge, ICMP, idle timeout and route health injection (RHI). The failaction command determines the action that the load balancer module takes if a real server goes down. The ICMP protocol can be used to test connectivity and to deliver error, control, and informational messages. The inactivity timeout or idle timer is the minimum length of time that an idle connection will remain open before the resource is reclaimed. The load balancer can advertise the IP address of the virtual server as the host route using the loadbalance VIP advertise command in policy map class configuration. This function is used with RHI to allow the load balancer to advertise the availability of a VIP address throughout the network.

This section contains the following topics:

- Failaction Purge—ACE, page 23
- Failaction Purge—CSM, page 25
- Idle Timeout—ACE, page 27
- Idle Timeout—CSM, page 29
- Route Health Injection—ACE, page 31
- Route Health Injection—CSM, page 38

Failaction Purge—ACE

Failaction Purge allows the load balancer to clear long lived connections to a server that has failed. Without Failaction Purge enabled, the long lived connections would remain until they timeout or close themselves. This is more important for UDP because it is a connectionless protocol. If there were a probe failure UDP connections to the failed server would remain for an extended period of time.

This test verified that the load balancer purged connections to a failed server, when Failaction Purge was configured. Multiple protocols were tested.

Relevant Load Balancer Configuration

```
probe icmp FA-PURGE-ICMP
  ip address 172.29.0.253 routed
  interval 5
  passdetect interval 2
parameter-map type connection INFINITE-IDLE
  set timeout inactivity 0
  set tcp timeout half-closed 30
serverfarm host FA-PURGE
  failaction purge
  rserver BRG-11
    inservice
  rserver BRG-14
    probe FA-PURGE-ICMP
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-244
    probe FA-PURGE-ICMP
    inservice
  rserver RT-151
    inservice
  rserver RT-154
    probe FA-PURGE-ICMP
    inservice
class-map match-all FA-PURGE-VIP_113:ANY
  2 match virtual-address 192.168.120.113 any
policy-map type loadbalance first-match PLBSF_FA-PURGE
  class class-default
    serverfarm FA-PURGE
policy-map multi-match SH-Gold-VIPs
  class FA-PURGE-VIP_113:ANY
    loadbalance vip inservice
    loadbalance policy PLBSF_FA-PURGE
    nat dynamic 1 VLAN 120
    connection advanced-options INFINITE-IDLE
```

Test Procedure

The procedure used to perform the Failaction Purge—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Verify that select serverfarms are configured to failaction purge connections on the load balancer.

Step 4 On a Linux client, launch a test tool that will generate a series of HTTP and UDP connections. This will generate persistent long lived flows for both HTTP and UDP.

Step 5 Verify that the connections are load balanced across all servers.

Step 6 Change the probe IP address to an address that will cause the probe fail. Half of the configured servers use this probe and should fail.

Step 7 Verify that half of the servers are in a PROBE_FAILED state. Repeat the show commands in quick succession to verify that any open connections to the failed servers have been purged.

Step 8 Bring the servers back to an operational status and remove failaction purge.

Step 9 Stop the client tool, started in an earlier step. Clear select load balancer counters and connections.

Step 10 On a Linux client, launch a test tool that will generate a series of HTTP and UDP connections. This will generate persistent long lived flows for both HTTP and UDP.

Step 11 Verify that the connections are load balanced across all servers.

Step 12 Change the probe IP address to an address that will cause the probe fail. Half of the configured servers use this probe and should fail.

Step 13 Verify that half of the servers are in a PROBE_FAILED state. Repeat the show commands in quick succession to verify that none of the open connections to the failed servers have been purged.

Step 14 On select serverfarms, change the serverfarm setting back to failaction purge.

Step 15 Bring the failed servers back to an operational status.

Step 16 Stop the client tool, started in an earlier step. Verify through the test tool’s GUI that no errors were seen.

Step 17 Clear connections and select counters on the load balancer.

Expected Results

The following test results are anticipated:

- We expect that open connections to a failed server will be purged before the idle timer expires while failaction purge is configured.
- We expect that open connections to a failed server will remain until the idle timer expires or the application closes them with the default setting of no failaction purge.
- We expect that the load balancer will not crash or become unresponsive.

Results

Failaction Purge—ACE passed.

Failaction Purge—CSM

Failaction Purge allows the load balancer to clear long lived connections to a server that has failed. Without Failaction Purge enabled, the long lived connections would remain until they timeout or close themselves. This is more important for UDP because it is a connectionless protocol. If there were a probe failure UDP connections to the failed server would remain for an extended period of time.

This test verified that the load balancer purged connections to a failed server, when Failaction Purge was configured. Multiple protocols were tested.

Relevant Load Balancer Configuration

```
serverfarm PRED-CONNS
 nat server
 nat client CLIENT_NAT
 predictor leastconns
 real name BRG-LINUX-11
  inservice
 real name BRG-LINUX-12
  inservice
 real name BRG-LINUX-13
  inservice
 real name BRG-LINUX-14
  inservice
 real name BRG-LINUX-15
  inservice
 real name LOCAL-LINUX-240
  inservice
 real name LOCAL-IIS-241
  inservice
 real name LOCAL-LINUX-242
  inservice
 real name LOCAL-IIS-243
  inservice
 real name LOCAL-LINUX-244
  inservice
 real name LOCAL-IIS-245
  inservice
 real name RT-LINUX-151
  inservice
 real name RT-IIS-152
  inservice
 real name RT-LINUX-153
  inservice
  real name RT-LINUX-154
   health probe PRED-PING
   inservice
  probe HTTP
!
 serverfarm PRED-CONNS-UDP
  nat server
  nat client CLIENT_NAT
  predictor leastconns
  failaction purge
  real name BRG-LINUX-11 2222
   inservice
  real name BRG-LINUX-12 2222
   inservice
  real name LOCAL-LINUX-240 2222
   inservice
  real name LOCAL-LINUX-242 2222
   inservice
  real name LOCAL-LINUX-244 2222
   inservice
  real name RT-LINUX-151 2222
   inservice
  real name RT-LINUX-153 2222
   inservice
  real name RT-LINUX-154 2222
   health probe PRED-PING
   inservice
  probe ICMP
 vserver PRED-CONNS-UDP
  virtual 192.168.120.204 udp 0
  unidirectional
  serverfarm PRED-CONNS-UDP
  idle 300
  pending 60
  persistent rebalance
  inservice
```

Test Procedure

The procedure used to perform the Failaction Purge—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Add failaction purge to select serverfarms, if needed.

Step 4 Verify that select serverfarms are configured to failaction purge connections on the load balancer.

Step 5 On a Linux client, launch a test tool that will generate a series of HTTP and UDP connections. This will generate persistent long lived flows for both HTTP and UDP.

Step 6 Verify that the connections are load balanced across all servers.

Step 7 Change the probe IP address to an address that will cause the probe fail. One of the configured servers uses this probe and should fail.

Step 8 Verify that one of the servers is in a PROBE_FAILED state. Repeat the show commands in quick succession to verify that any open connections to the failed server have been purged.

Step 9 Bring the servers back to an operational status and remove failaction purge.

Step 10 Stop the client tool, started in an earlier step. Clear select load balancer counters and connections.

Step 11 On a Linux client, launch a test tool that will generate a series of HTTP and UDP connections. This will generate persistent long lived flows for both HTTP and UDP.

Step 12 Verify that the connections are load balanced across all servers.

Step 13 Change the probe IP address to an address that will cause the probe fail. One of the configured servers uses this probe and should fail.

Step 14 Verify that one of the servers are in a PROBE_FAILED state. Repeat the show commands in quick succession to verify that any open connections to the failed server have not been purged.

Step 15 On select serverfarms, change the serverfarm setting back to failaction purge.

Step 16 Bring the failed server back to an operational status.

Step 17 Verify through the test tool's GUI that no errors were seen.

Step 18 Stop the client tool. Clear connections and select counters on the load balancer.

Expected Results

The following test results are anticipated:

- We expect open connections to a failed rserver will be purged before the idle timer expires while the failaction purge is configured.
- We expect open connections to a failed rserver will remain until the idle timer expires or the application closes them with the default setting of no failaction purge.
- We expect that the load balancer will not crash or become unresponsive.

Results

Failaction Purge—CSM passed.

Idle Timeout—ACE

The inactivity timeout or idle timer is the minimum length of time that an idle connection will remain open before the resource is reclaimed. The default idle timers are 3600, 120, and two seconds for TCP, UDP, and ICMP respectively, which can range from zero to 4294967294 seconds. A setting of zero disables the timer and allows these connections to remain open indefinitely. Care needs to be taken to prevent this from consuming all of the resources on the load balancer. The resources can be reclaimed by manually changing the idle timer back to a non zero value.

This test verified that the load balancer closed connections based on the configured timeout value.

Test Procedure

The procedure used to perform the Idle Timeout —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the load balancer, clear select counters.

Step 3 On the aggregation Supervisor 720, remove any existing SPAN configuration for session 2.

Step 4 Configure a monitor session on the aggregation Supervisor 720. All traffic on interface Te1/1 (receive and transmit) will be copied to destination network analysis module (NAM) data port 2.

Step 5 On the access Supervisor 720, start a packet capture on the network analysis module (NAM) using data port 2.

Step 6 On a Linux client, initiate HTTP traffic, so that the first GET will be sent immediately and each subsequent GET will follow 58 seconds later.

Step 7 After the stream has completed, stop the packet capture and view the results. Verify that only one TCP connection is made and that the second and subsequent HTTP GET’s were successful coming in at a frequency of 58 seconds.

Step 8 Copy and paste the client’s tool results. The output will show only one TCP connection and no errors.

Step 9 On the aggregation Supervisor 720, start a packet capture on the NAM using data port 2.

Step 10 On a Linux client, initiate HTTP traffic so that the first GET will be sent immediately and each subsequent GET will follow 65 seconds later.

Step 11 With the idle timer set to 60 seconds, the load balancer will send a reset before the second GET can be transmitted. Verify on the client’s tool output that it is getting resets and that multiple connection attempts are made.

Step 12 Stop the packet capture and view the results. Verify that the load balancer sends a reset approximately 60 seconds after the connection is opened.

Step 13 On a Linux client initiate client traffic to generate a number of open TCP and UDP connections.

Step 14 Verify that the serverfarm shows a number of open connections.

Step 15 Shut down the access Supervisor 720 access port to the Linux client, which will abnormally terminate any open connections.

Step 16 The load balancer will timeout half open connections in 30 seconds and will idle timeout established connections after 60 seconds of inactivity. Verify that all of the UDP connections have been closed within a short time after the idle period expires (60 seconds).

Step 17 On the primary load balancer, after the UDP conns have been removed, quickly look at the open TCP connections on the serverfarm.

Step 18 On the access Supervisor 720, stop the client generated traffic and enable the access port.

Step 19 Clear select load balancer counters.

Step 20 Change the idle timeout for servers IDLE-TCP and IDLE-UDP to 120 seconds.

Step 21 On two Linux clients, open up a telnet session on one and four UDP connections on the other to virtual address 192.168.120.111. Let these connections remain idle and confirm that have been closed within a short time after the idle period expires (120 seconds).

Step 22 Change the idle timeout to infinite for servers IDLE-TCP and IDLE-UDP.

Step 23 On two Linux clients, open up a telnet session on one and four UDP connections on the other, to virtual address 192.168.120.111. Let these connections remain idle for at least 180 seconds. Verify that they do not timeout.

Step 24 Let these connections remain idle overnight. The next morning, verify that the original five connections are still present.

Step 25 Change the idle time back to 60 seconds for server IDLE-UDP.

Step 26 Verify that only the UDP connections have been closed within a short time after the idle period expires (60 seconds).

Step 27 Change the idle time back to 60 seconds for server IDLE-TCP.

Step 28 Verify that all of the connections have been closed within a short time after the idle period expires (60 seconds).

Expected Results

The following test results are anticipated:

- We expect idle connections to remain in the connection table until the idle timer expires.
- We expect idle connections exceeding the idle timer to be closed.
- We expect these connections not to timeout when the idle timer is set to infinite.
- We expect connections using an infinite timer to be reclaimed by changing the idle timer back to a non zero number.
- We expect that the load balancer will not crash or become unresponsive.

Results

Idle Timeout —ACE passed.

Idle Timeout—CSM

The inactivity timeout or idle timer is the minimum length of time that an idle connection will remain open before the resource is reclaimed. The default idle timers are 3600, 120, and two seconds for TCP, UDP, and ICMP respectively, which can range from zero to 4294967294 seconds. A setting of zero disables the timer and allows these connections to remain open indefinitely. Care needs to be taken to prevent this from consuming all of the resources on the load balancer. The resources can be reclaimed by manually changing the idle timer back to a non zero value.

This test verified that the load balancer closed connections based on the configured timeout value.

Relevant Load Balancer Configuration

```
!!
 real LOCAL-IIS-241
  address 172.29.0.241
  inservice
 real RT-LINUX-153
  address 172.28.0.153
  inservice
 real LOCAL-LINUX-240
  address 172.29.0.240
  inservice
 real RT-LINUX-154
  address 172.28.0.154
  inservice
!
 serverfarm IDLE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm IDLE-UDP
```

```
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-LINUX-154
   inservice
!
 vserver IDLE
  virtual 192.168.120.211 tcp 0
  serverfarm IDLE
  idle 60
  persistent rebalance
  inservice
!
 vserver IDLE-UDP
  virtual 192.168.120.211 udp 0
  serverfarm IDLE-UDP
  idle 60
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Idle Timeout—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the load balancer, clear select counters.

Step 3 On the access Supervisor 720, start a packet capture on the network analysis module (NAM).

Step 4 On a Linux client, initiate HTTP traffic, so that the first GET will be sent immediately and each subsequent GET will follow 58 seconds later.

Step 5 After the stream has completed, stop the packet capture and view the results. Verify that only one TCP connection is made and that the second and subsequent HTTP GET’s were successful coming in at a frequency of 58 seconds.

Step 6 Copy and paste the client’s tool results. The output will show only one TCP connection and no errors.

Step 7 On the aggregation Supervisor 720, start a packet capture on the NAM using data port 2.

Step 8 On a Linux client, initiate HTTP traffic so that the first GET will be sent immediately and each subsequent GET will follow 65 seconds later.

Step 9 With the idle timer set to 60 seconds, the load balancer will send a reset before the second GET can be transmitted. Verify on the client’s tool output that it is getting resets and that multiple connection attempts are made.

Step 10 Stop the packet capture and view the results. Verify that the load balancer sends a reset approximately 60 seconds after the connection is opened.

Step 11 On a Linux client initiate client traffic to generate a number of open TCP and UDP connections.

Step 12 Verify that the serverfarm shows a number of open connections.

Step 13 Shut down the access Supervisor 720 access port to the Linux client, which will abnormally terminate any open connections.

Step 14 The load balancer will timeout half open connections in 30 seconds and will idle timeout established connections after 60 seconds of inactivity. Verify that all of the UDP connections have been closed within a short time after the idle period expires (60 seconds).

Step 15 On the primary load balancer, after the UDP conns have been removed, quickly look at the open TCP connections on the serverfarm.

Step 16 On the access Supervisor 720, stop the client generated traffic and enable the access port.

Step 17 Clear select load balancer counters.

Step 18 Change the idle timeout for servers IDLE-TCP and IDLE-UDP to 120 seconds.

Step 19 On two Linux clients, open up a telnet session on one and four UDP connections on the other to virtual address 192.168.120.111. Let these connections remain idle and confirm that have been closed within a short time after the idle period expires (120 seconds).

Step 20 Change the idle timeout to infinite for the serverfarms handling TCP and UDP traffic.

Step 21 On two Linux clients, open up a telnet session on one and four UDP connections on the other, to virtual address 192.168.120.111. Let these connections remain idle for at least 180 seconds. Verify that they do not timeout.

Step 22 Let these connections remain idle overnight. The next morning, verify that the original five connections are still present.

Step 23 Change the idle time back to 60 seconds for the serverfarms handling TCP and UDP traffic.

Step 24 Verify that the TCP and UDP connections have been closed within a short time after the idle period expires (60 seconds).

Expected Results

The following test results are anticipated:

- We expect idle connections to remain in the connection table until the idle timer expires.
- We expect idle connections exceeding the idle timer to be closed.
- We expect these connections not to timeout when the idle timer is set to infinite.
- We expect connections using an infinite timer to be reclaimed by changing the idle timer back to a non zero number.
- We expect that the load balancer will not crash or become unresponsive.

Results

Idle Timeout—CSM passed.

Route Health Injection—ACE

The load balancer can advertise the IP address of the virtual server as a static host route injected into the local MSFC. This function is commonly used with OSPF or similar protocol to advertise the availability of a VIP address throughout the network. RHI is used primarily in environments where the VIP is in a subnet that is not directly attached to the MSFC. This test verified that the host route was properly injected or removed in various conditions, probe failure, admin down, ft switchover, etc.

Relevant ACE Configuration

```
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
parameter-map type http PERSIST-REBALANCE
  persistence-rebalance
serverfarm host RHI
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-154
    inservice
class-map type http loadbalance match-all URL*_L7
  2 match http url .*
class-map match-all RHI_125.100:80
  2 match virtual-address 192.168.125.100 tcp eq www
class-map match-all RHI_125.101:80
  2 match virtual-address 192.168.125.101 tcp eq www
class-map match-all RHI_125.102:80
  2 match virtual-address 192.168.125.102 tcp eq www
class-map match-all RHI_125.103:80
  2 match virtual-address 192.168.125.103 tcp eq www
class-map match-all RHI_125.104:80
  2 match virtual-address 192.168.125.104 tcp eq www
class-map match-all RHI_125.105:80
  2 match virtual-address 192.168.125.105 tcp eq www
class-map match-all RHI_125.106:80
  2 match virtual-address 192.168.125.106 tcp eq www
class-map match-all RHI_125.107:80
  2 match virtual-address 192.168.125.107 tcp eq www
class-map match-all RHI_125.108:80
  2 match virtual-address 192.168.125.108 tcp eq www
class-map match-all RHI_125.109:80
  2 match virtual-address 192.168.125.109 tcp eq www
class-map match-all RHI_125.110:80
  2 match virtual-address 192.168.125.110 tcp eq www
class-map match-all RHI_125.111:80
  2 match virtual-address 192.168.125.111 tcp eq www
class-map match-all RHI_125.112:80
  2 match virtual-address 192.168.125.112 tcp eq www
class-map match-all RHI_125.113:80
  2 match virtual-address 192.168.125.113 tcp eq www
class-map match-all RHI_125.114:80
  2 match virtual-address 192.168.125.114 tcp eq www
class-map match-all RHI_125.115:80
  2 match virtual-address 192.168.125.115 tcp eq www
class-map match-all RHI_125.116:80
  2 match virtual-address 192.168.125.116 tcp eq www
class-map match-all RHI_125.117:80
  2 match virtual-address 192.168.125.117 tcp eq www
class-map match-all RHI_125.118:80
  2 match virtual-address 192.168.125.118 tcp eq www
class-map match-all RHI_125.119:80
  2 match virtual-address 192.168.125.119 tcp eq www
class-map match-all RHI_125.120-127
  2 match virtual-address 192.168.125.127 255.255.255.248 any
policy-map type loadbalance first-match PLBSF_RHI
  class URL*_L7
    serverfarm RHI
policy-map multi-match SH-Gold-VIPs3
  class RHI_125.100:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.101:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.102:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.103:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.104:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.105:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.106:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.107:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.108:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.109:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.110:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.111:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.112:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.113:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.114:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.115:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.116:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.117:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.118:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.119:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.120-127
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
interface vlan 105
  ip address 192.168.105.2 255.255.255.0
  peer ip address 192.168.105.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
!
6K-1
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
!
interface Vlan105
 ip address 192.168.105.251 255.255.255.0
 no ip proxy-arp
 standby 105 ip 192.168.105.254
 standby 105 priority 200
 standby 105 preempt
!
router ospf 105
 router-id 192.168.105.251
 log-adjacency-changes
 nsf
 redistribute static subnets route-map rhi-vip
 network 192.168.105.0 0.0.0.255 area 0
 network 192.168.106.0 0.0.0.255 area 0
!
access-list 104 permit ip 172.31.111.0 0.0.0.255 any
access-list 105 permit ip 192.168.125.0 0.0.0.255 any
access-list 106 permit ip 172.28.125.0 0.0.0.255 any
access-list 107 permit ip 192.168.140.0 0.0.0.255 any
!
route-map rhi-vip permit 10
 match ip address 105 106 107 104
!
6K3
!
interface Vlan105
 ip address 192.168.105.253 255.255.255.0
!
router ospf 105
 router-id 192.168.105.253
 log-adjacency-changes
 network 192.168.105.0 0.0.0.255 area 0
 network 192.168.106.0 0.0.0.255 area 0
!
```

Test Procedure

The procedure used to perform the Route Health Injection—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary CAT in the SH-Gold context, verify that the vserver (class: RHI_125.100:80) is in service and configured to advertise the route when active by issuing the **show service-policy SH-Gold-VIPs3 detail | begin RHI** command.

Step 3 On the 6k1 (primary CAT—sh-ace-6k-1) this VIP is listed as a static route. On 6k3 (upstream CAT—sh-ace-6k-3) it is an OSPF route with a next hop 192.168.105.2. Verify this by issuing the following commands:

```
show ip route static (6k1)
show ip route ospf   (6k3)
```

Step 4 Take the vserver out of service and verify the static route is removed from the routing table on 6k1 by issuing the following commands:

```
config
policy-map multi-match SH-Gold-VIPs3
class RHI_125.100:80
no loadbalance vip inservice
end
show ip route static  (6K1)
```

Step 5 Put the vserver back in service and verify the static route is in the routing table on 6k1 by issuing the following commands:

```
config
policy-map multi-match SH-Gold-VIPs3
class RHI_125.100:80
loadbalance vip inservice
end
show ip route static
```

Step 6 Take the vserver out of service by adding a probe to serverfarm RHI. Once the serverfarm has failed verify that the static route is removed from the routing table on 6k1. Issuing the following commands:

```
config
serverfarm RHI
probe FORCED-FAIL
end
show ip route static | inc .100
```

Step 7 Put the vserver back in service by removing the probe forced-fail from serverfarm RHI. Verify that the static route is in the routing table on 6k1 by issuing the following commands:

```
config
serverfarm RHI
no probe FORCED-FAIL
end
show ip route static
```

Step 8 The active ACE when injecting a VIP into the routing table will use the alias address of the VLAN if available. If an alias address is not used it will use the circuit IP. In this test there is no ACE alias IP for VLAN 105, so it will use 192.168.105.2 and .3 for the primary and standby modules, respectively. Force a fault tolerant failover by resetting the module. Issue the **hw-module module 1 rest** command.

Step 9 Verify that all 21 routes (RHI VIPs) are learned on 6k1 and 6k3 is using 192.168.120.3 as the next hop address.

Step 10 On a Linux client send a burst of traffic and verify through the CLI output that these requests were successful.

Step 11 Once the primary ACE reloads verify that it preempts and becomes active by issuing the **show ft group detail** command.

Step 12 Verify that all 21 routes are learned on 6k1 and 6k3 using 192.168.120.2 as the next hop address. Issue the following commands:

```
show ip route static | inc 125. (6k1)
show ip route ospf | inc 125.   (6k3)
```

Step 13 On a Linux client send a burst of traffic and verify through the CLI output that these requests were successful.

Step 14 The primary CAT has two SUP-720 running in redundancy state. These steps will verify that the ACE injected routes persist after a supervisor failover. Verify that the supervisors are redundant and the static route is in the routing table using 192.168.105.2 as a next hop. Issue the following commands:

```
show redundancy states
show module
show ip route static | inc 125.
```

Step 15 Force a supervisor failover. Verify that all 21 injected routes on 6k1 and 6k3 still have a route using 192.168.105.2 as a next hop. Issue the following commands:

```
redundancy force-switchover
show redundancy states
show module
show ip route static | inc 125. (6k1)
show ip route ospf | inc 125.   (6k3)
```

Step 16 Wait until the SUP-720 that was failed comes back online in a standby mode. Verify the that primary Supervisor 720 comes back online in a standby state and that all 21 injected routes on 6k1 and 6k3 still have a route using 192.168.105.2 as a next hop. Issue the following commands:

```
show redundancy states
show module
show ip route static | inc 125.
show ip route ospf  | inc 125.
```

Step 17 Force a supervisor failover to switch back to the primary. Verify that all 21 injected routes are still using 192.168.105.2 as a next hop. Issue the following commands:

```
redundancy force-switchover
show redundancy states
show module
show ip route static | inc 125.
```

Expected Results

The following test results are anticipated:

- We expect host routes to be injected when the vservers are active.
- We expect the host routes to be removed when the vserver becomes out of service.
- We expect the appropriate route and gateway to be injected during a redundancy failure.
- We expect the host routes to remain after a failover of redundant Supervisor 720’s.

Results

Route Health Injection—ACE passed.

Route Health Injection—CSM

The load balancer can advertise the IP address of the virtual server as a static host route injected into the local MSFC. This function is commonly used with OSPF or similar protocol to advertise the availability of a VIP address throughout the network. RHI is used primarily in environments where the VIP is in a subnet that is not directly attached to the MSFC. This test verified that the host route was properly injected or removed in various conditions, probe failure, no inservice, ft switchover, etc.

Relevant CSM Configuration

```
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
```

```
!
 vlan 105 client
  ip address 192.168.105.12 255.255.255.0 alt 192.168.105.13 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
  ip address 10.86.83.13 255.255.255.0 alt 10.86.83.14 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 serverfarm RHI
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
  probe ICMP-RTR
!
 vserver RHI-21
  virtual 172.31.111.201 tcp 0 service ftp
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  inservice
!
 vserver RHI-80
  virtual 172.31.111.200 tcp www
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  slb-policy RHI
  inservice
!
 vserver RHI-TERM
  virtual 172.31.111.202 tcp 0 service termination
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  inservice
!
6K-1
!
redundancy
```

```
 mode sso
 main-cpu
  auto-sync running-config
!
interface Vlan105
 ip address 192.168.105.251 255.255.255.0
 no ip proxy-arp
 standby 105 ip 192.168.105.254
 standby 105 priority 200
 standby 105 preempt
!
router ospf 105
 router-id 192.168.105.251
 log-adjacency-changes
 nsf
 redistribute static subnets route-map rhi-vip
 network 192.168.105.0 0.0.0.255 area 0
 network 192.168.106.0 0.0.0.255 area 0
!
access-list 104 permit ip 172.31.111.0 0.0.0.255 any
access-list 105 permit ip 192.168.125.0 0.0.0.255 any
access-list 106 permit ip 172.28.125.0 0.0.0.255 any
access-list 107 permit ip 192.168.140.0 0.0.0.255 any
!
route-map rhi-vip permit 10
 match ip address 105 106 107 104
!
6K3
!
interface Vlan105
 ip address 192.168.105.253 255.255.255.0
!
router ospf 105
 router-id 192.168.105.253
 log-adjacency-changes
 network 192.168.105.0 0.0.0.255 area 0
 network 192.168.106.0 0.0.0.255 area 0
!
```

Test Procedure

The procedure used to perform the Route Health Injection—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary CSM verify that the vserver RHI-80 is in service and configured to advertise host routes via RHI by issuing the **show mod csm 8 vserver name RHI-80 detail** command.

Step 3 On the 6k1 (primary CAT—sh-ace2-6k-1) this VIP is listed as a static route and on 6k3 (upstream CAT—sh-ace2-6k3) it is an OSPF route with both having a next hop of 192.168.105.12. Verify this by issuing the following commands:

```
show ip route static | beg 172.31.0.0/32 (6k1)
show ip route ospf | beg 172.31.0.0/32   (6k3)
```

Step 4 Take the vserver out of service and verify that the static route is removed from the routing table on 6k1 by issuing the following commands:

```
config t
mod csm 8
vserver RHI-80
no inservice
end
show ip route static | beg 172.31.0.0/32  (6K1)
```

Step 5 Put the vserver back in service and verify that the static route is in the routing table on 6k1 by issuing the following commands:

```
config t
mod csm 8
vserver RHI-80
inservice
end
show ip route static | beg 172.31.0.0/32  (6K1)
```

Step 6 Take the vserver out of service by adding a probe to serverfarm RHI-80. Once the serverfarm has failed verify that the static route is removed from the routing table on 6k1. Issuing the following commands:

```
config t
mod csm 8
serverfarm RHI
probe FORCED-FAIL
end
show ip route static | beg 172.31.0.0/32  (6K1)
```

Step 7 Put the vserver back in service by removing the probe FORCED-FAIL from serverfarm RHI-80. Verify that the static route is in the routing table on 6k1 by issuing the following commands:

```
config t
mod csm 8
serverfarm RHI
no probe FORCED-FAIL
end
show ip route static | beg 172.31.0.0/32  (6K1)
```

Step 8 The active CSM when injecting a VIP into the routing table will use the alias address of the VLAN if available. If an alias address is not used it will use the circuit IP. In this test there is no alias IP for VLAN 105, so it will use 192.168.105.12 and .13 for the primary and standby (6K-2_ACE1) CSM, respectively.

Force a fault tolerant failover by resetting the primary CSM by issuing the **hw-module module 8 reset** command and then quickly move through the next several steps.

Step 9 Verify that all 3 routes learned on 6k1 and 6k3 are using 192.168.120.13 as the next hop address.

Step 10 On a Linux client send a burst of traffic and verify through the CLI output that these requests were successful.

Step 11 Once the primary CSM is finished booting verify that it preempts and becomes active by issuing the show mod csm 8 ft detail command:

Step 12 Verify that all 3 routes are learned on 6k1 and 6k3 are using 192.168.120.12 as the next hop address. Issue the following commands:

```
show ip route static | beg 172.31.0.0/32 (6k1)
show ip route ospf | beg 172.31.0.0/32   (6k3)
```

Step 13 On a Linux client send a burst of traffic and verify through the CLI output that these requests were successful.

Step 14 The primary CAT (sh-ace-6k-1) has two SUP-720 running in redundancy state. These steps will verify that the CSM injected routes persist after a supervisor failover. Verify that the supervisors are redundant and the static route is in the routing table using 192.168.105.12 as a next hop. Issue the following commands:

```
show redundancy states
show module
show ip route static | beg 172.31.0.0/32
```

Step 15 Force a supervisor failover. Verify that all 3 injected routes on 6k1 and 6k3 still have a route using 192.168.105.12 as a next hop. Issue the following commands:

```
redundancy force-switchover
show redundancy states
show module
show ip route static | beg 172.31.0.0/32   (6k1)
show ip route ospf | beg 172.31.0.0/32     (6k3)
```

Step 16 Wait until the SUP-720 that was failed over comes back online in a standby mode. Verify that the primary Supervisor 720 comes back online in a standby state and that all 3 injected routes on 6k1 and 6k3 still have a route using 192.168.105.12 as a next hop. Issue the following commands:

```
show redundancy states
show module
show ip route static | beg 172.31.0.0/32
show ip route ospf  | beg 172.31.0.0/32
```

Step 17 Force a supervisor failover to switch back to the primary. Verify that all 3 injected routes are still using 192.168.105.12 as a next hop. Issue the following commands:

```
redundancy force-switchover
show redundancy states
show module
show ip route static | beg 172.31.0.0/32
```

Expected Results

The following test results are anticipated:

- We expect host routes to be injected when the vservers are active.
- We expect the host routes to be removed when the vserver becomes out of service.
- We expect the appropriate route and gateway to be injected during a redundancy failure.
- We expect the host routes to remain after a failover of redundant Supervisor 720's.

Results

Route Health Injection—CSM passed.

Health and Redundancy

Testing related to the health of the load balancer.

This section contains the following topics:

- Backup Serverfarm, page 43
- Configuration Syncronization, page 46
- Probes, page 53
- Redundancy, page 91
- Tracking, page 99

Backup Serverfarm

In the event of a failure of the primary serverfarm the backup serverfarm should be used. The backup serverfarm is made up of a collection of rservers which are unique to the primary serverfarm. This feature is used to prevent a full site outage.

This section contains the following topics:

- Backup Serverfarm—ACE, page 43
- Backup Serverfarm—CSM, page 45

Backup Serverfarm—ACE

This verified the ability of the load balancer to switchover incoming connections to a backup serverfarm and back again depending on the state of the primary serverfarm.

Relevant Load Balancer Configuration

```
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
```

```
serverfarm host SORRY
  rserver LOCAL-240
    inservice
serverfarm host SORRY-BACK
  rserver LOCAL-243
    inservice
  rserver RT-151
    inservice
class-map match-all SORRY-VIP_137:80
  2 match virtual-address 192.168.120.137 tcp eq www
policy-map type loadbalance first-match PLBSF_SORRY
  class class-default
    serverfarm SORRY backup SORRY-BACK
policy-map multi-match SH-Gold-VIPs
  class SORRY-VIP_137:80
    loadbalance vip inservice
    loadbalance policy PLBSF_SORRY
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
```

Test Procedure

The procedure used to perform the Backup Serverfarm—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that all servers in the primary and backup serverfarms are operational.

Step 3 Start a series of HTTP v1.0 GET requests on two clients.

Step 4 Verify that connections are being made to the primary serverfarm and NOT to the backup serverfarm.

Step 5 Configure the probe FORCED-FAIL on the primary serverfarm, causing all of its servers to fail.

Step 6 The servers in the primary serverfarm will fail, forcing traffic to the backup serverfarm. The state of the VIP will remain INSERVICE because the load balancer takes into account the state of the backup serverfarm when reporting status.

Step 7 Remove the probe FORCED-FAIL from the primary serverfarm.

Step 8 Verify that the servers within the primary serverfarm are operational again and accepting connections, while the servers within the backup serverfarm are no longer hosting connections.

Step 9 Configure the probe FORCED-FAIL on the backup serverfarm, forcing all of its servers to fail.

Step 10 Verify that connections are still being made to the primary serverfarm.

Step 11 Configure the probe FORCED-FAIL on the primary serverfarm, causing all of its servers to fail.

Step 12 Verify that both serverfarms are down and that the VIP state is OUTOFSERVICE.

Step 13 Remove the probe FORCED-FAIL from the primary and backup serverfarms.

Step 14 Verify that both serverfarms are up and that the VIP state is INSERVICE.

Expected Results

The following test results are anticipated:

- We expect traffic to be forwarded to a backup serverfarm only when the probes have failed on the primary serverfarm.

- We expect traffic to be forwarded back to the primary serverfarm when it becomes operational.
- We expect that the load balancer will not crash or become unresponsive.

Results

Backup Serverfarm—ACE passed.

Backup Serverfarm—CSM

This verified the ability of the load balancer to switchover incoming connections to a backup serverfarm and back again depending on the state of the primary serverfarm.

Relevant ACE Configuration

```
!
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 real LOCAL-LINUX-244
  address 172.29.0.244
  inservice
 real LOCAL-IIS-243
  address 172.29.0.243
  inservice
!
 serverfarm SORRY
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
!
 serverfarm SORRY-BACK
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
!
 vserver SORRY
  virtual 192.168.120.228 tcp www
  serverfarm SORRY backup SORRY-BACK
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Backup Serverfarm—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that all servers in the primary and backup serverfarms are operational.

Step 3 Start a series of HTTP v1.0 GET requests on two clients.

Step 4 Verify that connections are being made to the primary serverfarm and NOT to the backup serverfarm.

Step 5 Configure the probe FORCED-FAIL on the primary serverfarm, causing all of its servers to fail.

Step 6 The servers in the primary serverfarm will fail, forcing traffic to the backup serverfarm. The state of the VIP will remain INSERVICE because the load balancer takes into account the state of the backup serverfarm when reporting status.

Step 7 Remove the probe FORCED-FAIL from the primary serverfarm.

Step 8 Verify that the servers within the primary serverfarm are operational again and accepting connections, while the servers within the backup serverfarm are no longer hosting connections.

Step 9 Configure the probe FORCED-FAIL on the backup serverfarm, forcing all of its servers to fail.

Step 10 Verify that connections are still being made to the primary serverfarm.

Step 11 Configure the probe FORCED-FAIL on the primary serverfarm, causing all of its servers to fail.

Step 12 Verify that both serverfarms are down and that the VIP state is OUTOFSERVICE.

Step 13 Remove the probe FORCED-FAIL from the primary and backup serverfarms.

Step 14 Verify that both serverfarms are up and that the VIP state is OPERATIONAL.

Expected Results

The following test results are anticipated:

- We expect traffic to be forwarded to a backup serverfarm only when the probes have failed on the primary serverfarm.
- We expect traffic to be forwarded back to the primary serverfarm when it becomes operational.
- We expect that the load balancer will not crash or become unresponsive.

Results

Backup Serverfarm—CSM passed.

Configuration Syncronization

For redundancy to function properly, both members of an FT group must have identical configurations. To facilitate this requirement, the load balancer automatically replicates the active configuration on the standby member using a process called configuration synchronization (config sync). Config sync automatically replicates on the standby member any changes made to the configuration of the active member. After the load balancer syncs the redundancy configuration from the active member to the standby peer, it disables configuration mode on the standby.

This section contains the following topics:

Config Sync Large—ACE

For redundancy to function properly, both members of an FT group must have identical configurations. To facilitate this requirement, the load balancer automatically replicates the active configuration on the standby member using a process called configuration synchronization (config sync). Config sync automatically replicates on the standby member any changes made to the configuration of the active member. After the load balancer syncs the redundancy configuration from the active member to the standby peer, it disables configuration mode on the standby.

This test verified that an load balancer module running a large config, approximately 24K lines, remained synchronized when running in a redundant configuration. This was tested by forcing a config bulk sync to occur in several different ways. The results were then compared for accuracy.

Test Procedure

The procedure used to perform the Config Sync Large—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Add a resource allocation.

Step 3 Verify that the primary and standby load balancer states are active (for the primary) standby hot (for the standby) and that the FT auto-sync is enabled.

Step 4 On both the active and standby load balancers verify that no running config exists.

Step 5 On the active load balancer verify that that the expected checkpoints are present.

Step 6 Restore a large config into the running config.

Step 7 Verify that a message indicating that the rollback was successful is displayed, when the rollback procedure completes.

Step 8 On the active load balancer verify that the FT state for the peer is in a standby hot state.

Step 9 Capture the running config from the context on both the active and standby load balancers. Compare the primary and standby load balancer configs with a diff utility. Verify that the configs are synchronized.

Step 10 On the standby load balancer, verify that there is no startup-config and then reload the standby load balancer.

Step 11 On the active load balancer periodically check the FT status of the standby load balancer.

Step 12 Capture the running config from the context on both the active and standby load balancers. Compare the primary and standby load balancer configs with a diff utility. Verify that the configs are synchronized.

Step 13 On the active and standby load balancers verify that a select group of interfaces, ARP, routes, static sticky table, and probes are up on both modules.

Step 14 On the **active** load balancer, disable auto sync for the running config and then reload the standby load balancer.

Step 15 When the standby load balancer comes back online verify that the context has no running config and that the FT state is standby hot.

Step 16 On the active load balancer, enable FT auto sync for the running config and verify that the FT state indicates that it is in a bulk sync state. Continue to monitor this until finished and the state for the standby load balancer is standby hot.

Step 17 Capture the running config from the context on both the active and standby load balancers. Compare the primary and standby load balancer configs with a diff utility. Verify that the configs are synchronized.

Step 18 This will test the ability of the standby to receive the startup config from the active load balancer. Verify that the *standby load balancer* does not have a startup config and that the *active load balancer* shows FT auto sync startup config configured and enabled.

Step 19 On the active load balancer save the running config to the startup config and force a replication of the startup config to the standby load balancer.

Step 20 Capture the running config from the context on both the active and standby load balancers. Compare the primary and standby load balancer configs with a diff utility. Verify that the configs are synchronized.

Step 21 On the active load balancer, clear the startup config and running config.

Step 22 Verify that the startup config and running config are cleared from both the active and standby load balancers.

Step 23 Remove the resource allocation and then save the config.

Expected Results

The following test results are anticipated:

- We expect the configuration to maintain synchronization after a checkpoint rollback occurs.
- We expect the configuration to maintain synchronization after a reload of the standby load balancer.
- We expect the configuration to maintain synchronization after a bulk sync is forced by enabling FT auto sync running config.
- We expect the startup config to maintain synchronization after a write memory is executed on the active load balancer module.
- We expect that the load balancer will not crash or become unresponsive.

Results

Config Sync Large—ACE passed.

Config Sync Large—CSM

For redundancy to function properly, both members of an FT group must have identical configurations. To facilitate this requirement, the load balancer automatically replicates the active configuration on the standby member using a process called configuration synchronization (config sync). Config sync automatically replicates on the standby member any changes made to the configuration of the active member. After the load balancer syncs the redundancy configuration from the active member to the standby peer, it disables configuration mode on the standby.

This test verified that an load balancer module running a large config, approximately 24K lines, remained synchronized when running in a redundant configuration. This was tested by forcing a config bulk sync to occur in several different ways. The results were then compared for accuracy.**Relevant Load Balancer Configuration**

```
Standby CSM
!
variable SASP_CSM_UNIQUE_ID Cisco-CSM
!
 ft group 2 vlan 900
  priority 100 alt 110
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.252
  track interface GigabitEthernet4/38
  track mode any
!
```

Test Procedure

The procedure used to perform the Config Sync Large—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that the fault-tolerant (FT) status is in-sync.

Step 3 On the active load balancer, load a large config from a flash card into the running config. Verify that the FT status is out-of-sync.

Step 4 On the active load balancer execute a configuration sync.

Step 5 Verify that after the configuration sync completes, the FT status is in-sync.

Step 6 On the standby load balancer clear the load balancer running configuration and add a minimal FT configuration.

Step 7 On the primary load balancer, initiate a configuration sync with the standby load balancer.

Step 8 Verify that after the configuration sync completes, the FT status is in-sync.

Step 9 On the active load balancer, restore the original configuration from a flash disk to the startup configuration. Reset the aggregation switch.

Step 10 When the load balancer preempts to become active, initiate a configuration sync with the standby load balancer.

Step 11 Verify that after the configuration sync completes, the FT status is in-sync.

Expected Results

The following test results are anticipated:

- We expect the configuration to sync with the standby load balancer regardless of the size of the configuration.
- We expect that the load balancer will not crash or become unresponsive.

Results

Config Sync Large—CSM failed. The following failures were noted: CSCsu79370.

Configuration Sync—ACE

For redundancy to function properly, both members of an FT group must have identical configurations. To facilitate this requirement, the load balancer automatically replicates the active configuration on the standby member using a process called configuration synchronization (config sync). Config sync automatically replicates on the standby member any changes made to the configuration of the active member. After the load balancer syncs the redundancy configuration from the active member to the standby peer, it disables configuration mode on the standby.

This test verified that load balancer running and startup configs were synchronized when running in a redundant configuration. This was tested by adding and removing various parts of the config, while verifying the changes took place. This also tested the ability of a reloaded blade to learn the config from the active module.

Test Procedure

The procedure used to perform the Configuration Sync —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Configure a new context and make it fault tolerant.

Step 3 Verify that the primary and standby load balancer state is active/standby hot and that FT auto sync is enabled.

Step 4 Verify that no running config exists and create blank checkpoints in new context for the primary and standby load balancers.

Step 5 On the primary load balancer, change the configuration to allow direct management access to the new context.

Step 6 On the primary load balancer copy a scripted probe file to the CFG-SYNC1 context and import an SSL key and some certs.

Step 7 Verify that the copy of a scripted probe file and the import of an SSL key was successful.

Step 8 Copy and paste a config into the new context through a telnet session. There is a limitation of only being able to paste in about a 150 lines of config before overrunning the buffer, so the config will need to be pasted in sections. Verify that no errors were displayed during the configuration.

Step 9 On the primary load balancer, verify that the FT state for the peer is in an out-of-sync (COLD) state and that auto-sync disabled.

Step 10 On the standby module load the necessary probe script, force the config to sync, and verify that they are synchronized with an active/standby_hot state.

Step 11 Copy and paste another piece of the config into the context through a telnet session. Verify that no errors were displayed.

Step 12 Verify that the FT state for the peer is FSM_FT_STATE_STANDBY_COLD and that auto-sync is disabled for the running config.

Step 13 On the standby module load the necessary SSL key and certs, force the config to sync, and verify that they are synchronized with an active/standby_hot state.

Step 14 Through a console or terminal session, capture the running config of the new context from the primary and standby load balancer. Using a utility program, compare the configs and verify that they are correct.

Step 15 On the primary and standby load balancer, verify that ARP, interfaces, routes, static sticky table, and probes are up.

Step 16 Verify the top portion of the config on the standby and then remove various parts of the config from the primary. Verify this action by checking the running config on the standby load balancer.

Step 17 Verify the relevant portion of the config on the standby and then remove various parts of the config from the primary. Verify this action by checking the running config on the standby load balancer.

Step 18 Verify the relevant portion of the config on the standby and then remove various parts of the config from the primary. Verify this action by checking the running config on the standby load balancer.

Step 19 Verify that the arp entry and the routing table using this entry are on the standby load balancer.

Step 20 Remove a static ARP command from the primary load balancer. Verify on the standby load balancer that the static ARP entry is removed along with the static routes that were using this address as a next hop gateway.

Step 21 This will test the ability of the standby to load the config from the active load balancer. Verify on the standby load balancer that there is no startup config and then reload it.

Step 22 Once the standby comes back online, verify that the new context is in a standby hot state.

Step 23 Verify that the running config is in sync with the primary active load balancer.

Step 24 This will test the ability of the standby to receive the startup config from the active load balancer. Verify that the standby load balancer does not have a startup config and that the active load balancer shows FT auto sync startup config configured and enabled.

Step 25 On the primary load balancer save the running config to the startup config and force a replication of startup config to the standby load balancer.

Step 26 Verify on the standby load balancer that the startup config exits and is in sync with the active load balancer's running-config.

Step 27 On the active load balancer, clear the startup config. Verify that the startup config is cleared on both the active and standby load balancer's.

Step 28 On the active load balancer, execute a checkpoint rollback of a blank config. Verify that the running config is cleared on both the active and standby load balancer.

Step 29 These steps will test that deleting a context will not effect script files, SSL keys and certs in another context. Configure a new context and make it fault tolerant.

Step 30 On the active load balancer verify that the FT primary and peer is in an ACTIVE / standby hot state and that FT auto sync is enabled.

Step 31 On the active load balancer, configure the new context to allow direct management access to the new context.

Step 32 In the new context on both the active standby load balancer, copy a scripted probe file and import an SSL key and some certs.

Step 33 On both load balancers, verify that the file transfer and import was successful.

Step 34 In the admin context on the active load balancer, delete the new context.

Step 35 On the active load balancer, save the config and then reload the standby load balancer.

Step 36 Once the standby load balancer comes back online, verify in the new context on the active and standby load balancer that the script and SSL files are still present and accessible.

Step 37 On the active load balancer, delete the new context and save the config.

Expected Results

The following test results are anticipated:

- We expect the configuration to maintain synchronization with the standby load balancer when commands are dynamically added or removed.
- We expect that the load balancer to properly warn or copy over the required files if configuration parameters with dependencies (scripted probes, SSL keys, and certs) are not present.
- We expect deleting a context and then forcing a reload not to have an effect on files or SSL keys and certs in another context.
- We expect that the load balancer will not crash or become unresponsive.

Results

Configuration Sync —ACE passed with exception. The following exceptions were noted: CSCsj68643, CSCso77725, CSCsq63242.

Configuration Sync—CSM

For redundancy to function properly, both members of an FT group must have identical configurations. To facilitate this requirement, the load balancer automatically replicates the active configuration on the standby member using a process called configuration synchronization (config sync). Config sync automatically replicates on the standby member any changes made to the configuration of the active member. After the load balancer syncs the redundancy configuration from the active member to the standby peer, it disables configuration mode on the standby.

This test verified that load balancer running and startup configs were synchronized when running in a redundant configuration. This was tested by adding and removing various parts of the config, while verifying the changes took place. This also tested the ability of a reloaded blade to learn the config from the active module.

Test Procedure

The procedure used to perform the Configuration Sync—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that the configs between the active and standby load balancers are in sync by checking the FT status and by comparing the configs.

Step 3 The FT config has interface tracking configured with a unique GW and interface. Take note of the differences.

Step 4 The variable SASP_CSM_UNIQUE_ID is unique for the active and standby load balancers. Take note of the differences.

Step 5 On the primary load balancer, initiate a config sync with the standby load balancer.

Step 6 Verify that the config for the fault tolerant group and variable SASP_CSM_UNIQUE_ID is still unique and accurate on both the primary and standby load balancers.

Step 7 Return the FT priority on the standby load balancer back to its default. On the primary load balancer, initiate a config sync with the standby load balancer and verify that the FT priority was propagated to the standby load balancer.

Step 8 Make a series of changes to the config on the primary load balancer.

Step 9 On the primary load balancer, initiate a config sync with the standby load balancer.

Step 10 Verify with show commands that the changes were propagated to the standby load balancer.

Step 11 Make a series of changes to the config on the primary load balancer.

Step 12 On the primary load balancer, initiate a config sync with the standby load balancer.

Step 13 Verify with show commands that the changes were propagated to the standby load balancer.

Step 14 Make a series of changes to the config on the primary load balancer.

Step 15 On the primary load balancer, initiate a config sync with the standby load balancer.

Step 16 Verify with show commands that the changes were propagated to the standby load balancer.

Step 17 Verify that the configs between the active and standby load balancers are in sync by checking the FT status and by comparing the configs.

Step 18 Remove all of the Virtual IP (VIP) config from the standby load balancer.

Step 19 Clear the entire config from the the standby load balancer.

Step 20 Configure a basic FT config on the standby load balancer, and verify that it is in a standby state.

Step 21 On the primary load balancer, initiate a config sync with the standby load balancer.

Step 22 Verify that the configs between the active and standby load balancers are in sync by checking the FT status and by comparing the configs.

Step 23 Remove all of the added commands from the primary load balancer.

Step 24 On the primary load balancer, initiate a config sync with the standby load balancer.

Step 25 Verify that the configs between the active and standby load balancers are in sync by checking the FT status and by comparing the configs.

Expected Results

The following test results are anticipated:

- We expect the configuration to properly sync with the standby CSM.
- We expect that the load balancer will not crash or become unresponsive.

Results

Configuration Sync—CSM passed.

Probes

Health monitoring on the load balancer tracks the state of a server by sending out probes. Also referred as out-of-band health monitoring, the load balancer verifies the server response or checks for any network problems that can prevent a client to reach a server. Based on the server response, the load balancer can place the server in or out of service, and can make reliable load balancing decisions.

This section contains the following topics:

DNS probe—ACE

Health monitoring on the load balancer tracks the state of a server by sending out probes. The load balancer verifies the server response or checks for any network problems that can prevent a client to reach a server. Based on the server response, the load balancer can place the server in or out of service and can make reliable load balancing decisions. This test verified the functionality of the DNS probes in a number of situations that included differing number of valid IP responses, domain name case insensitivity, bad IP responses and no DNS server listening.

Relevant Load Balancer Configuration

```
probe dns PRB-DNS1
  description "all good addresses"
  interval 5
  passdetect interval 2
  domain www1.safeharbor.com
  expect address 1.1.1.1
  expect address 1.1.1.2
  expect address 1.1.1.3
probe dns PRB-DNS2
  description "one good address"
  interval 5
  passdetect interval 2
  domain www2.safeharbor.com
  expect address 2.1.1.1
probe dns PRB-DNS3
  description "2 good addresses, bumpy case"
  interval 5
  passdetect interval 2
  domain WwW3.SaFeHaRbOr.CoM
  expect address 3.1.1.3
  expect address 3.1.1.2
probe dns PRB-DNS4
  description "one good and one bad address"
  interval 5
  passdetect interval 2
  domain www4.safeharbor.com
  expect address 192.168.1.4
  expect address 4.1.1.1
probe dns PRB-DNS5
  description "all bad addresses"
  interval 5
  passdetect interval 2
  domain www4.safeharbor.com
  expect address 192.168.1.5
  expect address 192.168.1.6
  expect address 1.1.1.1
probe dns PRB-DNS6:2222
  description "dns not running on this port, but addresses good"
  port 2222
  interval 5
  passdetect interval 2
  domain www1.safeharbor.com
  expect address 1.1.1.1
  expect address 1.1.1.2
  expect address 1.1.1.3
serverfarm host PROBES-2
  predictor leastconns
  probe PRB-SSL:443
  rserver BRG-11
    inservice
```

```
rserver LOCAL-241
  inservice
rserver LOCAL-244
  inservice
rserver RT-152
  inservice
rserver RT-154
  inservice
```

Test Procedure

The procedure used to perform the DNS probe—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Add a DNS probe to a serverfarm.

Step 3 The probe is querying a domain name that will return three IP addresses from a DNS server. This probe is configured with all three valid IP addresses. Verify that the servers in the serverfarm are operational and that the probe executes without error.

Step 4 Add a DNS probe to a serverfarm.

Step 5 The probe is querying a domain name that will return three IP addresses from the DNS server. This probe is configured with one valid IP address. Verify that the servers in the serverfarm are operational and that the probe executes without error.

Step 6 Add a DNS probe to a serverfarm.

Step 7 The probe is querying a domain name in mixed case that will return three IP addresses from a DNS server. This probe is configured with two valid IP addresses. Verify that the servers in the serverfarm are operational and that the probes execute without error.

Step 8 Add a DNS probe to a serverfarm.

Step 9 The probe is querying a domain name that will return three IP addresses from the DNS server. This probe is configured with one valid and one bad IP address. Verify that the servers in this serverfarm are operational and that the probes execute without error.

Step 10 Add a DNS probe to a serverfarm.

Step 11 The probe is querying a domain name that will return three IP addresses from the DNS server. This probe is configured with three incorrect IP addresses. Verify that the servers in this serverfarm and the probe are FAILED.

Step 12 Remove a DNS probe from the serverfarm.

Step 13 Verify that the servers in the serverfarm become operational.

Step 14 Add a DNS probe to a serverfarm.

Step 15 The probe is querying a domain name that will return three IP addresses from the DNS server. This probe is configured with all three valid IP addresses, but is configured to use a UDP port that is not configured for DNS. Verify that the servers in this serverfarm and the probe are FAILED.

Step 16 Remove a DNS probe from the serverfarm.

Step 17 Verify that the servers in this serverfarm are operational and that all of the remaining probes execute without error.

Step 18 Remove the remaining probes from the serverfarm.

Expected Results

The following test results are anticipated:

- We expect the probe to be operational when at least one returned address matches the expected response configured.
- We expect the probe not to be case sensitive with the configured domain name.
- We expect the probe to fail if none of the addresses returned match the expected response.
- We expect the probe to fail if there is no DNS server running on that configured UDP port.
- We expect that the load balancer will not crash or become unresponsive.

Results

DNS probe—ACE passed.

DNS probe—CSM

Health monitoring on the load balancer tracks the state of a server by sending out probes. The load balancer verifies the server response or checks for any network problems that can prevent a client to reach a server. Based on the server response, the load balancer can place the server in or out of service and can make reliable load balancing decisions. This test verified that the load balancer fails when the DNS daemon is not running and succeeds when the DNS daemon is running.

Relevant Load Balancer Configuration

```
 probe TCP tcp
  interval 10
  retries 2
  failed 10
  open 3
  port 53
!
 serverfarm PROBES
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   inservice
  real name LOCAL-LINUX-244
   inservice
  real name RT-LINUX-151
   inservice
  probe ICMP
  probe HTTP-PROBE
  probe TELNET
  probe FTP
  probe TCP
!
```

Test Procedure

The procedure used to perform the DNS probe—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify the serverfarm configuration.

Step 3 Remove all probes but the TCP probe from the default configuration

Step 4 Verify that the TCP probe configuration monitors TCP port 53.

Step 5 Verify that the servers in the serverfarm are operational.

Step 6 On a Linux server, stop the DNS (named) daemon.

Step 7 Verify that the Linux server that is no longer running DNS (named) displays a PROBE_FAILED status.

Step 8 On a Linux server, start the DNS (named) daemon.

Step 9 Verify that the Linux server that had its DNS (named) daemon started displays an OPERATIONAL status.

Step 10 Return to the default serverfarm configuration.

Expected Results

The following test results are anticipated:

- We expect the probe to succeed when the DNS daemon is running on the server.
- We expect the probe to fail when the DNS daemon is not running on the server.
- We expect that the load balancer will not crash or become unresponsive.

Results

DNS probe—CSM passed.

HTTP Probes—ACE

Health monitoring on the load balancer tracks the state of a server by sending out probes. The load balancer verifies the server response based on configured constraints. Based on the server response, the load balancer can place the server in or out of service, and can make reliable load balancing decisions.

By default probes configured without a protocol port number use the default port. Probes will not inherit it from the serverfarm or vserver configuration. TCP based probes close the connection with a FIN and can be configured to use a RST. This test verified the functionality of the HTTP probe in various situations.

Relevant Load Balancer Configuration

```
 probe http PRB-HTTP:84
  port 84
  interval 5
  passdetect interval 4
  passdetect count 10
  expect status 200 200
probe http PRB-HTTP:85
```

```
  description Server RST 1byte data
  port 85
  interval 5
  passdetect interval 2
  expect status 200 200
  connection term forced
probe http PRB-HTTP:86
  description Server RST 3200byte data
  port 86
  interval 5
  passdetect interval 2
  expect status 200 200
  connection term forced
serverfarm host PROBES
  predictor leastconns
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-154
    inservice
```

Test Procedure

The procedure used to perform the HTTP Probes—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Add a couple of probes to a serverfarm. These probes have a mix of small and large amounts of data being returned. These will be checked later for failures.

Step 3 Add a probe to an server, which will cause it fail as it is requesting a page that does not exist.

Step 4 Verify that the server has failed.

Step 5 Add an expect status to a probe containing a value that will allow it to come alive.

Step 6 Verify that the probe for the server has become operational.

Step 7 Remove the probe from the server and change its expect status back to the original setting.

Step 8 Add a probe to an server.

Step 9 Apply an access list on the server access switch to block TCP port 84 traffic.

Step 10 Verify that the probe fails.

Step 11 Remove the access-list blocking the probe traffic.

Step 12 Verify that the probe becomes active.

Step 13 The probe has a receive timeout of 10, which means that it must get a response from a server within 10 seconds to consider the probe successful. Configure a server with a 12 seconds delay on TCP port 84.

Step 14 Verify that the probe has failed with a server reply timeout.

Step 15 Remove the delay from the server.

Step 16 Verify that the probe becomes active.

Step 17 Verify that there have been no failed states seen on the probes that were added when this test was first started.

Step 18 Remove the added probes and return the config back to its original state.

Expected Results

The following test results are anticipated:

- We expect the probe to fail when traffic to it is blocked.
- We expect the probe to fail when the server response takes longer than the configured wait time.
- We expect the probe to fail when the expected response code is not received.
- We expect that the load balancer will not crash or become unresponsive.

Results

HTTP Probes—ACE passed.

HTTP Probes—CSM

Health monitoring on the load balancer tracks the state of a server by sending out probes. The load balancer verifies the server response based on configured constraints. Based on the server response, the load balancer can place the server in or out of service, and can make reliable load balancing decisions.

Probes configured without a protocol port number will inherit it from the serverfarm or vserver configuration. TCP based probes close the connection with a reset and cannot be configured to use a FIN/ACK. If that is a requirement, scripted probes can be configured to close gracefully and substituted. This test verified the functionality of the HTTP probe in various situations.

Relevant Load Balancer Configuration

```
 !
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 probe HTTP-PROBE http
  interval 5
  failed 10
  open 3
  receive 5
  port 80
!
 serverfarm PROBES
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   inservice
  real name LOCAL-LINUX-244
```

```
    inservice
   real name RT-LINUX-151
    inservice
   probe TCP
   probe ICMP
   probe HTTP-PROBE
   probe TELNET
   probe FTP
!
```

Test Procedure

The procedure used to perform the HTTP Probes—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify the serverfarm configuration.

Step 3 Remove all probes but the HTTP probe from the default configuration.

Step 4 Verify that the servers in the serverfarm are operational.

Step 5 Block HTTP traffic to a server.

Step 6 Verify that the probe for the blocked server has failed.

Step 7 Unblock the HTTP traffic.

Step 8 Verify that the blocked server becomes operable again.

Step 9 Add the FORCED-FAILED probe to a serverfarm.

Step 10 Verify that the FORCED-FAIL probe has caused the servers in the serverfarm to fail.

Step 11 Change the FORCED-FAILED probe url to /index.html.

Step 12 Verify the FORCED-FAILED probe is operational.

Step 13 Change the FORCED-FAILED probe url back to /notthere.html.

Step 14 Verify the probe FORCED-FAIL fails after two attempts.

Step 15 Remove the FORCED-FAIL probe from the PROBES serverfarm.

Step 16 Return to the default serverfarm configuration.

Expected Results

The following test results are anticipated:

- We expect the probe to fail when traffic to it is blocked.
- We expect the probe to fail when the expected response code is not received.
- We expect that the load balancer will not crash or become unresponsive.

Results

HTTP Probes—CSM passed.

KALAP by Tag—ACE

The keepalive-appliance protocol (KAL-AP) on the load balancer allows communication between the load balancer and the Global Site Selector (GSS), which send KAL-AP requests, to report the server states and loads for global-server load-balancing (GSLB) decisions. The load balancer uses KAL-AP through a UDP connection to calculate weights and provide information for server availability. The load balancer supports VIP-based and TAG-based KAL-AP probes. For a Tag-based KAL-AP, when the load balancer receives a kal-ap-by-tag request, it verifies whether the tagged VIP address is active and returns a load value back to the GSS.

This test verified that the load balancer applies an appropriate load value based upon the current conditions and passes this information to the GSS through a KALAP by TAG response.

Relevant ACE Configuration

```
kalap udp
  ip address 10.1.0.214 encryption md5 Safe Harbor
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
parameter-map type connection 2SECOND-IDLE
  set timeout inactivity 2
parameter-map type http PERSIST-REBALANCE
  persistence-rebalance
serverfarm host GEN-443
  probe SSL
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-80
  predictor leastconns
  probe TCP
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-FTP
  probe FTP
  rserver BRG-13
```

```
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-UDP
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host GEN2-80
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-153
    inservice
sticky ip-netmask 255.255.255.255 address source STKY-GRP-30
  timeout 40
  replicate sticky
  serverfarm GEN-80
sticky http-cookie cookie-gold-grp40 STKY-GRP-40
  cookie insert browser-expire
  timeout 1
  replicate sticky
  serverfarm GEN2-80
class-map type management match-any MGT
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
  8 match protocol kalap-udp any
class-map match-all GEN-NAT_120
  2 match virtual-address 192.168.120.120 any
class-map match-all GEN-VIP_120:21
  2 match virtual-address 192.168.120.120 tcp eq ftp
class-map match-all GEN-VIP_120:443
  2 match virtual-address 192.168.120.120 tcp eq https
class-map match-all GEN-VIP_120:80
  2 match virtual-address 192.168.120.120 tcp eq www
class-map match-all GEN-VIP_120:UDP
  2 match virtual-address 192.168.120.120 udp any
policy-map type management first-match P-MGT
  class MGT
    permit
policy-map type loadbalance first-match PLBSF_GEN-443
  class class-default
    serverfarm GEN-443
policy-map type loadbalance first-match PLBSF_GEN-80
  class INDEX.HTML
    sticky-serverfarm STKY-GRP-40
  class class-default
    sticky-serverfarm STKY-GRP-30
policy-map type loadbalance first-match PLBSF_GEN-FTP
  class class-default
    sticky-serverfarm STKY-GRP-32
policy-map type loadbalance first-match PLBSF_GEN-UDP
```

```
  class class-default
    sticky-serverfarm STKY-GRP-31
policy-map multi-match SH-Gold-VIPs
  class GEN-NAT_120
    nat dynamic 1 vlan 120
  class GEN-VIP_120:80
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-80
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class GEN-VIP_120:443
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-443
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class GEN-VIP_120:21
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class GEN-VIP_120:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-UDP
    loadbalance vip icmp-reply active
    connection advanced-options 2SECOND-IDLE
service-policy input P-MGT
interface vlan 29
  ip address 172.29.0.2 255.255.255.0
  alias 172.29.0.1 255.255.255.0
  peer ip address 172.29.0.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.71 192.168.120.71 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  no shutdown
interface vlan 99
  ip address 192.168.99.2 255.255.255.0
  peer ip address 192.168.99.3 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 105
  ip address 192.168.105.2 255.255.255.0
  peer ip address 192.168.105.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
interface vlan 120
  description Upstream VLAN_120—Clients and VIPs
  ip address 192.168.120.2 255.255.255.0
  alias 192.168.120.1 255.255.255.0
  peer ip address 192.168.120.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.70 192.168.120.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  service-policy input NAT_POLICY
  no shutdown
domain KALAP_TAG
  add-object class-map GEN-VIP_120:21
```

Test Procedure

The procedure used to perform the KALAP by Tag—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary ACE in the SH-Gold context, clear the relevant counters by issuing the following commands:

```
clear service-policy SH-Gold-VIPs
clear serverfarm GEN-443
clear serverfarm GEN-80
clear serverfarm GEN2-80
clear stats all
```

Step 3 Verify on the ACE and through the GSS CLI that it is reporting a load of 2 for VIP 192.168.120.120.

Step 4 Take all of the rservers in serverfarm GEN2-80 out-of-service. Issue the following commands:

```
config
serverfarm host GEN2-80
  rserver BRG-11
   no inservice
  rserver LOCAL-241
   no inservice
  rserver RT-153
   no inservice
end
```

Step 5 Since this policy has two L7 class-maps and serverfarms, if one of these goes down, it does not bring down the VIP. Verify that all of the rservers are OUTOFSERVICE and that the policy shows the serverfarm OUTOFSERVICE, but the VIP is still INSERVICE. Issue the following commands:

```
show serverfarm GEN2-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 6 Verify on the ACE and through the GSS CLI that it is reporting a load of 37 for VIP 192.168.120.120.

Step 7 Take all of the rservers in serverfarm GEN-80 out-of-service. Issue the following commands:

```
config
serverfarm host GEN-80
  rserver BRG-12
   no inservice
  rserver BRG-13
   no inservice
  rserver LOCAL-244
   no inservice
  rserver LOCAL-245
   no inservice
  rserver RT-151
   no inservice
  rserver RT-152
```

```
   no inservice
end
```

Step 8 This is the last serverfarm remaining, so the VIP will go down. Since this is kalap by vip if one class-map sharing an IP address (VIP) with others goes down, then the VIP is considered down and a load of 255 will be reported. Verify that all of the rservers are OUTOFSERVICE and that the policy shows both serverfarms and VIP as OUTOFSERVICE. Issue the following commands:

```
show serverfarm GEN-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 9 Verify on the ACE and through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.120.

Step 10 Bring both serverfarms back online and verify that the ACE reports VIP 192.168.120.120 with a load of two. Issue the following commands:

```
config
serverfarm host GEN2-80
  rserver BRG-11
   inservice
  rserver LOCAL-241
   inservice
  rserver RT-153
   inservice
serverfarm host GEN-80
  rserver BRG-12
   inservice
  rserver BRG-13
   inservice
  rserver LOCAL-244
   inservice
  rserver LOCAL-245
   inservice
  rserver RT-151
   inservice
  rserver RT-152
   inservice
end
show kalap udp load vip 192.168.120.120
```

Step 11 Force all of the rservers in serverfarm GEN2-80 down by adding a probe. Issue the following commands:

```
config
serverfarm host GEN2-80
probe FORCED-FAIL
end
```

Step 12 Since this policy has two L7 class-maps and serverfarms, if one of these goes down, it does not bring down the VIP. Verify that all of the rservers are PROBEFAILED and that the policy shows the serverfarm OUTOFSERVICE, but the VIP is still INSERVICE. Issue the following commands:

```
show serverfarm GEN2-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 13 Verify on the ACE and through the GSS CLI that it is reporting a load of 37 for VIP 192.168.120.120.

Step 14 Force all of the rservers in serverfarm GEN-80 down by adding a probe. Issue the following commands:

```
config
serverfarm host GEN-80
probe FORCED-FAIL
end
```

Step 15 Since this is the last serverfarm remaining the VIP will go down. Since this is kalap by vip if one class-map sharing an IP address (VIP) with others goes down, then the VIP is considered down and a load of 255 will be reported. Verify that all of the rservers are PROBEFAILED and that the policy shows both serverfarms and VIP as OUTOFSERVICE. Issue the following commands:

```
show serverfarm GEN-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 16 Verify on the ACE and through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.120.

Step 17 Bring both serverfarms back online and verify that the ACE reports VIP 192.168.120.120 with a load of two. Issue the following commands:

```
config
serverfarm host GEN2-80
  no probe FORCED-FAIL
serverfarm host GEN-80
  no probe FORCED-FAIL
end
show kalap udp load vip 192.168.120.120
```

Step 18 Configure one rserver in serverfarm GEN-443 for maxconn and take the rest out-of-service. Issue the following commands:

```
config
serverfarm host GEN-443
 rserver BRG-12
   no inservice
 rserver BRG-13
   conn-limit max 4 min 2
 rserver LOCAL-244
   no inservice
 rserver LOCAL-245
   no inservice
 rserver RT-151
   no inservice
 rserver RT-152
   no inservice
end
```

Step 19 Since one rserver remains up the serverfarm and VIP will remain inservice. Verify that only one rserver is up and that the policy shows the serverfarm and VIP INSERVICE. Issue the following commands:

```
show serverfarm GEN-443
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:443
```

Step 20 Verify on the ACE and through the GSS CLI that it is reporting a load of 61 for VIP 192.168.120.120.

Step 21 From a Linux client launch a series of requests that will force the remaining rserver into a maxconn state.

Step 22 The traffic launched will put the only rserver available in the GEN-443 serverfarm in a MAXCONN state. Since this particular class-map only has one serverfarm configured this will force the ACE to advertise the VIP as fully loaded. This condition is from a MAXCONN condition, so only the serverfarm will go down and not the VIP. Since this is kalap by vip if one class-map sharing an IP address (VIP) with others goes down, then the VIP is considered down and a load of 255 will be reported. Verify that the serverfarm is OUTOFSERVICE, but the VIP remains INSERVICE. Issue the following commands:

```
show serverfarm GEN-443
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:443
```

Step 23 Even though the VIP is inservice the serverfarm is fully loaded, so the load reported will be 255. Verify on the ACE and through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.120.

Step 24 Stop the client traffic and return the config back to its original state. Issue the following commands:

```
config
serverfarm host GEN-443
 rserver BRG-12
   inservice
 rserver BRG-13
   no conn-limit
 rserver LOCAL-244
   inservice
 rserver LOCAL-245
   inservice
 rserver RT-151
   inservice
 rserver RT-152
   inservice
end
```

Expected Results

The following test results are anticipated:

- We expect the load reported to be inline with the number of rservers operational for a given VIP associated with a domain tag.
- We expect the load reported to be inline with the number of rservers operational for a given VIP if at least one serverfarm is operational for each class-map within a shared (IP address) VIP associated with a domain tag.

- We expect the load reported to be 255 (fully loaded) if a shared class-map vip has only one serverfarm configured and it is considered down.
- We expect the load reported to be 255 (fully loaded) if all serverfarms are considered down in a particular shared class-map vip associated with a domain tag.
- We expect no CPU or memory problems.

Results

KALAP by Tag—ACE passed with exception. The following exceptions were noted: CSCsu54970 and CSCsu55144.

KALAP by Tag—CSM

The keepalive-appliance protocol (KAL-AP) on the load balancer allows communication between the load balancer and the Global Site Selector (GSS), which send KAL-AP requests, to report the server states and loads for global-server load-balancing (GSLB) decisions. The load balancer uses KAL-AP through a UDP connection to calculate weights and provide information for server availability. The load balancer supports VIP-based and TAG-based KAL-AP probes. For a Tag-based KAL-AP, when the load balancer receives a kal-ap-by-tag request, it verifies whether the tagged VIP address is active and returns a load value back to the GSS.

This test verified that the load balancer applies an appropriate load value based upon the current conditions and passes this information to the GSS through a KALAP by TAG response.

Relevant CSM Configuration

```
module ContentSwitchingModule 8
 variable GSLB_LICENSE_KEY Q1NDT0dTTEJCRU9XVUxG
 variable INFINITE_IDLE_TIME_MAXCONNS 1
 variable REAL_SLOW_START_ENABLE 3
 variable SASP_CSM_UNIQUE_ID abcdefghijklmnopqrstuvwxy_01234567890-BCDEFGHIJKL
NOPQRSTUVWXYZ
 variable SASP_FIRST_BIND_ID 1000
 variable SASP_GWM_BIND_ID_MAX 8
!
 ft group 2 vlan 900
  priority 110 alt 100
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.251
  track interface GigabitEthernet4/37
  track mode any
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 105 client
  ip address 192.168.105.12 255.255.255.0 alt 192.168.105.13 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
```

```
  ip address 10.86.83.13 255.255.255.0 alt 10.86.83.14 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 map BROWSER_MSIE header
  match protocol http header User-Agent header-value *MSIE*
!
 serverfarm CS-MSIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm GEN
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
  probe ICMP
!
 serverfarm GEN-443
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-IIS-152
   inservice
  probe SSLPROBE
!
 serverfarm GEN-80
  nat server
  nat client CLIENT_NAT
  real name RT-IIS-152
   inservice
  real name BRG-LINUX-12
   inservice
  real name LOCAL-IIS-241
   inservice
  probe TCP-GENERIC
!
 serverfarm GEN-UDP
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-154
   inservice
  real name BRG-LINUX-13
   inservice
  probe ICMP
!
 sticky 33 netmask 255.255.255.255 address source timeout 1
!
 sticky 113 ssl timeout 1
!
 policy BROWSER_MSIE
  header-map BROWSER_MSIE
  serverfarm CS-MSIE
!
 vserver GEN
  virtual 192.168.120.200 any
  serverfarm GEN
  idle 4
  persistent rebalance
  inservice
!
 vserver GEN-443
  virtual 192.168.120.200 tcp https
  serverfarm GEN-443
  sticky 1 group 113
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-80
  virtual 192.168.120.200 tcp www
  serverfarm GEN-80
  sticky 1 group 33
  domain kalaptag
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-UDP
  virtual 192.168.120.200 udp 0
  serverfarm GEN-UDP
  idle 2
  replicate csrp connection
  persistent rebalance
  inservice
!
 capp udp
  secure
  options 10.1.0.214 encryption md5 Safe Harbor
!
```

Test Procedure

The procedure used to perform the KALAP by Tag—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary CSM, clear the relevant counters by issuing the **clear mod csm 8 count** command.

Step 3 Configure a second policy for vserver GEN-80. Issue the following commands:

```
config t
mod csm 8
vserver GEN-80
slb-policy BROWSER_MSIE
end
```

Step 4 Verify through the GSS CLI that it is reporting a load of 2 for VIP 192.168.120.200.

Step 5 Take all of the real servers in serverfarm GEN-80 out-of-service. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
real name RT-IIS-152
no inservice
real name BRG-LINUX-12
no inservice
real name LOCAL-IIS-241
no inservice
end
```

Step 6 Since this vserver has two policies configured if one goes down, it does not bring down the vserver. Verify that all of the real servers are OUTOFSERVICE in serverfarm GEN-80 and that the vserver is still OPERATIONAL. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
```

Step 7 Take all of the real servers in serverfarm CS-MSIE out-of-service. Issue the following commands:

```
config t
mod csm 8
 serverfarm CS-MSIE
  real name LOCAL-IIS-243
   no inservice
  real name RT-LINUX-151
   no inservice
end
```

Step 8 This is the last serverfarm remaining, so the VIP will go down. Verify that all of the real servers are OUTOFSERVICE in both serverfarms and that the vserver is OUTOFSERVICE. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 serverfarms name cs-msie detail
show mod csm 8 vserver name gen-80 detail
```

Step 9 Since this is kalap-by-tag only the tagged vserver is checked for availability and load, even if sharing an IP address (VIP) with others vservers. Verify through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.200.

Step 10 Bring the serverfarms back online and verify that the GSS reports VIP 192.168.120.200 with a load of two. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
real name RT-IIS-152
inservice
real name BRG-LINUX-12
inservice
real name LOCAL-IIS-241
inservice
serverfarm CS-MSIE
real name LOCAL-IIS-243
inservice
real name RT-LINUX-151
inservice
end
```

Step 11 Remove the policy that was added to vserver GEN-80. Issue the following commands:

```
config t
mod csm 8
vserver GEN-80
no slb-policy BROWSER_MSIE
end
```

Step 12 Force all of the rservers in serverfarm GEN-80 down by adding a probe. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
probe FORCED-FAIL
end
```

Step 13 Verify that all of the real servers are PROBE_FAILED and the vserver is OUTOFSERVICE. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
```

Step 14 Verify through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.200.

Step 15 Bring the serverfarm back online and verify that the GSS reports VIP 192.168.120.200 with a load of two. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
no probe FORCED-FAIL
end
```

Step 16 Configure one real server in serverfarm GEN-80 for maxconn and take the rest out-of-service. Issue the following commands:

```
config t
mod csm 8
server GEN-80
real name RT-IIS-152
 no inservice
real name BRG-LINUX-12
 maxconn 4
real name LOCAL-IIS-241
 no inservice
end
```

Step 17 Since one real server remains up the serverfarm and VIP will remain inservice. Verify that only one real server is up and that the VIP is INSERVICE. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
```

Step 18 Verify through the GSS CLI that it is reporting a load of 2 for VIP 192.168.120.200.

Step 19 From a Linux client launch a series of requests that will force the remaining rserver into a maxconn state.

Step 20 The traffic launched will put the only rserver available in the GEN-80 serverfarm in a MAXCONN state. This makes the serverfarm unavailable, so the CSM will advertise the VIP as fully loaded. Verify that the serverfarm is OUTOFSERVICE. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
```

Step 21 Verify through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.200.

Step 22 Stop the client traffic and return the config back to its original state. Issue the following commands:

```
config t
mod csm 8
server GEN-80
real name RT-IIS-152
 inservice
real name BRG-LINUX-12
 no maxconn
real name LOCAL-IIS-241
 inservice
end
```

Expected Results

The following test results are anticipated:

- We expect an operational load to be reported if at least one serverfarm is operational within a vserver configuration
- We expect the load reported to be 255 (fully loaded) if a single shared vserver vip is considered down.
- We expect the load reported to be 255 (fully loaded) if all serverfarms are considered down in a particular shared vserver vip.

Results

KALAP by Tag—CSM passed.

KALAP by VIP—ACE

The keepalive-appliance protocol (KAL-AP) on the load balancer allows communication between the load balancer and the Global Site Selector (GSS), which send KAL-AP requests, to report the server states and loads for global-server load-balancing (GSLB) decisions. The load balancer uses KAL-AP through a UDP connection to calculate weights and provide information for server availability. The load balancer supports VIP-based and TAG-based KAL-AP probes. For a VIP-based KAL-AP, when the load balancer receives a kal-ap-by-vip request, it verifies whether the tagged VIP address is active and returns a load value back to the GSS.

This test verified that the load balancer applies an appropriate load value based upon the current conditions and passes this information to the GSS through a KALAP by VIP response.

Relevant ACE Configuration

```
kalap udp
  ip address 10.1.0.214 encryption md5 Safe Harbor
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
parameter-map type connection 2SECOND-IDLE
  set timeout inactivity 2
parameter-map type http PERSIST-REBALANCE
  persistence-rebalance
serverfarm host GEN-443
  probe SSL
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
```

```
serverfarm host GEN-80
  predictor leastconns
  probe TCP
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-FTP
  probe FTP
  rserver BRG-13
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-UDP
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host GEN2-80
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-153
    inservice
sticky ip-netmask 255.255.255.255 address source STKY-GRP-30
  timeout 40
  replicate sticky
  serverfarm GEN-80
sticky http-cookie cookie-gold-grp40 STKY-GRP-40
  cookie insert browser-expire
  timeout 1
  replicate sticky
  serverfarm GEN2-80
class-map type management match-any MGT
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
  8 match protocol kalap-udp any
class-map match-all GEN-NAT_120
  2 match virtual-address 192.168.120.120 any
class-map match-all GEN-VIP_120:21
  2 match virtual-address 192.168.120.120 tcp eq ftp
class-map match-all GEN-VIP_120:443
  2 match virtual-address 192.168.120.120 tcp eq https
class-map match-all GEN-VIP_120:80
```

```
  2 match virtual-address 192.168.120.120 tcp eq www
class-map match-all GEN-VIP_120:UDP
  2 match virtual-address 192.168.120.120 udp any
policy-map type management first-match P-MGT
  class MGT
    permit
policy-map type loadbalance first-match PLBSF_GEN-443
  class class-default
    serverfarm GEN-443
policy-map type loadbalance first-match PLBSF_GEN-80
  class INDEX.HTML
    sticky-serverfarm STKY-GRP-40
  class class-default
    sticky-serverfarm STKY-GRP-30
policy-map type loadbalance first-match PLBSF_GEN-FTP
  class class-default
    sticky-serverfarm STKY-GRP-32
policy-map type loadbalance first-match PLBSF_GEN-UDP
  class class-default
    sticky-serverfarm STKY-GRP-31
policy-map multi-match SH-Gold-VIPs
  class GEN-NAT_120
    nat dynamic 1 vlan 120
  class GEN-VIP_120:80
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-80
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class GEN-VIP_120:443
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-443
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class GEN-VIP_120:21
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class GEN-VIP_120:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-UDP
    loadbalance vip icmp-reply active
    connection advanced-options 2SECOND-IDLE
service-policy input P-MGT
interface vlan 29
  ip address 172.29.0.2 255.255.255.0
  alias 172.29.0.1 255.255.255.0
  peer ip address 172.29.0.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.71 192.168.120.71 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  no shutdown
interface vlan 99
  ip address 192.168.99.2 255.255.255.0
  peer ip address 192.168.99.3 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 105
  ip address 192.168.105.2 255.255.255.0
  peer ip address 192.168.105.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
interface vlan 120
  description Upstream VLAN_120—Clients and VIPs
  ip address 192.168.120.2 255.255.255.0
  alias 192.168.120.1 255.255.255.0
  peer ip address 192.168.120.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.70 192.168.120.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  service-policy input NAT_POLICY
  no shutdown
domain KALAP_TAG
  add-object class-map GEN-VIP_120:21
```

Test Procedure

The procedure used to perform the KALAP by VIP—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary ACE in the SH-Gold context, clear the relevant counters by issuing the following commands:

```
clear service-policy SH-Gold-VIPs
clear serverfarm GEN-443
clear serverfarm GEN-80
clear serverfarm GEN2-80
clear stats all
```

Step 3 Verify on the ACE and through the GSS CLI that it is reporting a load of 2 for VIP 192.168.120.120.

Step 4 Take all of the rservers in serverfarm GEN2-80 out-of-service. Issue the following commands:

```
config
serverfarm host GEN2-80
  rserver BRG-11
   no inservice
  rserver LOCAL-241
   no inservice
  rserver RT-153
   no inservice
end
```

Step 5 Since this policy has two L7 class-maps and serverfarms, if one of these goes down, it does not bring down the VIP. Verify that all of the rservers are OUTOFSERVICE and that the policy shows the serverfarm OUTOFSERVICE, but the VIP is still INSERVICE. Issue the following commands:

```
show serverfarm GEN2-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 6 Verify on the ACE and through the GSS CLI that it is reporting a load of 37 for VIP 192.168.120.120.

Step 7 Take all of the rservers in serverfarm GEN-80 out-of-service. Issue the following commands:

```
config
serverfarm host GEN-80
  rserver BRG-12
   no inservice
  rserver BRG-13
   no inservice
  rserver LOCAL-244
   no inservice
  rserver LOCAL-245
   no inservice
  rserver RT-151
   no inservice
  rserver RT-152
   no inservice
end
```

Step 8 This is the last serverfarm remaining, so the VIP will go down. Since this is kalap-by-vip if one class-map sharing an IP address (VIP) with others goes down, then the VIP is considered down and a load of 255 will be reported. Verify that all of the rservers are OUTOFSERVICE and that the policy shows both serverfarms and VIP as OUTOFSERVICE. Issue the following commands:

```
show serverfarm GEN-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 9 Verify on the ACE and through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.120.

Step 10 Bring both serverfarms back online and verify that the ACE reports VIP 192.168.120.120 with a load of two. Issue the following commands:

```
config
serverfarm host GEN2-80
  rserver BRG-11
   inservice
  rserver LOCAL-241
   inservice
  rserver RT-153
   inservice
serverfarm host GEN-80
  rserver BRG-12
   inservice
  rserver BRG-13
   inservice
  rserver LOCAL-244
   inservice
  rserver LOCAL-245
   inservice
  rserver RT-151
   inservice
  rserver RT-152
   inservice
end
```

```
show kalap udp load vip 192.168.120.120
```

Step 11 Force all of the rservers in serverfarm GEN2-80 down by adding a probe. Issue the following commands:

```
config
serverfarm host GEN2-80
probe FORCED-FAIL
end
```

Step 12 Since this policy has two L7 class-maps and serverfarms, if one of these goes down, it does not bring down the VIP. Verify that all of the rservers are PROBEFAILED and that the policy shows the serverfarm OUTOFSERVICE, but the VIP is still INSERVICE. Issue the following commands:

```
show serverfarm GEN2-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 13 Verify on the ACE and through the GSS CLI that it is reporting a load of 37 for VIP 192.168.120.120.

Step 14 Force all of the rservers in serverfarm GEN-80 down by adding a probe. Issue the following commands:

```
config
serverfarm host GEN-80
probe FORCED-FAIL
end
```

Step 15 Since this is the last serverfarm remaining the VIP will go down. Since this is kalap by vip if one class-map sharing an IP address (VIP) with others goes down, then the VIP is considered down and a load of 255 will be reported. Verify that all of the rservers are PROBEFAILED and that the policy shows both serverfarms and VIP as OUTOFSERVICE. Issue the following commands:

```
show serverfarm GEN-80
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:80
```

Step 16 Verify on the ACE and through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.120.

Step 17 Bring both serverfarms back online and verify that the ACE reports VIP 192.168.120.120 with a load of two. Issue the following commands:

```
config
serverfarm host GEN2-80
  no probe FORCED-FAIL
serverfarm host GEN-80
  no probe FORCED-FAIL
end
show kalap udp load vip 192.168.120.120
```

Step 18 Configure one rserver in serverfarm GEN-443 for maxconn and take the rest out-of-service. Issue the following commands:

```
config
serverfarm host GEN-443
 rserver BRG-12
   no inservice
 rserver BRG-13
   conn-limit max 4 min 2
 rserver LOCAL-244
   no inservice
 rserver LOCAL-245
   no inservice
 rserver RT-151
   no inservice
 rserver RT-152
   no inservice
end
```

Step 19 Since one rserver remains up the serverfarm and VIP will remain inservice. Verify that only one rserver is up and that the policy shows the serverfarm and VIP INSERVICE. Issue the following commands:

```
show serverfarm GEN-443
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:443
```

Step 20 Verify on the ACE and through the GSS CLI that it is reporting a load of 61 for VIP 192.168.120.120.

Step 21 From a Linux client launch a series of requests that will force the remaining rserver into a maxconn state.

Step 22 The traffic launched will put the only rserver available in the GEN-443 serverfarm in a MAXCONN state. Since this particular class-map only has one serverfarm configured this will force the ACE to advertise the VIP as fully loaded. This condition is from a MAXCONN condition, so only the serverfarm will go down and not the VIP. Since this is kalap-by-vip if one class-map sharing an IP address (VIP) with others is considered down (maxconn), then the VIP is considered down and a load of 255 will be reported. Verify that the serverfarm is OUTOFSERVICE, but the VIP remains INSERVICE. Issue the following commands:

```
show serverfarm GEN-443
show service-policy SH-Gold-VIPs detail | beg GEN-VIP_120:443
```

Step 23 Even though the VIP is inservice the serverfarm is fully loaded, so the load reported will be 255. Verify on the ACE and through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.120.

Step 24 Stop the client traffic and return the config back to its original state. Issue the following commands:

```
config
serverfarm host GEN-443
 rserver BRG-12
   inservice
 rserver BRG-13
   no conn-limit
 rserver LOCAL-244
   inservice
 rserver LOCAL-245
   inservice
 rserver RT-151
```

```
  inservice
 rserver RT-152
  inservice
end
```

Expected Results

The following test results are anticipated:

- We expect the load reported to be inline with the number of rservers operational for a given VIP.
- We expect the load reported to be inline with the number of rservers operational for a given VIP if at least one serverfarm is operational for each class-map within a shared (IP address) VIP.
- We expect the load reported to be 255 (fully loaded) if a shared class-map vip has only one serverfarm configured and it is considered down.
- We expect the load reported to be 255 (fully loaded) if all serverfarms are considered down in a particular shared class-map vip.
- We expect no CPU or memory problems.

Results

KALAP by VIP—ACE passed with exception. The following exceptions were noted: CSCsu54970 and CSCsu55144.

KALAL by VIP—CSM

The keepalive-appliance protocol (KAL-AP) on the load balancer allows communication between the load balancer and the Global Site Selector (GSS), which send KAL-AP requests, to report the server states and loads for global-server load-balancing (GSLB) decisions. The load balancer uses KAL-AP through a UDP connection to calculate weights and provide information for server availability. The load balancer supports VIP-based and TAG-based KAL-AP probes. For a VIP-based KAL-AP, when the load balancer receives a kal-ap-by-vip request, it verifies whether the tagged VIP address is active and returns a load value back to the GSS.

This test verified that the load balancer applies an appropriate load value based upon the current conditions and passes this information to the GSS through a KALAP by VIP response.

Relevant CSM Configuration

```
module ContentSwitchingModule 8
 variable GSLB_LICENSE_KEY Q1NDT0dTTEJCRU9XVUxG
 variable INFINITE_IDLE_TIME_MAXCONNS 1
 variable REAL_SLOW_START_ENABLE 3
 variable SASP_CSM_UNIQUE_ID abcdefghijklmnopqrstuvwxy_01234567890-BCDEFGHIJKL
NOPQRSTUVWXYZ
 variable SASP_FIRST_BIND_ID 1000
 variable SASP_GWM_BIND_ID_MAX 8
!
 ft group 2 vlan 900
  priority 110 alt 100
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.251
  track interface GigabitEthernet4/37
  track mode any
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 105 client
  ip address 192.168.105.12 255.255.255.0 alt 192.168.105.13 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
  ip address 10.86.83.13 255.255.255.0 alt 10.86.83.14 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 map BROWSER_MSIE header
  match protocol http header User-Agent header-value *MSIE*
!
 serverfarm CS-MSIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm GEN
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
  probe ICMP
!
 serverfarm GEN-443
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-IIS-152
   inservice
  probe SSLPROBE
```

```
!
 serverfarm GEN-80
  nat server
  nat client CLIENT_NAT
  real name RT-IIS-152
   inservice
  real name BRG-LINUX-12
   inservice
  real name LOCAL-IIS-241
   inservice
  probe TCP-GENERIC
!
 serverfarm GEN-UDP
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-154
   inservice
  real name BRG-LINUX-13
   inservice
  probe ICMP
!
 sticky 33 netmask 255.255.255.255 address source timeout 1
!
 sticky 113 ssl timeout 1
!
 policy BROWSER_MSIE
  header-map BROWSER_MSIE
  serverfarm CS-MSIE
!
 vserver GEN
  virtual 192.168.120.200 any
  serverfarm GEN
  idle 4
  persistent rebalance
  inservice
!
 vserver GEN-443
  virtual 192.168.120.200 tcp https
  serverfarm GEN-443
  sticky 1 group 113
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-80
  virtual 192.168.120.200 tcp www
  serverfarm GEN-80
  sticky 1 group 33
  domain kalaptag.csm.com
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-UDP
  virtual 192.168.120.200 udp 0
  serverfarm GEN-UDP
  idle 2
  replicate csrp connection
  persistent rebalance
  inservice
!
 capp udp
```

```
  secure
  options 10.1.0.214 encryption md5 Safe Harbor
!
```

Test Procedure

The procedure used to perform the KALAL by VIP—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary CSM, clear the relevant counters by issuing the **clear mod csm 8 count** command.

Step 3 Configure a second policy for vserver GEN-80. Issue the following commands:

```
config t
mod csm 8
vserver GEN-80
slb-policy BROWSER_MSIE
end
```

Step 4 Verify through the GSS CLI that it is reporting a load of 2 for VIP 192.168.120.200.

Step 5 Take all of the real servers in serverfarm GEN-80 out-of-service. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
real name RT-IIS-152
no inservice
real name BRG-LINUX-12
no inservice
real name LOCAL-IIS-241
no inservice
end
```

Step 6 Since this vserver has two policies configured if one goes down, it does not bring down the vserver. Verify that all of the real servers are OUTOFSERVICE in serverfarm GEN-80 and that the vserver is still OPERATIONAL. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
```

Step 7 Take all of the real servers in serverfarm CS-MSIE out-of-service. Issue the following commands:

```
config t
mod csm 8
 serverfarm CS-MSIE
  real name LOCAL-IIS-243
   no inservice
  real name RT-LINUX-151
   no inservice
end
```

Step 8 This is the last serverfarm remaining, so the VIP will go down. Verify that all of the real servers are OUTOFSERVICE in both serverfarms and that the vserver is OUTOFSERVICE. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 serverfarms name cs-msie detail
show mod csm 8 vserver name gen-80 detail
show mod csm 8 vserver name gen detail
show mod csm 8 vserver name gen-443 detail
show mod csm 8 vserver name gen-udp detail
```

Step 9 Since this is kalap by vip if one vserver sharing an IP address (VIP) with others goes down, the VIP is considered down and a load of 255 will be reported. Verify through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.200.

Step 10 Bring the serverfarms back online and verify that the GSS reports VIP 192.168.120.200 with a load of two. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
real name RT-IIS-152
inservice
real name BRG-LINUX-12
inservice
real name LOCAL-IIS-241
inservice
serverfarm CS-MSIE
real name LOCAL-IIS-243
inservice
real name RT-LINUX-151
inservice
end
```

Step 11 Remove the policy that was added to vserver GEN-80. Issue the following commands:

```
config t
mod csm 8
vserver GEN-80
no slb-policy BROWSER_MSIE
end
```

Step 12 Force all of the rservers in serverfarm GEN-80 down by adding a probe. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
probe FORCED-FAIL
end
```

Step 13 Verify that all of the real servers are PROBE_FAILED and that only vserver GEN-80 is OUTOFSERVICE. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
show mod csm 8 vserver name gen detail
show mod csm 8 vserver name gen-443 detail
show mod csm 8 vserver name gen-udp detail
```

Step 14 Verify through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.200.

Step 15 Bring the serverfarm back online and verify that the GSS reports VIP 192.168.120.200 with a load of two. Issue the following commands:

```
config t
mod csm 8
serverfarm GEN-80
no probe FORCED-FAIL
end
```

Step 16 Configure one real server in serverfarm GEN-80 for maxconn and take the rest out-of-service. Issue the following commands:

```
config t
mod csm 8
server GEN-80
real name RT-IIS-152
 no inservice
real name BRG-LINUX-12
 maxconn 4
real name LOCAL-IIS-241
 no inservice
end
```

Step 17 Since one real server remains up the serverfarm and VIP will remain inservice. Verify that only one real server is up and that the VIP is OPERATIONAL. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
show mod csm 8 vserver name gen detail
show mod csm 8 vserver name gen-443 detail
show mod csm 8 vserver name gen-udp detail
```

Step 18 Verify through the GSS CLI that it is reporting a load of 2 for VIP 192.168.120.200.

Step 19 From a Linux client launch a series of requests that will force the remaining rserver into a maxconn state.

Step 20 The traffic launched will put the only rserver available in the GEN-80 serverfarm in a MAXCONN state. This makes the serverfarm unavailable, so the CSM will advertise the VIP as fully loaded. Verify that the serverfarm has no real servers OPERATIONAL. Issue the following commands:

```
show mod csm 8 serverfarms name gen-80 detail
show mod csm 8 vserver name gen-80 detail
show mod csm 8 vserver name gen detail
show mod csm 8 vserver name gen-443 detail
show mod csm 8 vserver name gen-udp detail
```

Step 21 Verify through the GSS CLI that it is reporting a load of 255 for VIP 192.168.120.200.

Step 22 Stop the client traffic and return the config back to its original state. Issue the following commands:

```
config t
mod csm 8
server GEN-80
real name RT-IIS-152
 inservice
real name BRG-LINUX-12
 no maxconn
real name LOCAL-IIS-241
 inservice
end
```

Expected Results

The following test results are anticipated:

- We expect an operational load to be reported if at least one serverfarm is operational within a vserver configuration
- We expect the load reported to be 255 (fully loaded) if a single shared vserver vip is considered down.
- We expect the load reported to be 255 (fully loaded) if all serverfarms are considered down in a particular shared vserver vip.

Results

KALAL by VIP—CSM failed. The following failures were noted: CSCsj26410.

SSL Probe—ACE

Health monitoring on the load balancer tracks the state of a server by sending out probes. The load balancer verifies the server response or checks for any network problems that can prevent a client to reach a server. Based on the server response, the load balancer can place the server in or out of service, and can make reliable load balancing decisions. This test verified the functionality of SSL probes in a number of successful and failure situations.

Relevant Load Balancer Configuration

```
probe https PRB-SSL:443
  interval 5
  passdetect interval 10
  request method get url /index.txt
  expect status 200 200
  header Via header-value "PRB-SSL:443 Probe Header"
  hash
serverfarm host PROBES-2
  predictor leastconns
  probe PRB-SSL:443
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
  rserver RT-154
    inservice
```

Test Procedure

The procedure used to perform the SSL Probe—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary load balancer, remove the SSL probe from select serverfarms. This will make any packet captures taken easier to read.

Step 3 Add an SSL probe to a serverfarm.

Step 4 Verify that the servers in this serverfarm are operational and that the probes execute without error.

Step 5 Start a packet capture on the NAM to capture the SSL probe traffic.

Step 6 Change the SSL probe to force it to use SSL version TLS 1.0 and a single cipher.

Step 7 Verify that the SSL probe is operational and that the probes execute without error, except for on server. This failing server does not support TLS 1.0.

Step 8 Stop the packet capture and verify that the load balancer switches to sending client hello packets using SSL version TLS 1.0 with only cipher RSA_WITH_3DES_EDE_CBC_SHA included.

Step 9 On the SSL probe remove the SSL version and cipher.

Step 10 Start a packet capture on the NAM to capture the SSL probe traffic.

Step 11 The load balancer calculates a hash on the first successful GET request from the probe and checks this value each time after that. On a server modify the index.txt to alter the hash value that would be generated.

Step 12 Verify that the probe fails for the server whose index.txt was modified.

Step 13 Change the SSL probe to issue a head request for a large file.

Step 14 Verify that the probe is successful for all servers.

Step 15 Change the SSL probe to issue a GET request for a large file.

Step 16 Verify that the probe is successful for all servers.

Step 17 Stop the packet capture. This capture will contain packets from before and after the index.txt file modification and the large file request (using a HTTP HEAD and GET request). Verify that this behavior was captured.

Step 18 On a server modify the index.txt by removing the text that was added to alter the hash.

Step 19 Change the SSL probe back to requesting a GET for index.txt. Clear the probe counters and verify that the servers and probes are operational.

Step 20 Start a packet capture on the NAM to capture the SSL probe traffic.

Step 21 Configure the SSL probe to do a regex match for a specific string in the data that is returned by the probe.

Step 22 The index.txt page does not contain the regex expression that the probe is looking for, so all of the probes and servers should fail.

Step 23 Change the SSL probe to issue a GET request for a different file that has the correct regex expression.

Step 24 The index_regex.txt page contains the regex expression, so all of the probes and servers should become operational.

Step 25 On server LOCAL-244 rename the index_regex.txt file to something different.

Step 26 By renaming the file from rserver LOCAL-244, the server will return a status code of 404, causing the probe to fail. Verify that the probe and rserver fail due to an invalid response code (404) by issuing the following commands:

Step 27 Stop the packet capture. This capture will contain packets from before and after the URL file requested was changed and after the file was removed. Verify that the regex data is in the index_regex.txt and not the index.txt and also that 404 responses were sent after the file was removed.

Step 28 On a server, rename the index_regex.txt.bak file back its original name.

Step 29 Change the probe back to its original config by removing the expect regex and issuing a GET to the index.txt url.

Step 30 Remove the SSL probe from the serverfarm .

Step 31 Add back the SSL probe to any serverfarm from which it was removed at the beginning of this test.

Expected Results

The following test results are anticipated:

- We expect the probe to fail when a requested URL is modified altering the computed hash.
- We expect the probe to fail when the server response does not contain an expected string.
- We expect the probe to modify its client hello packets when configured for specific SSL options.
- We expect the probe to fail when the server response code does not match or falls outside the range configured.
- We expect that the load balancer will not crash or become unresponsive.

Results

SSL Probe—ACE passed.

SSL Probes—CSM

Health monitoring on the load balancer tracks the state of a server by sending out probes. The load balancer verifies the server response or checks for any network problems that can prevent a client to reach a server. Based on the server response, the load balancer can place the server in or out of service, and can make reliable load balancing decisions. This test verified the functionality of SSL probes in a number of successful and failure situations.

Relevant Load Balancer Configuration

```
!
 script file disk0:c6slb-apc.tcl
!
 probe SSLPROBE script
  script SSL_PROBE_SCRIPT 0
  interval 5
  failed 10
!
 serverfarm GEN-443
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-IIS-152
   inservice
  probe SSLPROBE
!
 vserver GEN-443
  virtual 192.168.120.200 tcp https
  serverfarm GEN-443
  sticky 1 group 113
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the SSL Probes—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify the serverfarm configuration.

Step 3 Verify the probe configuration.

Step 4 Verify that the servers in the serverfarm are operational.

Step 5 Clear the load balancer counters and send an SSL traffic stream. Verify that the servers received the traffic.

Step 6 Verify that the servers in the serverfarm received the SSL traffic.

Step 7 On a Linux server, stop the SSL daemon.

Step 8 Clear the load balancer counters and send an SSL traffic stream. Verify that the server whose SSL daemon was stopped DID NOT receive the traffic.

Step 9 Verify that the Linux server that had its SSL daemon stopped displays an PROBE_FAILED status.

Step 10 On a Linux server, start the SSL daemon.

Step 11 Verify that the Linux server that had its SSL daemon started displays an OPERATIONAL status.

Step 12 Clear the load balancer counters and send an SSL traffic stream. Verify that the servers received the traffic.

Expected Results

The following test results are anticipated:

- We expect the probe to fail when traffic to it is blocked to the rserver for TCP based probes.
- We expect the probe to fail when the rserver responds with ICMP port unreachables for UDP based probes.
- We expect the probe to fail when the server does not have the requested file.
- We expect that the load balancer will not crash or become unresponsive.

Results

SSL Probes—CSM passed.

Redundancy

Health and Redundancy tests verify that the load balancer module can perform the features which continually monitor health and server availability. Such features include backup serverfarms, configuration synchronization, probes, redundancy, recovery from interface/vlan/module resets, and fault tolerant tracking.

Redundancy (or fault tolerance) uses a maximum of two load balancers in the same Catalyst 6500 switch or in separate switches to ensure that your network remains operational even if one of the modules becomes unresponsive. This feature enhances your network users' experience by helping to ensure that your network services and applications are available to them.

This section contains the following topics:

- Multiple Chassis Redundancy—ACE, page 91
- Multiple Chassis Redundancy—CSM, page 95

Multiple Chassis Redundancy—ACE

Redundancy for the load balancer may be built into the 6500 in a couple of ways. A maximum of two load balancers (peers) may be configured in the same Catalyst 6500 switch or in different chassis for redundancy. Each peer module may contain one or more fault tolerant (FT) groups. Each FT group consists of two members: one active context and one standby context. By default, the load balancer will replicate most types of traffic and sticky entries if configured. The health of each load balancer (peer) is monitored primarily through a dedicated VLAN, called the fault tolerant (FT) VLAN, but additional tracking can be configured to cause a redundancy failover.

This test verified the ability of the load balancer configured in multiple chassis to properly failover, maintain replicated connections and sticky entries during a number of redundancy changes.

Relevant Load Balancer Configuration

```
PRIMARY
ft interface VLAN 900
  ip address 192.168.1.1 255.255.255.0
  peer ip address 192.168.1.2 255.255.255.0
  no shutdown
ft peer 1
  heartbeat interval 300
  heartbeat count 10
  ft-interface vlan 900
ft group 1
  peer 1
  priority 200
  associate-context Admin
  inservice
ft group 2
  peer 1
  no preempt
  priority 200
  associate-context SH-Gold
  inservice
STANDBY
serverfarm host GEN-443
  probe SSL
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-80
  probe TCP
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-FTP
  probe FTP
  rserver BRG-13
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-UDP
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host GEN2-80
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-153
    inservice
sticky ip-netmask 255.255.255.255 address source STKY-GRP-30
  timeout 40
  replicate sticky
  serverfarm GEN-80
sticky http-cookie cookie-gold-grp40 STKY-GRP-40
  cookie insert browser-expire
  timeout 1
  replicate sticky
  serverfarm GEN2-80
sticky ip-netmask 255.255.255.255 address both STKY-GRP-31
  timeout 40
  replicate sticky
  serverfarm GEN-UDP
sticky ip-netmask 255.255.255.255 address both STKY-GRP-32
  timeout 40
  replicate sticky
  serverfarm GEN-FTP
class-map match-all GEN-NAT_120
  2 match virtual-address 192.168.120.120 any
class-map match-all GEN-VIP_120:21
  2 match virtual-address 192.168.120.120 tcp eq ftp
class-map match-all GEN-VIP_120:443
  2 match virtual-address 192.168.120.120 tcp eq https
class-map match-all GEN-VIP_120:80
  2 match virtual-address 192.168.120.120 tcp eq www
class-map match-all GEN-VIP_120:UDP
  2 match virtual-address 192.168.120.120 udp any
class-map type http loadbalance match-all INDEX.HTML
  2 match http url /index.html
policy-map type loadbalance first-match PLBSF_GEN-443
  class class-default
    serverfarm GEN-443
policy-map type loadbalance first-match PLBSF_GEN-80
  class INDEX.HTML
    sticky-serverfarm STKY-GRP-40
  class class-default
    sticky-serverfarm STKY-GRP-30
policy-map type loadbalance first-match PLBSF_GEN-FTP
  class class-default
    sticky-serverfarm STKY-GRP-32
policy-map type loadbalance first-match PLBSF_GEN-UDP
  class class-default
    sticky-serverfarm STKY-GRP-31
policy-map multi-match SH-Gold-VIPs
  class GEN-NAT_120
    nat dynamic 1 vlan 120
  class GEN-VIP_120:80
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-80
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class GEN-VIP_120:443
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-443
    loadbalance vip icmp-reply active
  class GEN-VIP_120:21
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class GEN-VIP_120:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-UDP
    loadbalance vip icmp-reply active
    connection advanced-options 2SECOND-IDLE
ft track host GW_251-252
  track-host 192.168.16.251
  peer track-host 192.168.16.252
  probe HA-ICMP priority 10
  peer probe HA-ICMP priority 5
  priority 110
  peer priority 5
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 110
  peer priority 5
ft track interface Int_4/37_6/13_V99
  track-interface vlan 99
  peer track-interface vlan 99
  priority 110
  peer priority 5
ft track host RT-241
  track-host 172.29.0.241
  probe HA-TCP:554 priority 50
  probe HA-TCP:1755 priority 60
```

Test Procedure

The procedure used to perform the Multiple Chassis Redundancy—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary load balancer module, verify that HA (High Availability) redundancy is operational.

Step 3 Clear select counters on the primary and secondary load balancers.

Step 4 On a Linux client, launch a test tool that will generate long-lived HTTP, HTTPS, FTP, and UDP connections. These flows represent server load balanced traffic and are being sent to a specific VIP.

Step 5 Verify that the Primary load balancer is servicing these requests.

Step 6 Verify that two sticky entries were created for each sticky group specified, on both the primary and secondary load balancers.

Step 7 On a client, start a continuous series of pings to a specific VIP.

Step 8 On the primary load balancer, disable FT preemption to allow a command forced failover and then force a failover.

Step 9 After failover has occurred stop client pings to the VIP and verify that pings to the VIP were not severely impacted by failover. From the output of the ping command, verify that no more then 1 packet was lost during failover.

Step 10 Verify that the secondary load balancer has become active and is now servicing requests.

Step 11 Verify on the Standby load balancer that the sticky entries remain unchanged from those captured in an earlier step.

Step 12 Force a failover back to the Primary load balancer by enabling preemption for a specific FT group. This will allow the higher priority load balancer to preempt and become active.

Step 13 Verify that the Primary has become active and is now servicing requests.

Step 14 Verify on the Primary load balancer that the sticky entries remain unchanged from those captured in an earlier step.

Step 15 On a client, start a continuous series of pings to a specific VIP.

Step 16 On the aggregation switch, shutdown the primary load balancer.

Step 17 After failover has occurred stop client pings to the VIP and verify that pings to the VIP were not severely impacted by the failover. Verify that no more then 3-4 packets were lost during failover.

Step 18 Verify that the Standby load balancer has become active and is now servicing requests.

Step 19 Verify on the Standby load balancer that the sticky entries remain unchanged from those previously captured.

Step 20 Verify that when the reloaded Primary load balancer comes back online it preempts, becomes active, and the sticky entries remain unchanged from the entries previously observed.

Step 21 Force a failover of a specific context on the Primary load balancer by causing the tracked HSRP group to fail. Verify that redundancy has changed only on the index for that context.

Step 22 Allow tracked HSRP to become active again on the Primary load balancer and verify that the Primary load balancer becomes active for all contexts.

Step 23 Verify on the Primary load balancer that the sticky entries remain unchanged from those previously captured.

Step 24 Verify that with preempt disabled, a reloaded module with a higher priority will not become active. Disable FT preemption on the Primary load balancer and reload the Primary load balancer.

Step 25 Verify that when the reloaded Primary load balancer comes back online it remains in standby and the sticky entries remain unchanged from those previously observed.

Step 26 Stop the client generated traffic and verify that no errors were seen.

Step 27 Enable preemption on the Primary load balancer for ft group 2.

Expected Results

The following test results are anticipated:

- We expect the load balancer to replicate connections and sticky table entries.
- We expect the standby to become active and service the persistent replicated connections maintaining sticky.
- We expect the load balancer to preempt after a failure when configured.
- We expect that the load balancer will not crash or become unresponsive.

Results

Multiple Chassis Redundancy—ACE passed.

Multiple Chassis Redundancy—CSM

Redundancy for the load balancer may be built into the 6500 in a couple of ways. A maximum of two load balancers (peers) may be configured in the same Catalyst 6500 switch or in different chassis for redundancy. Each peer module may contain one or more fault tolerant (FT) groups. Each FT group consists of two members: one active context and one standby context. By default, the load balancer will replicate most types of traffic and sticky entries if configured. The health of each load balancer (peer) is monitored primarily through a dedicated VLAN, called the fault tolerant (FT) VLAN, but additional tracking can be configured to cause a redundancy failover.

This test verified the ability of the load balancer configured in multiple chassis to properly failover, maintain replicated connections and sticky entries during a number of redundancy changes.

Relevant Load Balancer Configuration

```
Primary Load Balancer (6k1)
!
 real LOCAL-LINUX-242
  address 172.29.0.242
  inservice
 real LOCAL-IIS-241
  address 172.29.0.241
  inservice
 real RT-IIS-152
  address 172.28.0.152
  inservice
 real RT-LINUX-154
  address 172.28.0.154
  inservice
!
 sticky 30 netmask 255.255.255.255 address source timeout 30
!
 sticky 110 ssl timeout 30
!
 serverfarm CS-SERVERS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm CS-SERVERS-80
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-154
   inservice
!
 vserver WEB
  virtual 192.168.120.210 tcp https
  serverfarm CS-SERVERS
  sticky 30 group 110
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver WEB-80
  virtual 192.168.120.210 tcp www
  serverfarm CS-SERVERS-80
  sticky 30 group 30
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
```

```
 ft group 2 vlan 900
  priority 110 alt 100
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.251
  track interface GigabitEthernet4/37
  track mode any
!
Standby Load Balancer (6k2)
!
 real LOCAL-LINUX-242
  address 172.29.0.242
  inservice
 real LOCAL-IIS-241
  address 172.29.0.241
  inservice
real RT-IIS-152
  address 172.28.0.152
  inservice
 real RT-LINUX-154
  address 172.28.0.154
  inservice
!
 sticky 30 netmask 255.255.255.255 address source timeout 30
!
 sticky 110 ssl timeout 30
!
 serverfarm CS-SERVERS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm CS-SERVERS-80
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-154
   inservice
!
 vserver WEB
  virtual 192.168.120.210 tcp https
  serverfarm CS-SERVERS
  sticky 30 group 110
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver WEB-80
  virtual 192.168.120.210 tcp www
  serverfarm CS-SERVERS-80
  sticky 30 group 30
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 ft group 2 vlan 900
  priority 100 alt 110
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.252
  track interface GigabitEthernet4/38
  track mode any
!
```

Test Procedure

The procedure used to perform the Multiple Chassis Redundancy—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the primary load balancer module, verify that HA (High Availability) redundancy is operational.

Step 3 Clear select counters on the primary and secondary load balancers.

Step 4 On a Linux client, launch a test tool that will generate long-lived HTTP, HTTPS, FTP, and UDP connections.

Step 5 Verify that the Primary load balancer is servicing these requests.

Step 6 Verify that two sticky entries were created for each sticky group specified, on both the primary and secondary load balancers.

Step 7 On a client, start a continuous series of pings to a specific VIP.

Step 8 On the primary load balancer, disable FT preemption to allow a command forced failover and then force a failover.

Step 9 After failover has occurred stop client pings to the VIP and verify that pings to the VIP were not severely impacted by failover.

Step 10 Verify that the secondary load balancer has become active and is now servicing requests.

Step 11 Verify on the Standby load balancer that the sticky entries remain unchanged from those captured in an earlier step.

Step 12 Force a failover back to the Primary load balancer by enabling preemption for a specific FT group. This will allow the higher priority load balancer to preempt and become active.

Step 13 Verify that the Primary has become active and is now servicing requests.

Step 14 Verify on the Primary load balancer that the sticky entries remain unchanged from those captured in an earlier step.

Step 15 On a client, start a continuous series of pings to a specific VIP.

Step 16 On the aggregation switch, shutdown the primary load balancer.

Step 17 After failover has occurred stop client pings to the VIP and verify that pings to the VIP were not severely impacted by the failover.

Step 18 Verify that the Standby load balancer has become active and is now servicing requests.

Step 19 Verify on the Standby load balancer that the sticky entries remain unchanged from those previously captured.

Step 20 Verify that when the reloaded Primary load balancer comes back online it preempts, becomes active, and the sticky entries remain unchanged from the entries previously observed.

Step 21 Force a failover of a specific context on the Primary load balancer by causing the tracked HSRP group to fail. Verify that redundancy has changed only on the index for that context.

Step 22 Allow tracked HSRP to become active again on the Primary load balancer and verify that the Primary load balancer becomes active for all contexts.

Step 23 Verify on the Primary load balancer that the sticky entries remain unchanged from those previously captured.

Step 24 Verify that with preempt disabled, a reloaded module with a higher priority will not become active. Disable FT preemption on the Primary load balancer and reload the Primary load balancer.

Step 25 Verify that when the reloaded Primary load balancer comes back online it remains in standby and the sticky entries remain unchanged from those previously observed.

Step 26 Stop the client generated traffic and verify that no errors were seen.

Step 27 Enable preemption on the Primary load balancer for ft group 2.

Expected Results

The following test results are anticipated:

- We expect the load balancer to replicate connections and sticky table entries.
- We expect the standby to become active and service the persistent replicated connections maintaining sticky.
- We expect the load balancer to preempt after a failure when configured.
- We expect that the load balancer will not crash or become unresponsive.

Results

Multiple Chassis Redundancy—CSM passed with exception. The following exceptions were noted: CSCek51826.

Tracking

Health and Redundancy tests verify that the load balancer module can perform the features which continually monitor health and server availability. Such features include backup serverfarms, configuration synchronization, probes, redundancy, recovery from interface/vlan/module resets, and fault tolerant tracking.

The Tracking feature allows you to designate certain network items as critical so that, if one or more items fail, the load balancer module reduces the priority of the associated active FT group accordingly. If the priority of the active FT group falls below the priority of the corresponding FT group on the standby, a switchover occurs.

This section contains the following topics:

- Fault Tolerant Tracking—ACE, page 100
- Fault Tolerant Tracking—CSM, page 102

Fault Tolerant Tracking—ACE

This feature allows the load balancer to track the health of an HSRP group, interface, or gateway, forcing a failover when its priority is less than the standby load balancer. This test verified that the load balancer performed as expected and failover redundancy appropriately. HSRP tracking is no available on Scimitar so these steps will be skipped.

Relevant Load Balancer Configuration

```
Active Load Balancer
!
probe icmp HA-ICMP
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:1755
  port 1755
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:554
  port 554
  interval 2
  faildetect 2
  passdetect interval 2
ft track host GW_251-252
  track-host 192.168.16.251
  peer track-host 192.168.16.252
  probe HA-ICMP priority 10
  peer probe HA-ICMP priority 5
  priority 110
  peer priority 5
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 110
  peer priority 5
ft track interface Int_4/37_6/13_V99
  track-interface VLAN 99
  peer track-interface VLAN 99
  priority 110
  peer priority 5
ft track host RT-241
  track-host 172.29.0.241
  probe HA-TCP:554 priority 50
  probe HA-TCP:1755 priority 60
!
Standby Load Balancer
!
ft track host GW_251-252
  track-host 192.168.16.252
  peer track-host 192.168.16.251
  peer probe HA-ICMP priority 10
  probe HA-ICMP priority 5
  priority 5
  peer priority 110
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 5
  peer priority 110
ft track interface Int_4/37_6/13_V99
```

```
  track-interface VLAN 99
  peer track-interface vlan 99
  priority 5
  peer priority 110
ft track host RT-241
  peer track-host 172.29.0.241
  peer probe HA-TCP:554 priority 50
  peer probe HA-TCP:1755 priority 60
```

Test Procedure

The procedure used to perform the Fault Tolerant Tracking —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the FT statistics for both the primary and secondary load balancers.

Step 3 On the aggregation switch, verify the standby state of the configured interfaces.

Step 4 Check the FT group and tracking states for both the primary and secondary load balancers.

Step 5 Disable access to the tracked gateway.

Step 6 Allow access to the tracked gateway.

Step 7 On the aggregation switch, force the tracked HSRP group to go into standby by shutting down its tracked interface.

Step 8 On the aggregation switch, bring the HSRP group back online by enabling the tracked interface. Verify that the tracked HSRP group becomes active and that the primary load balancer preempts to resume the active role.

Step 9 Interface tracking actually tracks the VLAN that is configured. In order for the tracked interface to fail both links would need to go down. On a port-channel with two GigabitEthernet interfaces, verify that redundancy *DOES NOT* switch to the secondary load balancer by disabling one of the GigabitEthernet interfaces.

Step 10 Force a second GigabitEthernet interface failure, and verify that redundancy *DOES* switch to the secondary load balancer.

Step 11 Bring the tracked interface back online by enabling the port-channel. Verify that the primary load balancer preempts to resume the active role.

Step 12 Verify that the primary load balancer *WILL NOT* failover when the priority is equal to the standby load balancer prior to initiating a failover. Change the peer priority on the active load balancer that tracks a specific HSRP group. This will cause the net priority on the standby load balancer to change to a priority which will equal the priority on the primary load balancer when failover is induced and *WILL NOT* cause a failover.

Step 13 Disable access to the tracked gateway by disabling a specific VLAN on the aggregation switch. Verify that pings from the primary load balancer to the tracked GW fail and that redundancy *HAS NOT* switched to the secondary load balancer.

Step 14 Enable access to the tracked gateway. Change the peer priority for the tracked HSRP group to its previous value. Verify that pings from the primary load balancer to the tracked GW are successful.

Expected Results

The following test results are anticipated:

- We expect the active load balancer to failover to the standby when the priority is less than that of the standby load balancer.
- We expect the load balancer not to failover to the standby if the standbys priority is equal to or less than the active load balancer.
- We expect that the load balancer will not crash or become unresponsive.

Results

Fault Tolerant Tracking —ACE passed.

Fault Tolerant Tracking—CSM

This feature allows the load balancer to track the health of an HSRP group, interface, or gateway, forcing a failover when its priority is less than the standby load balancer. This test verified that the load balancer performed as expected and failover redundancy appropriately. HSRP tracking is no available on Scimitar so these steps will be skipped.

Relevant Load Balancer Configuration

```
Active Load Balancer
!
 ft group 2 vlan 900
  priority 110 alt 100
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.251
  track interface GigabitEthernet4/37
  track mode any
!
Standby Load Balancer
!
 ft group 2 vlan 900
  priority 100 alt 110
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.252
  track interface GigabitEthernet4/38
  track mode any
!
```

Test Procedure

The procedure used to perform the Fault Tolerant Tracking—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Display the FT statistics for both the primary and secondary load balancers.

Step 3 On the aggregation switch, verify the standby state of the configured interfaces.

Step 4 Disable access to the tracked gateway. Verify that pings to the tracked GW fail and that redundancy has switched to the other device.

Step 5 Allow access to the tracked gateway. Verify that the tracked GW becomes active and that the original load balancer preempts to resume the active role.

Step 6 On the aggregation switch, force the tracked HSRP group to go into standby by shutting down its tracked interface. Verify that the tracked HSRP group is down and that redundancy has switched to the other device.

Step 7 On the aggregation switch, bring the HSRP group back online by enabling the tracked interface. Verify that the tracked HSRP group becomes active and that the primary load balancer preempts to resume the active role.

Step 8 Interface tracking actually tracks the VLAN that is configured. In order for the tracked interface to fail both links would need to go down. On a port-channel with two GigabitEthernet interfaces, verify that redundancy *DOES NOT* switch to the secondary load balancer by disabling one of the GigabitEthernet interfaces.

Step 9 Force a second GigabitEthernet interface failure, and verify that redundancy *DOES* switch to the secondary load balancer.

Step 10 Bring the tracked interface back online by enabling the port-channel. Verify that the primary load balancer preempts to resume the active role.

Step 11 Verify that the primary load balancer *WILL NOT* failover when the priority is equal to the standby load balancer prior to initiating a failover. Change the peer priority on the active load balancer that tracks a specific HSRP group. This will cause the net priority on the standby load balancer to change to a priority which will equal the priority on the primary load balancer when failover is induced and *WILL NOT* cause a failover.

Step 12 On the primary load balancer, initiate a config sync with the standby load balancer. Verify that the standby load balancer priority has changed.

Step 13 Disable access to the tracked gateway by disabling a specific VLAN on the aggregation switch. Verify that pings from the primary load balancer to the tracked GW fail and that redundancy *HAS NOT* switched to the secondary load balancer.

Step 14 Enable access to the tracked gateway. Change the peer priority for the tracked HSRP group to its previous value. Verify that pings from the primary load balancer to the tracked GW are successful.

Step 15 On the primary load balancer, initiate a config sync with the standby load balancer. Verify that the standby load balancer priority has returned to its original value.

Expected Results

The following test results are anticipated:

- We expect the active load balancer to failover to the standby when the priority is less than that of the standby load balancer.
- We expect the load balancer not to failover to the standby if the standbys priority is equal to or less than the active load balancer.
- We expect that the load balancer will not crash or become unresponsive.

Results

Fault Tolerant Tracking—CSM passed with exception. The following exceptions were noted: CSCek51743.

Load Balancing

Load Balancing tests verify that the load balancer module can perform the features which facilitate server load balancing (SLB). Predictors are the specific SLB feature covered by this category.

This section contains the following topics:

- Predictors, page 104

Predictors

Load Balancing tests verify that the load balancer module can perform the features which facilitate server load balancing (SLB). Predictors are the specific SLB feature covered by this category.

Depending on the load-balancing algorithm or predictor that you configure, the load balancer performs a series of checks and calculations to determine the server that can best service each client request. The load balancer bases server selection on several factors, including the server with the fewest connections with respect to load, source or destination address, cookies, URLs, or HTTP headers.

This section contains the following topics:

- Least Connection Predictor—ACE, page 104
- Least Connection Predictor—CSM, page 108
- Maxconn Connection Limiter—ACE, page 111
- Maxconn Connection Limiter—CSM, page 113

Least Connection Predictor—ACE

When a server farm is chosen for a connection, the connection is sent to a rserver based on one of several load-balancing predictors. This procedure will test for the proper functionality of the leastconns load balancing predictor. The leastconns predictor load balances connections to the server that has the least number of open connections.

This test verified that connections were sent to the server with the least number of open connections and was not adversely effected by sticky.

Relevant Load Balancer Configuration

```
probe icmp PRED-PING
  ip address 172.29.0.243 routed
  interval 5
  faildetect 2
  passdetect interval 2
parameter-map type http PERSIST-REBALANCE
  persistence-rebalance
parameter-map type connection PRED-CONNS-UDP_CONN
  set timeout inactivity 300
serverfarm host PRED-CONNS
  predictor leastconns
  rserver BRG-11
    inservice
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver BRG-14
```

```
    inservice
  rserver BRG-15
    inservice
  rserver LOCAL-240
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
  rserver RT-154
    inservice
  rserver RT-155
    inservice
serverfarm host PRED-CONNS-UDP
  failaction purge
  predictor leastconns
  rserver BRG-11 2222
    inservice
  rserver LOCAL-240 2222
    inservice
  rserver LOCAL-242 2222
    inservice
  rserver LOCAL-244 2222
    probe PRED-PING
    inservice
  rserver RT-151 2222
    inservice
  rserver RT-153 2222
    inservice
  rserver RT-154 2222
    inservice
serverfarm host PREDICTOR
  probe TCP
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
sticky http-cookie COOKIE_TEST STKY-GRP-43
  cookie offset 1 length 999
  timeout 30
  replicate sticky
  serverfarm PREDICTOR
class-map match-all PRED-CONNS-UDP-VIP_128:2222
  2 match virtual-address 192.168.120.128 udp eq 0
class-map match-all PRED-CONNS-VIP_128:80
  2 match virtual-address 192.168.120.128 tcp eq www
```

```
class-map match-all PREDICTOR_117:80
  2 match virtual-address 192.168.120.117 tcp eq www
policy-map type loadbalance first-match PLBSF_PRED-CONNS
  class class-default
    serverfarm PRED-CONNS
policy-map type loadbalance first-match PLBSF_PRED-CONNS-UDP
  class class-default
    serverfarm PRED-CONNS-UDP
policy-map type loadbalance first-match PLBSF_PREDICTOR
  class class-default
    serverfarm PREDICTOR
policy-map multi-match SH-Gold-VIPs
  class PREDICTOR_117:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PREDICTOR
    loadbalance vip icmp-reply active
    nat dynamic 1 VLAN 30
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PRED-CONNS-VIP_128:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PRED-CONNS
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 30
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PRED-CONNS-UDP-VIP_128:2222
    loadbalance vip inservice
    loadbalance policy PLBSF_PRED-CONNS-UDP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 30
    appl-parameter http advanced-options PERSIST-REBALANCE
    connection advanced-options PRED-CONNS-UDP_CONN
```

Test Procedure

The procedure used to perform the Least Connection Predictor—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the primary and standby load balancers.

Step 3 Verify that the predictor leastconn is configured as the serverfarm predictor.

Step 4 On a Linux client, send a series of HTTP traffic consisting of six long lived (will not close for four minutes) HTTP 1.1 connections. This is used to keep a current connection count on several servers.

Step 5 Make sure at least one server has a current connection count of two before you proceed. Verify that the servers getting these connections.

Step 6 On a Linux client, before the HTTP 1.1 test traffic finishes, send a series of short lived HTTP 1.0 connections.

Step 7 Because this is a multi CPU module it requires a server with two current connections to be considered at a higher connection count than zero or one. Any server with two current connections prior to start of the HTTP 1.0 traffic will not get any new connections. Verify that these connections were load balanced correctly.

Step 8 The following steps will test the effect of introducing a two second response delay on a subset of load balanced servers. Clear select counters on the primary and standby load balancers.

Step 9 On a Linux client, launch a test tool that will generate a number of concurrent connections to each server.

Step 10 Verify that the total connection count is fairly even amongst each group of servers in the serverfarm.

Step 11 Introduce a response delay of two seconds on select servers.

Step 12 Clear select counters on the load balancer and then check the connection distribution on the serverfarm.

Step 13 To test the effect of busy servers recovering from being under load, remove the delay that was artificially introduced.

Step 14 Clear the counters and then check the connection distribution on the serverfarm. Verify that the total connection count is fairly even amongst each group of servers.

Step 15 Stop the client generated traffic.

Step 16 These steps will test the effect of fault tolerant failover with an unstable server by forcing a reload, while an server is down, and then having it come online afterward. Clear select counters on the load balancer and verify that the leastconn predictor is configured.

Step 17 Remove preempt from the fault tolerant group.

Step 18 On a Linux client, generate UDP client requests.

Step 19 Force a probe failure. Verify that a server has failed and is no longer receiving new connections, then reset the module.

Step 20 Once the load balancer is back online it will remain in a standby state because preempt was removed. Verify the state and that the server is still down and not receiving connections.

Step 21 On the active load balancer, force fault tolerance to fail back over to the original active (primary) load balancer and verify the change.

Step 22 Make the probe operational by changing the IP address to a valid address. Verify that the server becomes operational and that after short period of time its connection count is even.

Step 23 Stop the UDP client traffic.

Step 24 Add the preempt command back to the fault-tolerant configuration on the active load balancer.

Expected Results

The following test results are anticipated:

- We expect minimal traffic to be load balanced to rservers with a higher connection count.
- We expect slower servers to receive less total connections than faster servers.
- We expect normal operation to resume if the slowdown on these servers is temporary and corrected.
- We expect a newly activated server would behave correctly after a redundancy transition.
- We expect that having sticky configured will not adversely effect load balancing of non sticky connections.
- We expect no CPU or memory problems.

Results

Least Connection Predictor—ACE passed.

Least Connection Predictor—CSM

When a server farm is chosen for a connection, the connection is sent to a rserver based on one of several load-balancing predictors. This procedure will test for the proper functionality of the leastconns load balancing predictor. The leastconns predictor load balances connections to the server that has the least number of open connections.

This test verified that connections were sent to the server with the least number of open connections and was not adversely effected by sticky.

Relevant Load Balancer Configuration

```
!
 probe PRED-PING icmp
  address 172.29.0.1 routed
  interval 5
  retries 2
  failed 3
!
 serverfarm PREDICTOR
  nat server
  nat client CLIENT_NAT
  predictor hash url
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm PRED-CONNS
  nat server
  nat client CLIENT_NAT
  predictor leastconns
  real name BRG-LINUX-11
   inservice
  real name BRG-LINUX-12
   inservice
  real name BRG-LINUX-13
   inservice
  real name BRG-LINUX-14
   inservice
  real name BRG-LINUX-15
   inservice
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-IIS-241
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-243
   inservice
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-151
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   health probe PRED-PING
   inservice
  probe HTTP
!
 serverfarm PRED-CONNS-UDP
  nat server
  nat client CLIENT_NAT
  predictor leastconns
  failaction purge
  real name BRG-LINUX-11 2222
   inservice
  real name BRG-LINUX-12 2222
   inservice
  real name LOCAL-LINUX-240 2222
   inservice
  real name LOCAL-LINUX-242 2222
   inservice
  real name LOCAL-LINUX-244 2222
   inservice
  real name RT-LINUX-151 2222
   inservice
  real name RT-LINUX-153 2222
   inservice
  real name RT-LINUX-154 2222
   health probe PRED-PING
   inservice
  probe ICMP
!
 vserver PREDICTOR
  virtual 192.168.120.217 tcp www
  serverfarm PREDICTOR
  persistent rebalance
  inservice
!
 vserver PRED-CONNS
  virtual 192.168.120.204 tcp www
  serverfarm PRED-CONNS
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver PRED-CONNS-UDP
  virtual 192.168.120.204 udp 0
  unidirectional
  serverfarm PRED-CONNS-UDP
  idle 300
  pending 60
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Least Connection Predictor—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Verify that the predictor leastconn is configured as the serverfarm predictor.

Step 4 On a Linux client, send a series of HTTP traffic consisting of six long lived (will not close for four minutes) HTTP 1.1 connections. This is used to keep a current connection count on several servers.

Step 5 Make sure at least one server has a current connection count of two before you proceed. Verify that the servers getting these connections.

Step 6 On a Linux client, before the HTTP 1.1 test traffic finishes, send a series of short lived HTTP 1.0 connections.

Step 7 Because this is a multi CPU module it requires a server with two current connections to be considered at a higher connection count than zero or one. Any server with two current connections prior to start of the HTTP 1.0 traffic will not get any new connections. Verify that these connections were load balanced correctly.

Step 8 The following steps will test the effect of introducing a two second response delay on a subset of load balanced servers. Clear select counters on the load balancer.

Step 9 On a Linux client, launch a test tool that will generate a number of concurrent connections to each server.

Step 10 Verify that the total connection count is fairly even amongst each group of servers in the serverfarm.

Step 11 Introduce a response delay of two seconds on select servers.

Step 12 Clear select counters on the load balancer and then check the connection distribution on the serverfarm.

Step 13 To test the effect of busy servers recovering from being under load, remove the delay that was artificially introduced.

Step 14 Clear the counters and then check the connection distribution on the serverfarm. Verify that the total connection count is fairly even amongst each group of servers.

Step 15 Stop the client generated traffic.

Step 16 These steps will test the effect of fault tolerant failover with an unstable server by forcing a reload, while an server is down, and then having it come online afterward. Clear select counters on the load balancer and verify that the leastconn predictor is configured.

Step 17 Remove preempt from the fault tolerant group.

Step 18 On a Linux client, generate UDP client requests.

Step 19 Force a probe failure. Verify that a server has failed and is no longer receiving new connections, then reset the module.

Step 20 Once the load balancer is back online it will remain in a standby state because preempt was removed. Verify the state and that the server is still down and not receiving connections.

Step 21 Make the probe operational by changing the IP address to a valid address. Verify that the server becomes operational and that after short period of time its connection count is even.

Step 22 Stop the UDP client traffic.

Step 23 Add the preempt command back to the fault-tolerant configuration on the active load balancer.

Expected Results

The following test results are anticipated:

- We expect minimal traffic to be load balanced to rservers with a higher connection count.
- We expect slower servers to receive less total connections than faster servers.
- We expect normal operation to resume if the slowdown on these servers is temporary and corrected.

- We expect a newly activated server would behave correctly after a redundancy transition.
- We expect no CPU or memory problems.

Results

Least Connection Predictor—CSM passed.

Maxconn Connection Limiter—ACE

With the maxconn feature configured the Load Balancer will limit the number of current connections to a particular server. Once this threshold is reached new connections will not be load balanced to this server until the current connection count goes below the configured minconn limit.

Relevant Load Balancer Configuration

```
parameter-map type http PERSIST-REBALANCE
  persistence-rebalance
parameter-map type connection HALF-CLOSE_4s
  set tcp timeout half-closed 4
serverfarm host MAX-CONN
  probe HTTP
  rserver RT-151
    conn-limit max 4 min 2
    inservice
serverfarm host MAX-CONN2
  probe HTTP
  rserver LOCAL-243
    conn-limit max 500 min 2
    inservice
  rserver RT-151
    conn-limit max 500 min 2
    inservice
  rserver RT-151 90
    conn-limit max 500 min 2
    inservice
  rserver RT-151 91
    conn-limit max 500 min 2
    inservice
  rserver RT-151 92
    conn-limit max 500 min 2
    inservice
  rserver RT-151 93
    conn-limit max 500 min 2
    inservice
  rserver RT-151 94
    conn-limit max 500 min 2
    inservice
  rserver RT-151 95
    conn-limit max 500 min 2
    inservice
  rserver RT-154
    inservice
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-29
  timeout 30
  replicate sticky
  serverfarm MAX-CONN
class-map match-all MAX-CONN-VIP_126:80
  2 match virtual-address 192.168.120.126 tcp eq www
policy-map type loadbalance first-match PLBSF_MAX-CONN
```

```
  class INDEX.HTML
    sticky-serverfarm STICKY-GROUP-29
  class class-default
    serverfarm MAX-CONN2
policy-map multi-match SH-Gold-VIPs2
  class MAX-CONN-VIP_126:80
    loadbalance vip inservice
    loadbalance policy PLBSF_MAX-CONN
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
    connection advanced-options HALF-CLOSE_4s
```

Test Procedure

The procedure used to perform the Maxconn Connection Limiter—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Test the configured maxconn limit of four connections by starting four client emulators from a single Linux client. Each client will generate a long lived HTTP connection. Verify that the connections were successful, persistent (tcp ports not changing), and without error.

Step 4 Start a fifth client emulator from the same client, which will immediately see a reset from the load balancer. Verify that the loadbalance statistics and serverfarm details increment server unavailable and out-of-rotation counts. Logging should also log a message indicating that max conns has been reached.

Step 5 Stop the fifth client emulator.

Step 6 From a different client start an emulator at a rate of several hundred CPS, hitting the default serverfarm.

Step 7 Verify that the default serverfarm is able to handle new connections even though shared servers with a different serverfarm are at the max-conn limit and are not accepting new connections.

Step 8 On both the primary and Standby load balancers verify that the connections and counters are accurate.

Step 9 These steps will test the effect of a fault-tolerant failover. On the primary load balancer, disable preempt for fault tolerance and force a failover to the standby load balancer.

Step 10 Verify that redundancy has changed and that the active load balancer is handling open connections.

Step 11 From the same client used for the long lived HTTP connections start a fifth client emulator. Verify that this connection fails with "Connection reset by peer" seen through the CLI. Stop the fifth client emulator.

Step 12 Test the effect of a second fault-tolerant failover back to the original active load balancer. On the active load balancer force a failover to the (primary) standby load balancer.

Step 13 Verify that redundancy has changed and that the active load balancer is handling the open connections.

Step 14 From the same client used for the long lived HTTP connections start a fifth client emulator. Verify that this connection fails with "Connection reset by peer" seen through the CLI. Stop the fifth client emulator.

Step 15 These steps will test the effect of a load balancer module reload. Force a temporary FT failover to the standby load balancer by resetting the load balancer on the aggregation switch from the router prompt.

Step 16 Verify that after the load balancer reloads it preempts, becomes active and is handling the long lived connections.

Step 17 From the same client used for the long lived HTTP connections start a fifth client emulator. Verify that this connection fails with "Connection reset by peer" seen through the CLI. Stop the fifth client emulator.

Step 18 Verify that the current connection count on the serverfarm does NOT increase once the maxconn limit has been reached.

Step 19 Kill two of the long lived HTTP connections and then check the current connection count on the serverfarm.

Step 20 Kill one of the long lived HTTP connections. Check the current connection count on the serverfarm.

Step 21 From the same client used for the long lived HTTP connections start another client emulator. Verify that this connection is accepted as the minconn value of two which has been exceeded.

Step 22 Stop all client traffic.

Expected Results

The following test results are anticipated:

- We expect the load balancer to either load balance these new connections to another active server or reset the connection once the maxconn limit is exceeded.
- We expect this feature to work reliably after fault-tolerant failovers have occurred.
- We expect this feature not to allow new connections until the minconn limit has been passed after the maxconn was reached surpassed.
- We expect no CPU or memory problems.

Results

Maxconn Connection Limiter—ACE failed. The following failures were noted: CSCse15530, CSCsm12883, CSCsu65844.

Maxconn Connection Limiter—CSM

With the maxconn feature configured the Load Balancer will limit the number of current connections to a particular server. Once this threshold is reached new connections will not be load balanced to this server until the current connection count goes below the configured minconn limit.

Relevant Load Balancer Configuration

```
!
 map M-MAX-CONN url
  match protocol http method GET url /index.html
!
 policy P-MAX-CONN
  url-map M-MAX-CONN
  sticky-group 31
  serverfarm MAX-CONN
!
 sticky 31 netmask 255.255.255.255 timeout 30
!
 serverfarm MAX-CONN
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 3
   minconns 2
   inservice
  probe HTTP
```

```
!
 serverfarm MAX-CONN2
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 500
   inservice
  real name LOCAL-LINUX-240
   maxconns 500
   inservice
  real name LOCAL-IIS-241
   maxconns 500
   inservice
  real name RT-IIS-152
   maxconns 500
   inservice
  real name RT-LINUX-151 90
   maxconns 500
   inservice
  real name RT-LINUX-151 91
   maxconns 500
   inservice
  real name RT-LINUX-151 92
   maxconns 500
   inservice
  real name RT-LINUX-151 93
   maxconns 500
   inservice
  real name RT-LINUX-151 94
   maxconns 500
   inservice
  real name RT-LINUX-151 95
   maxconns 500
   inservice
  probe HTTP
!
 vserver MAX-CONN
  virtual 192.168.120.205 tcp www
  serverfarm MAX-CONN2
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  slb-policy P-MAX-CONN
  inservice
!
```

Test Procedure

The procedure used to perform the Maxconn Connection Limiter—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer and verify serverfarm and server status.

Step 3 Change the variable INFINITE_IDLE_TIME_MAXCONNS to one. Test the configured maxconn limit of four connections by starting four client emulators from a single Linux client. Each client will generate a long lived HTTP connection. Verify that the connections were successful, persistent (tcp ports not changing), and without error.

Step 4 Start a fifth client emulator from the same client, which will immediately see a reset from the load balancer. Verify that the loadbalance statistics and serverfarm details increment server unavailable and out-of-rotation counts. Logging should also log a message indicating that max conns has been reached.

Step 5 Stop the fifth client emulator.

Step 6 From a different client start an emulator at a rate of several hundred CPS, hitting the default serverfarm.

Step 7 Verify that the default serverfarm is able to handle new connections even though shared servers with a different serverfarm are at the max-conn limit and are not accepting new connections.

Step 8 On both the primary and Standby load balancers verify that the connections and counters are accurate.

Step 9 These steps will test the effect of a fault-tolerant failover. On the primary load balancer, disable preempt for fault tolerance and force a failover to the standby load balancer.

Step 10 Verify that redundancy has changed and that the active load balancer is handling open connections.

Step 11 From the same client used for the long lived HTTP connections start a fifth client emulator. Verify that this connection fails with "Connection reset by peer" seen through the CLI. Stop the fifth client emulator.

Step 12 Test the effect of a second fault-tolerant failover back to the original active load balancer. On the active load balancer force a failover to the (primary) standby load balancer.

Step 13 Verify that redundancy has changed and that the active load balancer is handling the open connections.

Step 14 From the same client used for the long lived HTTP connections start a fifth client emulator. Verify that this connection fails with "Connection reset by peer" seen through the CLI. Stop the fifth client emulator.

Step 15 Return the primary and standby load balancers to their default FT configuration.

Step 16 These steps will test the effect of a load balancer module reload. Force a temporary FT failover to the standby load balancer by resetting the load balancer on the aggregation switch from the router prompt.

Step 17 Verify that after the load balancer reloads it preempts, becomes active and is handling the long lived connections.

Step 18 From the same client used for the long lived HTTP connections start a fifth client emulator. Verify that this connection fails with "Connection reset by peer" seen through the CLI. Stop the fifth client emulator.

Step 19 Verify that the current connection count on the serverfarm does NOT increase once the maxconn limit has been reached.

Step 20 Kill two of the long lived HTTP connections and then check the current connection count on the serverfarm.

Step 21 Kill one of the long lived HTTP connections. Check the current connection count on the serverfarm.

Step 22 From the same client used for the long lived HTTP connections start another client emulator. Verify that this connection is accepted as the minconn value of two which has been exceeded.

Step 23 Stop all client traffic.

Expected Results

The following test results are anticipated:

- We expect the load balancer to either load balance these new connections to another active server or reset the connection once the maxconn limit is exceeded.
- We expect this feature to work reliably after fault-tolerant failovers have occurred.
- We expect this feature not to allow new connections until the minconn limit has been passed after the maxconn was reached surpassed.

- We expect no CPU or memory problems.

Results

Maxconn Connection Limiter—CSM passed.

Traffic Handling

Traffic Handling tests verify that the load balancer module can manage various types of traffic.

This section contains the following topics:

FTP

Traffic Handling tests verify that the load balancer module can manage various types of traffic.

FTP (File Transfer Protocol) is a standard internet protocol used as a simple way to exchange files between computers on the internet. FTP uses IP addresses and port numbers embedded in the data portion of the control channel packets to determine the data connection.

This section contains the following topics:

Passive FTP—ACE

FTP (File Transfer Protocol) is a standard internet protocol used as a simple way to exchange files between computers on the internet. FTP uses IP addresses and port numbers embedded in the data portion of the control channel packets to determine the data connection. When this traffic is hitting a virtual address on the load balancer, it needs to have the class map configured with the **inspect ftp** command. This allows the load balancer to be application aware and make the proper translation to these embedded addresses and ports. Active (PORT) mode FTP is when the client opens up a data port and then the server initiates a connection to it, to transfer data. Passive (PASV) mode FTP is when the server opens up a data port and the client initiates a connection to it, to transfer data. This test verified proper load balancing of passive FTP traffic for small and large files with client NAT (Network Address Translation) configured.

Relevant ACE Configuration

```
serverfarm host FTP
  probe FTP
  rserver BRG-11 21
```

```
    inservice
  rserver BRG-12 21
    inservice
  rserver LOCAL-240 21
    inservice
  rserver LOCAL-241 21
    inservice
  rserver RT-251 21
    inservice
  rserver RT-252 21
    inservice
class-map match-all FTP-VIP-NAT_119
  2 match destination-address 192.168.120.119 255.255.255.255
class-map match-all FTP-VIP_119:1111
  2 match virtual-address 192.168.120.119 tcp eq 1111
policy-map type loadbalance first-match FTP-LB-SF_FTP
  class class-default
    serverfarm FTP
policy-map multi-match SH-Gold-VIPs
  class FTP-VIP_119:1111
    loadbalance vip inservice
    loadbalance policy FTP-LB- SF_FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class FTP-VIP-NAT_119
    nat dynamic 1 VLAN 120
    nat dynamic 1 vlan 29
interface vlan 29
  ip address 172.29.0.1 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  nat-pool 1 192.168.120.71 192.168.120.71 netmask 255.255.255.0 pat
  no shutdown
interface vlan 120
  description Upstream VLAN_120—Clients and VIPs
  ip address 192.168.120.1 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.70 192.168.120.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  no shutdown
ip route 10.1.0.0 255.255.255.0 192.168.120.254
ip route 172.28.0.0 255.252.0.0 172.29.0.253
```

Test Procedure

The procedure used to perform the Passive FTP—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the service policy and serverfarm counters on the load balancer.

Step 3 Start a packet capture on the load balancer's internal interface.

Step 4 Start a series of passive FTP requests, on a client that has to route to the VIP and on client that is L2 adjacent.

Step 5 Verify that NAT entries are created on the load balancer.

Step 6 Stop the packet capture. Verify on the control connection that the PASV command was used and that the client initiated the data connection back to the server.

Step 7 When traffic has completed, verify that no errors are displayed.

Step 8 Verify that the load balancer has the expected hit count and that each server has the expected number of connections.

Step 9 Start a series of passive FTP requests, on a client that has to route to the VIP and on client that is L2 adjacent.

Step 10 When traffic has completed, verify that no errors are displayed.

Step 11 Verify that the load balancer has the expected hit count and that each server has the expected number of connections.

Expected Results

The following test results are anticipated:

- We expect the ACE to properly load balance passive FTP requests for small and large files.
- We expect the ACE to allow an FTP control channel connection to remain idle for at least 40 minutes without issuing a reset.
- We expect the ACE to source NAT the client connections hitting the FTP virtual IP.
- We expect that the load balancer will not crash or become unresponsive.

Results

Passive FTP—ACE passed.

Passive FTP—CSM

FTP (File Transfer Protocol) is a standard internet protocol used as a simple way to exchange files between computers on the internet. FTP uses IP addresses and port numbers embedded in the data portion of the control channel packets to determine the data connection. When this traffic is hitting a virtual address on the load balancer, it needs to have the class map configured with the **inspect ftp** command. This allows the load balancer to be application aware and make the proper translation to these embedded addresses and ports. Active (PORT) mode FTP is when the client opens up a data port and then the server initiates a connection to it, to transfer data. Passive (PASV) mode FTP is when the server opens up a data port and the client initiates a connection to it, to transfer data. This test verified proper load balancing of passive FTP traffic for small and large files with client NAT (Network Address Translation) configured.

Relevant ACE Configuration

```
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 serverfarm FTP
  nat server
  nat client CLIENT_NAT
  real name RT-IIS-152 21
   inservice
  real name BRG-LINUX-11 21
   inservice
  real name LOCAL-IIS-245 21
   inservice
  probe FTP
```

```
!
 vserver FTP
  virtual 192.168.120.219 tcp 1111 service ftp
  serverfarm FTP
  no persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Passive FTP—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Start a packet capture on the load balancer's internal interface.

Step 4 Start a series of passive FTP requests, on a client that has to route to the VIP and on client that is L2 adjacent.

Step 5 Verify that NAT entries are created on the load balancer.

Step 6 Stop the packet capture. Verify on the control connection that the PASV command was used and that the client initiated the data connection back to the server.

Step 7 When traffic has completed, verify that no errors are displayed.

Step 8 Verify that the load balancer displays the correct values for the FTP traffic.

Step 9 Start a series of passive FTP requests, on a client that has to route to the VIP and on client that is L2 adjacent.

Step 10 When traffic has completed, verify that no errors are displayed.

Step 11 Verify that the load balancer displays the correct values for the FTP traffic.

Expected Results

The following test results are anticipated:

- We expect the ACE to properly load balance passive FTP requests for small and large files.
- We expect the ACE to allow an FTP control channel connection to remain idle for at least 40 minutes without issuing a reset.
- We expect the ACE to source NAT the client connections hitting the FTP virtual IP.
- We expect that the load balancer will not crash or become unresponsive.

Results

Passive FTP—CSM passed.

Insert

Traffic Handling tests verify that the load balancer module can manage various types of traffic.

The insert feature allows the load balancer module to insert information into a packet. For example, with the cookie insert feature the load balancer inserts the cookie in the server response to the client. With the header insert feature the load balancer inserts information, such as the client IP, destination IP, custom, and other types into the HTTP header. Both cookie and header inserts are covered by this category of testing.

This section contains the following topics:

- Cookie Insert—ACE, page 120
- Cookie Insert—CSM, page 122
- Header Insert—ACE, page 125
- Header Insert—CSM, page 128

Cookie Insert—ACE

The cookie insert feature is used when you want to use a session cookie for persistence if the server is not currently setting the appropriate cookie. With this feature enabled, the Load Balancer inserts the cookie in the server response to the client.

This test verified that Load Balancer performs cookie insert properly by checking the following items. The Load Balancer inserted the configured cookie and maintained sticky when tested with different browsers. A cookie was inserted when HTTP GETs and POSTs were sent from a client using a very small MTU (Maximum Transmission Unit), which caused these requests to span many packets.

Relevant Load Balancer Configuration

```
parameter-map type http COOKIE-INSERT-HDR-PARSE
  set header-maxparse-length 4000
  persistence-rebalance
serverfarm host COOKIE-INSERT
  probe HTTP-PROBE
  rserver BRG-14
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-152
    inservice
serverfarm host COOKIE-HASH
  rserver LOCAL-240
    inservice
  rserver LOCAL-240 90
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-241 90
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-242 90
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-243 90
    inservice
sticky http-cookie COOKIE_INSERT COOKIE-INSERT-GROUP-45
  cookie insert
  timeout 30
  serverfarm COOKIE-HASH
```

```
sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes
COOKIE-INSERT-GROUP-46
  cookie insert
  timeout 1
  serverfarm COOKIE-INSERT
sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes
COOKIE-INS2-GROUP-46
  cookie insert
  timeout 1
  serverfarm COOKIE-INSERT
class-map type http loadbalance match-any P-COOKIE-INS
  2 match http url /index.html* method GET
class-map type http loadbalance match-any P-COOKIE-INS2
  2 match http url .*
class-map match-all COOKIE-INSERT-VIP_118:80
  2 match virtual-address 192.168.120.118 tcp eq www
class-map match-all COOKIE-INS2-VIP_118:8888
  2 match virtual-address 192.168.120.118 tcp eq 8888
class-map match-all COOKIE-HASH-VIP_10.20.30.40:80
  2 match virtual-address 10.20.30.40 tcp eq www
policy-map type loadbalance first-match PLBSF_COOKIE-INSERT
  class P-COOKIE-INS
    insert-http Source-IP header-value %is
    insert-http Destination_IP header-value %id
    sticky-serverfarm COOKIE-INSERT-GROUP-46
  class P-COOKIE-INS2
    insert-http Source-IP header-value %is
    insert-http Destination_IP header-value %id
    sticky-serverfarm COOKIE-INSERT-GROUP-46
policy-map type loadbalance first-match PLBSF_COOKIE-INS2
  class class-default
    sticky-serverfarm COOKIE-INS2-GROUP-46
policy-map multi-match SH-Gold-VIPs
  class COOKIE-HASH-VIP_10.20.30.40:80
    loadbalance policy PLBSF_COOKIE-HASH
    loadbalance vip inservice
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class COOKIE-INSERT-VIP_118:80
    appl-parameter http advanced-options COOKIE-INSERT-HDR-PARSE
    loadbalance policy PLBSF_COOKIE-INSERT
    loadbalance vip inservice
    loadbalance vip icmp-reply active
    nat dynamic 1 VLAN 120
  class COOKIE-INS2-VIP_118:8888
    appl-parameter http advanced-options COOKIE-INSERT-HDR-PARSE
    loadbalance policy PLBSF_COOKIE-INS2
    loadbalance vip inservice
    loadbalance vip icmp-reply active
    nat dynamic 1 VLAN 120
```

Test Procedure

The procedure used to perform the Cookie Insert—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the Load Balancer.

Step 3 On a Windows client set the MTU to 150 bytes to simulate a dial-up or satellite user. Start a packet capture for the Windows client traffic.

Step 4 Send a series of HTTP GET requests from a Internet Explorer (IE) 7.0 browser, hitting CTRL refresh after the first page. Verify on the load balancer that one server serviced this request.

Step 5 Stop the capture. Verify that the first request was sent without a cookie and the load balancer inserted a cookie. Verify that the remaining GETs include the cookie learned from the load balancer and all the cookies inserted by the load balancer are the same.

Step 6 Send a series of HTTP GET request from a FireFox browser hitting CTRL refresh after the first page. Verify on the load balancer that one server serviced this request.

Step 7 Stop the capture. Verify that the first request was sent without a cookie and the load balancer inserted a cookie. Verify that the remaining GETs include the cookie learned from the load balancer and that all the cookies inserted by the load balancer are the same.

Step 8 On a Windows client set the MTU back to its original setting.

Step 9 On a Linux client, set the interface to a MTU of 150 to simulate a dial-up or satellite user. Start a packet capture on the internal port-channel for the load balancer or run ethereal from the client.

Step 10 Send a series of POST requests. These requests have a small POST header spanning several packets, while the POST body spans many.

Step 11 Stop the packet capture. Parse the output to verify the load balancer inserted a unique cookie for the POST request. Also verify that there are no TCP retransmissions, which would indicate packet loss.

Step 12 Start another packet capture. Send a series of POST requests. These requests have a large POST header and body spanning many packets.

Step 13 Stop the packet capture. Parse the output to verify the load balancer inserted a unique cookie for the POST request. Also verify that there are no TCP retransmissions, which would indicate packet loss.

Step 14 Set the mtu of the client back to 1500.

Expected Results

The following test results are anticipated:

- We expect the Load Balancer to insert the configured headers and cookies for GETs and POSTs from browser requests and client tools.
- We expect that when the MTU is set low, causing the requests to span multiple packets will not impact testing.
- We expect that a unique cookie will be in the sticky database for the same servers in the same serverfarm, but listening on different ports.
- We expect the Load Balancer to set an appropriate expiration timer based upon the configuration.
- We expect that the load balancer will not crash or become unresponsive.

Results

Cookie Insert—ACE passed.

Cookie Insert—CSM

The cookie insert feature is used when you want to use a session cookie for persistence if the server is not currently setting the appropriate cookie. With this feature enabled, the Load Balancer inserts the cookie in the server response to the client.

This test verified that Load Balancer performs cookie insert properly by checking the following items. The Load Balancer inserted the configured cookie and maintained sticky when tested with different browsers. A cookie was inserted when HTTP GETs and POSTs were sent from a client using a very small MTU (Maximum Transmission Unit), which caused these requests to span many packets.

Relevant Load Balancer Configuration

```
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 map INDEX.HTML url
  match protocol http method GET url /index.html*
!
 map M-HDR-SRCDST-IP header
  insert protocol http header Source-IP header-value %is
  insert protocol http header Destination_IP header-value %id
!
 serverfarm COOKIE-HASH
  nat server
  nat client CLIENT_NAT
  real 10.96.130.18
   inservice
  real 10.96.130.19
   inservice
  real 10.96.130.26
   inservice
  real 10.96.130.32
   inservice
  real 10.96.130.18 90
   inservice
  real 10.96.130.19 90
   inservice
  real 10.96.130.26 90
   inservice
  real 10.96.130.32 90
   inservice
!
 serverfarm COOKIE-INSERT
  nat server
  no nat client
  real name RT-IIS-152
   inservice
  real name BRG-LINUX-11
   inservice
  real name LOCAL-LINUX-242
   inservice
  probe HTTP-PROBE
!
 sticky 45 cookie Sticky_Cookie_Group_45 insert timeout 30
!
 sticky 46 cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes insert
timeout 1
!
 policy P-COOKIE-INS
  url-map INDEX.HTML
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 46
  serverfarm COOKIE-INSERT
!
 policy P-COOKIE-INS2
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 46
  serverfarm COOKIE-INSERT
!
 vserver COOKIE-HASH
  virtual 10.20.30.40 tcp www
  serverfarm COOKIE-HASH
  sticky 30 group 45
  persistent rebalance
  inservice
!
 vserver COOKIE-INSERT
  virtual 192.168.120.233 tcp www
  persistent rebalance
  parse-length 4000
  slb-policy P-COOKIE-INS
  slb-policy P-COOKIE-INS2
  inservice
!
```

Test Procedure

The procedure used to perform the Cookie Insert—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the Load Balancer.

Step 3 On a Windows client set the MTU to 150 bytes to simulate a dial-up or satellite user. Start a packet capture for the Windows client traffic.

Step 4 Send a series of HTTP GET requests from a Internet Explorer (IE) browser, hitting CTRL refresh after the first page. Verify on the load balancer that one server serviced this request.

Step 5 Stop the capture. Verify that the first request was sent without a cookie and the load balancer inserted a cookie. Verify that the remaining GETs include the cookie learned from the load balancer and all the cookies inserted by the load balancer are the same.

Step 6 Send a series of HTTP GET request from a FireFox browser hitting CTRL refresh after the first page. Verify on the load balancer that one server serviced this request.

Step 7 Stop the capture. Verify that the first request was sent without a cookie and the load balancer inserted a cookie. Verify that the remaining GETs include the cookie learned from the load balancer and that all the cookies inserted by the load balancer are the same.

Step 8 On a Windows client set the MTU back to its original setting.

Step 9 On a Linux client, set the interface to a MTU of 150 to simulate a dial-up or satellite user. Start a packet capture on the internal port-channel for the load balancer or run ethereal from the client.

Step 10 Send a series of POST requests. These requests have a small POST header spanning several packets, while the POST body spans many.

Step 11 Stop the packet capture. Parse the output to verify the load balancer inserted a unique cookie for the POST request. Also verify that there are no TCP retransmissions, which would indicate packet loss.

Step 12 Start another packet capture. Send a series of POST requests. These requests have a large POST header and body spanning many packets.

Step 13 Stop the packet capture. Parse the output to verify the load balancer inserted a unique cookie for the POST request. Also verify that there are no TCP retransmissions, which would indicate packet loss.

Step 14 Set the mtu of the client back to 1500.

Expected Results

The following test results are anticipated:

- We expect the Load Balancer to insert the configured headers and cookies for GETs and POSTs from browser requests and client tools.
- We expect that when the MTU is set low, causing the requests to span multiple packets will not impact testing.
- We expect that the load balancer will not crash or become unresponsive.

Results

Cookie Insert—CSM passed.

Header Insert—ACE

The HTTP header insert feature provides the Load Balancer with the ability to insert information, such as the client IP, destination IP, custom, and other types into the HTTP header. This feature is useful in situations where the Load Balancer is performing source NAT and the server application requires visibility to the original source IP. The Load Balancer performs the header insert function in the client-to-server direction. This test verified that Load Balancer inserted the configured headers when GETs and POSTs were issued, even when these requests spanned many packets.

Relevant Load Balancer Configuration

```
parameter-map type http PERSIST-INSERT
  header modify per-request
parameter-map type http PERSIST-REBAL-4K
  persistence-rebalance
  set header-maxparse-length 4096
serverfarm host HDR-IXIA
  rserver BRG-14
    inservice
  rserver LOCAL-241
    inservice
serverfarm host HEADER-INSERT
  rserver BRG-13
    inservice
  rserver RT-151
    inservice
  rserver RT-153
    inservice
serverfarm host HEADER-INSERT2
  rserver BRG-12
    inservice
  rserver BRG-14
    inservice
  rserver LOCAL-240
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-153
    inservice
  rserver RT-154
    inservice
class-map match-all HDR-IXIA-VIP_123:80
  match virtual-address 192.168.120.123 tcp eq www
class-map match-all HEADER-INSERT-VIP_121:80
  match virtual-address 192.168.120.121 tcp eq www
class-map match-all HEADER-INSERT2-VIP_122:80
  match virtual-address 192.168.120.122 tcp eq www
class-map type http loadbalance match-all P-HDR-INSERT
   2 match http url .*
class-map type http loadbalance match-all P-HDR-IXIA
   2 match http url .*
class-map type http loadbalance match-all P-HDR-SRCDST-IP
   2 match http url .*
policy-map type loadbalance first-match PLBSF_HDR-IXIA
  class P-HDR-IXIA
    serverfarm HDR-IXIA
policy-map type loadbalance first-match PLBSF_HEADER-INSERT
  class P-HDR-INSERT
    serverfarm HEADER-INSERT
    insert-http Custom-header_name_size_100bytes_abcdefghijklmnopqrstuvwxyz-0123
4567890_ABCDEFGHIJKLMNOPQRS_100BYTES header-value "Size of inserted header value
 is 100 bytes abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHI_100BYTES"
    insert-http Destination_iP header-value "%id"
    insert-http Pragma header-value "Pragma no Pragma that is the question"
    insert-http Accept header-value "anything"
    insert-http Source-IP header-value "%is"
policy-map type loadbalance first-match PLBSF_HEADER-INSERT2
  class P-HDR-SRCDST-IP
    serverfarm HEADER-INSERT2
    insert-http Destination_iP header-value "%id"
    insert-http Source-IP header-value "%is"
policy-map multi-match SH-Gold-VIPs
    class HEADER-INSERT-VIP_121:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class HDR-IXIA-VIP_123:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HDR-IXIA
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class HEADER-INSERT2-VIP_122:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
```

Test Procedure

The procedure used to perform the Header Insert —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Take selected servers out of service in select serverfarms, and clear HTTP stats.

Step 3 On the aggregation switch, use the NAM to start a packet capture on the Load Balancer's internal portchannel.

Step 4 On a Windows client, set the MTU (Maximum Transmission Unit) to 150 bytes to simulate a dial-up or satellite user. Use an Internet Explorer (IE) browser to issue a single request to select Virtual IP Addresses (VIPs).

Step 5 Stop the packet capture and view the results.

Step 6 Parse the same packet capture and isolate requests to VIP 192.168.120.122.

Step 7 Verify that headers were inserted and that no errors were seen.

Step 8 Restart the packet capture on the Load Balancer's internal portchannel.

Step 9 On a Windows client, use a Firefox browser to issue a single request to select VIPs.

Step 10 Stop the packet capture and view the results.

Step 11 Parse the same packet capture and isolate select requests.

Step 12 Verify that headers were inserted and that no errors were seen.

Step 13 On the windows client, set the MTU back to its original setting.

Step 14 Restart the packet capture on the Load Balancer's internal portchannel.

Step 15 On a Linux client, set the interface to an MTU of 100 to simulate a dial-up or satellite user. Send a short series of HTTP POST requests to select VIPs.

Step 16 Stop the packet capture and view the results.

Step 17 Parse the same packet capture and isolate select requests.

Step 18 Verify that headers were inserted and that no errors were seen.

Step 19 Modify the class to include two additional headers.

Step 20 Start a packet capture on the Load Balancer's internal portchannel.

Step 21 On a Linux client, send a short series of POST and GET requests.

Step 22 Stop the packet capture and view the results.

Step 23 Modify the class to remove one of the additional headers.

Step 24 Start a packet capture on the Load Balancer's internal portchannel.

Step 25 On a Linux client, send a short series of HTTP POST and GET requests.

Step 26 Stop the packet capture and view the results.

Step 27 Modify the class to remove the remaining additional header.

Step 28 These steps will test the ability of the load balancer to insert headers in all requests on a persistent connection. This happens by default with persistence-rebalance enabled. Start a packet capture on the load balancer's internal portchannel.

Step 29 On a Linux client start two long-lived flows to issue a continuous series of POST and GET requests to 192.168.120.121.

Step 30 Let the traffic run for a minimum of 30s. Then stop the client traffic.

Step 31 Stop the packet capture and view the results. Verify that the configured headers are inserted on the server side of the trace for all the GET and POST requests.

Step 32 Remove the parameter-map that the load balancer is using to allow header-insert to occur for each request.

Step 33 Start a packet capture on the load balancer's internal portchannel and then launch the client traffic.

Step 34 Let the traffic run for a minimum of 30s. Then stop the traffic and packet capture. View the packet capture results. Verify that the configured headers are inserted on the server side of the trace only for the first GET or POST, while subsequent requests do contain these headers.

Step 35 Verify that the there are no reported header insert errors and then clear the counters.

Step 36 Restore persistent-rebalance parameter map.

Step 37 On several Linux clients launch a continuous stream of traffic to the vservers 192.168.120.121 and 192.168.120.122.

Step 38 Let the test traffic run for 20 minutes while periodically gathering stats. After the time has passed stop the client traffic and verify that there are no header insert errors.

Step 39 Start a packet capture on the load balancer's internal portchannel.

Step 40 On a Linux client, send a short series of HTTP POST and GET requests to 192.168.120.121.

Step 41 Stop the packet capture and view the results. Verify that the configured headers are inserted on the server side of the trace for all the GET and POST requests.

Step 42 Set client mtu back to 1500.

Step 43 On the primary load balancer, bring selected servers back inservice.

Expected Results

The following test results are anticipated:

- We expect the ACE to properly insert the configured headers.
- We expect the ACE to properly insert the configured headers when a small MTU is used causing the requests to span multiple packets.
- We expect that the load balancer will not crash or become unresponsive.

Results

Header Insert —ACE passed.

Header Insert—CSM

The HTTP header insert feature provides the Load Balancer with the ability to insert information, such as the client IP, destination IP, custom, and other types into the HTTP header. This feature is useful in situations where the Load Balancer is performing source NAT and the server application requires visibility to the original source IP. The Load Balancer performs the header insert function in the client-to-server direction. This test verified that Load Balancer inserted the configured headers when GETs and POSTs were issued, even when these requests spanned many packets.

Relevant Load Balancer Configuration

```
!
 map M-HDR-INSERT header
  insert protocol http header Source-IP header-value %is
  insert protocol http header Accept header-value anything
  insert protocol http header Pragma header-value "Pragma no Pragma that is the question"
  insert protocol http header Destination_IP header-value %id
```

```
!
 map M-HDR-SRCDST-IP header
  insert protocol http header Source-IP header-value %is
  insert protocol http header Destination_IP header-value %id
!
  serverfarm HEADER-INSERT
  nat server
  no nat client
  real name RT-LINUX-153
   inservice
  real name BRG-LINUX-14
   inservice
  real name LOCAL-IIS-245
   inservice
  probe HTTP-PROBE
!
 serverfarm HEADER-INSERT2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-IIS-241
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   inservice
  probe HTTP-PROBE
!
 policy P-HDR-INSERT
  nat client CLIENT_NAT
  header-map M-HDR-INSERT
  sticky-group 42
  serverfarm HEADER-INSERT
!
 policy P-HDR-SRCDST-IP
  header-map M-HDR-SRCDST-IP
  sticky-group 43
  serverfarm HEADER-INSERT2
!
 vserver HEADER-INSERT
  virtual 192.168.120.231 tcp www
  serverfarm DEFAULT
  persistent rebalance
  parse-length 3000
  slb-policy P-HDR-INSERT
  inservice
!
 vserver HEADER-INSERT2
  virtual 192.168.120.232 tcp www
  persistent rebalance
  parse-length 3000
  slb-policy P-HDR-SRCDST-IP
  inservice
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
```

```
!
```

Test Procedure

The procedure used to perform the Header Insert—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the aggregation switch, use the NAM to start a packet capture on the Load Balancer's internal portchannel.

Step 3 On a Windows client, set the MTU (Maximum Transmission Unit) to 150 bytes to simulate a dial-up or satellite user. Use an Internet Explorer (IE) browser to issue a single request to a select Virtual IP Addresses (VIPs).

Step 4 Parse the same packet capture and isolate requests to a specific VIP.

Step 5 Parse the same packet capture and isolate requests to a specific VIP.

Step 6 Verify that headers were inserted and that no errors were seen.

Step 7 Restart the packet capture on the Load Balancer's internal portchannel.

Step 8 On a Windows client, use a Firefox browser to issue a single request to select VIPs.

Step 9 Stop the packet capture and view the results.

Step 10 Parse the same packet capture and isolate select requests.

Step 11 Verify that headers were inserted and that no errors were seen.

Step 12 On the windows client, set the MTU back to its original setting.

Step 13 Restart the packet capture on the Load Balancer's internal portchannel.

Step 14 On a Linux client, set the interface to an MTU of 100 to simulate a dial-up or satellite user. Send a short series of HTTP POST requests to select VIPs.

Step 15 Stop the packet capture and view the results.

Step 16 Parse the same packet capture and isolate select requests.

Step 17 Verify that headers were inserted and that no errors were seen.

Step 18 Modify the class map to include two additional headers.

Step 19 Start a packet capture on the Load Balancer's internal portchannel.

Step 20 On a Linux client, send a short series of POST and GET requests.

Step 21 Stop the packet capture and view the results.

Step 22 Modify the class to remove one of the additional headers.

Step 23 Start a packet capture on the Load Balancer's internal portchannel.

Step 24 On a Linux client, send a short series of HTTP POST and GET requests.

Step 25 Stop the packet capture and view the results.

Step 26 Modify the class to remove the remaining additional header.

Step 27 On the client set the MTU back to its original setting.

Step 28 Clear select counters on the load balancer.

Step 29 On a Linux client generate a combined load of approximately 100 connections per second to select Virtual IP Addresses (VIPs).

Step 30 Start a packet capture on the load balancer's internal portchannel.

Step 31 Parse the capture output looking for any IP or TCP checksum errors. There should be none.

Step 32 Clear the load balancer's ARP cache and verify that it has been cleared.

Step 33 Let the test traffic finish to completion and verify that the active and standby load balancers are responsive.

Step 34 Verify that the load balancer did not see any IP or TCP checksum errors and that the ARP table has been repopulated.

Expected Results

The following test results are anticipated:

- We expect the CSM to insert the configured headers for GETS and POSTs from browser requests and client tools.
- We expect the CSM to insert the configured headers for GETS and POSTs from browser requests and client tools when the MTU is set low causing the requests to span multiple packets.
- We expect the CSM to handle a moderate traffic load of cookie and header insert without any packet corruption.
- We expect that the load balancer will not crash or become unresponsive.

Results

Header Insert—CSM passed.

Maps

Traffic Handling tests verify that the load balancer module can manage various types of traffic.

Class map HTTP inspection configuration mode commands allow you to create a Layer 7 HTTP deep packet inspection class map. In this category of testing, cookie maps, header maps and url maps are verified.

This section contains the following topics:

Cookie Map—ACE

Cookies are unique strings that are assigned by a server to a client in response to an HTTP request. This unique string is contained within subsequent requests so the client can be identified. The load balancer can parse these unique strings out of HTTP requests in order to send a client's request to the same serverfarm time after time.

This test verified the ability of the load balancer to identify cookies in an HTTP header or within an URL and direct the connection to the appropriate serverfarm.

Relevant Load Balancer Configuration

```
parameter-map type http COOKIE-DELIM
  persistence-rebalance
  set secondary-cookie-delimiters @$
serverfarm host COOKIE
  probe ICMP
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host COOKIE1
  probe HTTP
  rserver BRG-12
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-154
    inservice
serverfarm host COOKIE2
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-80
  probe TCP
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
class-map match-all COOKIE-MAP-VIP_124:80
  2 match virtual-address 192.168.120.124 tcp eq www
class-map type http loadbalance match-all COOKIE-MAP:80
  2 match http cookie COOKIE_TEST cookie-value "This is a test0"
class-map type http loadbalance match-all URLCOOKIE-MAP1
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE1"
class-map type http loadbalance match-all URLCOOKIE-MAP2
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE2"
policy-map type loadbalance first-match PLBSF_COOKIE-MAP
  class URLCOOKIE-MAP2
    serverfarm COOKIE2
  class URLCOOKIE-MAP1
    serverfarm COOKIE1
  class COOKIE-MAP:80
    serverfarm COOKIE
  class class-default
```

```
    serverfarm GEN-80
policy-map multi-match SH-Gold-VIPs
  class COOKIE-MAP-VIP_124:80
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-MAP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-DELIM
```

Test Procedure

The procedure used to perform the Cookie Map—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Start a series of HTTP 1.0 requests sending a cookie in the request.

Step 4 Verify that all the requests are hitting the default policy and expected serverfarm.

Step 5 Clear select counters on the load balancer.

Step 6 Generate a series of HTTP 1.0 and 1.1 URL requests with some connections sending a cookie embedded in the request and others with none.

Step 7 Verify that all the requests hit the expected serverfarms based on the policy match.

Expected Results

The following test results are anticipated:

- We expect the proper serverfarm to get the connections depending on the order in which the cookies are sent and mapped.
- We expect that the load balancer will not crash or become unresponsive.

Results

Cookie Map—ACE passed.

Cookie Map—CSM

Cookies are unique strings that are assigned by a server to a client in response to an HTTP request. This unique string is contained within subsequent requests so the client can be identified. The load balancer can parse these unique strings out of HTTP requests in order to send a client’s request to the same serverfarm time after time.

This test verified the ability of the load balancer to identify cookies in an HTTP header or within an URL and direct the connection to the appropriate serverfarm.

Relevant Load Balancer Configuration

```
!
 map COOKIE-MAP cookie
  match protocol http cookie CSM_TEST cookie-value This*is*a*test0
!
```

```
 serverfarm COOKIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm CS-COOKIES
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-154
   inservice
!
 policy COOKIE-POLICY
  cookie-map COOKIE-MAP
  serverfarm CS-COOKIES
!
 vserver COOKIE
  virtual 192.168.120.215 tcp www
  serverfarm COOKIE
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the Cookie Map—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer and verify that no cookie policies are in use.

Step 3 Start a series of HTTP 1.0 requests sending a cookie in the request.

Step 4 Verify that all the requests are hitting the default policy and expected serverfarm.

Step 5 Clear select counters on the load balancer.

Step 6 Configure the load balancer to use a cookie policy.

Step 7 Verify the status of the cookie policy.

Step 8 Generate a series of HTTP 1.0 and 1.1 URL requests with some connections sending a cookie imbedded in the request and others with none

Step 9 Verify that all the requests hit the expected serverfarms based on the policy match

Step 10 Remove the configured policy, no slb-policy cookie-policy.

Expected Results

The following test results are anticipated:

- We expect the proper serverfarm to get the connections depending on the order in which the cookies are sent and mapped.
- We expect that the load balancer will not crash or become unresponsive.

Results

Cookie Map—CSM passed.

Header Map—ACE

The load balancer supports generic HTTP request header parsing. The HTTP request header contains fields that describe how content should be formatted to meet the user's requirements. For example, by parsing the browser-type field in the HTTP header, the load balancer can determine if a user is accessing the content with a mobile browser and can select a server that contains content formatted for a mobile browser. This test will send multiple HTTP requests with varying values in certain header fields and verify that the load balancer parses those correctly, based on the header map, and forwards them to the correct server.

Relevant ACE Configuration

```
serverfarm host CS-MOZILLA
  rserver LOCAL-240
    inservice
  rserver RT-151
    inservice
serverfarm host CS-MSIE
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
serverfarm host HEADER
  rserver LOCAL-240
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
class-map type http loadbalance match-all BROWSER_FIREFOX
  2 match http header User-Agent header-value ".*Firefox.*"
class-map type http loadbalance match-all BROWSER_MOZILLA
  2 match http header User-Agent header-value ".*Mozilla.*"
class-map type http loadbalance match-all BROWSER_MOZILLA40
  2 match http header User-Agent header-value ".*Mozilla/4.0.*"
class-map type http loadbalance match-all BROWSER_MOZILLA50
  2 match http header User-Agent header-value ".*Mozilla/5.0.*"
class-map type http loadbalance match-all BROWSER_MSIE
  2 match http header User-Agent header-value ".*MSIE.*"
class-map match-all HEADER-VIP_125:80
  2 match virtual-address 192.168.120.125 tcp eq www
policy-map type loadbalance first-match PLBSF_HEADER
  class BROWSER_FIREFOX
    serverfarm CS-MOZILLA
  class BROWSER_MOZILLA40
    serverfarm CS-MSIE
  class BROWSER_MOZILLA50
    serverfarm CS-MOZILLA
  class BROWSER_MSIE
    serverfarm CS-MSIE
  class BROWSER_MOZILLA
    serverfarm CS-MOZILLA
  class class-default
    serverfarm HEADER
policy-map multi-match SH-Gold-VIPs2
  class HEADER-VIP_125:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER
    loadbalance vip icmp-reply
    appl-parameter http advanced-options PERSIST-REBALANCE
```

Test Procedure

The procedure used to perform the Header Map —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Show the status of the serverfarms.

Step 4 Request an URL from the VIP using the user-agent string "MSIE".

Step 5 Verify that the correct serverfarm was hit,

Step 6 Request an URL from the VIP using the user-agent string "Mozilla".

Step 7 Verify that the correct serverfarm was hit.

Step 8 Clear select counters on the load balancer. Start a stream of traffic with clients requesting URLs that hit both configured class maps along with some that do not match and fall through to the default serverfarm.

Step 9 Verify that the correct serverfarms are hit.

Expected Results

The following test results are anticipated:

- We expect the correct serverfarm to be used based on the data in the client’s HTTP request.
- We expect that the load balancer will not crash or become unresponsive.

Results

Header Map —ACE passed.

Header Map—CSM

The load balancer supports generic HTTP request header parsing. The HTTP request header contains fields that describe how content should be formatted to meet the user’s requirements. For example, by parsing the browser-type field in the HTTP header, the load balancer can determine if a user is accessing the content with a mobile browser and can select a server that contains content formatted for a mobile browser. This test will send multiple HTTP requests with varying values in certain header fields and verify that the load balancer parses those correctly, based on the header map, and forwards them to the correct server.

Relevant ACE Configuration

```
 map BROWSER_MOZILLA header
  match protocol http header User-Agent header-value *Mozilla*
!
 map BROWSER_MSIE header
  match protocol http header User-Agent header-value *MSIE*
!
 serverfarm CS-MOZILLA
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm CS-MSIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm HEADER
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
!
 policy BROWSER_MOZILLA
  header-map BROWSER_MOZILLA
  serverfarm CS-MOZILLA
!
 policy BROWSER_MSIE
  header-map BROWSER_MSIE
  serverfarm CS-MSIE
!
 vserver HEADER
  virtual 192.168.120.216 tcp www
  serverfarm HEADER
  persistent rebalance
  slb-policy BROWSER_MSIE
  slb-policy BROWSER_MOZILLA
  inservice
!
```

Test Procedure

The procedure used to perform the Header Map—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Show the status of the serverfarms.

Step 4 Request an URL from the VIP using the user-agent string "MSIE".

Step 5 Verify that the correct serverfarm was hit,

Step 6 Request an URL from the VIP using the user-agent string "Mozilla".

Step 7 Verify that the correct serverfarm was hit.

Step 8 Clear select counters on the load balancer. Start a stream of traffic with clients requesting URLs that hit both configured class maps along with some that do not match and fall through to the default serverfarm.

Step 9 Verify that the correct serverfarms are hit.

Expected Results

The following test results are anticipated:

- We expect the correct serverfarm to be used based on the data in the client’s HTTP request.
- We expect that the load balancer will not crash or become unresponsive.

Results

Header Map—CSM passed.

URL Map—ACE

URL maps are used to designate which serverfarm should handle incoming connections matching a particular URL string. In this way incoming HTTP traffic can be directed to separate server farms depending on which data is being requested. This test measures the capability of the load balancer to direct incoming HTTP requests to the appropriate server farm.

There were five different URL maps applicable to this test. Each matched a certain page provided by the real Linux servers (16k.htm, 32k.htm, 64k.htm, 128k.htm, and 512k.htm). Five individual serverfarms were created for the five policies. Each serverfarm had two real servers associated with it. The test was run several times, first with individual GET requests for the individual pages matched to the URL map, then with all pages associated with the five maps and others that will hit the default policy.

Relevant Load Balancer Configuration

```
serverfarm host URL-MAP-128K
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
serverfarm host URL-MAP-16K
  rserver BRG-12
    inservice
  rserver LOCAL-242
    inservice
serverfarm host URL-MAP-32K
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
serverfarm host URL-MAP-512K
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
```

```
    inservice
serverfarm host URL-MAP-64K
  rserver LOCAL-244
    inservice
  rserver LOCAL-242 91
    inservice
serverfarm host URL-MAPS
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-244
    inservice
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-11
  timeout 30
  serverfarm URL-MAP-16K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-13
  timeout 30
  serverfarm URL-MAP-64K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-12
  timeout 30
  serverfarm URL-MAP-32K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-14
  timeout 30
  serverfarm URL-MAP-128K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-15
  timeout 30
  serverfarm URL-MAP-512K
class-map type http loadbalance match-all 128K-FORWARDING
  match http url .*128k.*
class-map type http loadbalance match-all 16K-FORWARDING
  match http url .*16k.*
class-map type http loadbalance match-all 32K-FORWARDING
  match http url .*32k.*
class-map type http loadbalance match-all 512K-FORWARDING
  match http url .*512k.*
class-map type http loadbalance match-all 64K-FORWARDING
  match http url .*64k.*
class-map match-all URL-MAPS-VIP_130:80
  match virtual-address 192.168.120.130 tcp eq www
policy-map type loadbalance first-match PLBSF_URL-MAPS
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm URL-MAPS
policy-map multi-match SH-Gold-VIPs
  class URL-MAPS-VIP_130:80
    loadbalance vip inservice
    loadbalance policy PLBSF_URL-MAPS
    loadbalance vip icmp-reply active
    nat dynamic 1 VLAN 30
    appl-parameter http advanced-options PERSIST-REBALANCE
```

Test Procedure

The procedure used to perform the URL Map—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Send HTTP GET requests for the /16k.htm pages.

Step 4 Verify that the connection requests hit the expected serverfarm.

Step 5 Send HTTP GET requests for the /32k.htm pages.

Step 6 Verify that the connection requests hit the expected serverfarm.

Step 7 Send HTTP GET requests for the /64k.htm pages.

Step 8 Verify that the connection requests hit the expected serverfarm.

Step 9 Send HTTP GET requests for the /128k.htm pages.

Step 10 Verify that the connection requests hit the expected serverfarm.

Step 11 Send HTTP GET requests for the /512k.htm pages.

Step 12 Verify that the connection requests hit the expected serverfarm.

Step 13 Clear select counters on the serverfarms.

Step 14 Send HTTP GET requests for all of the matched pages, as well as non-matched pages.

Step 15 Verify that the connection requests are forwarded by the load balancer to the correct serverfarms.

Expected Results

The following test results are anticipated:

- We expect the load balancer to select the correct serverfarm when various URL’s are sent to the VIP.
- We expect that the load balancer will not crash or become unresponsive.

Results

URL Map—ACE passed.

URL Map—CSM

URL maps are used to designate which serverfarm should handle incoming connections matching a particular URL string. In this way incoming HTTP traffic can be directed to separate server farms depending on which data is being requested. This test measures the capability of the load balancer to direct incoming HTTP requests to the appropriate server farm.

There were five different URL maps applicable to this test. Each matched a certain page provided by the real Linux servers (16k.htm, 32k.htm, 64k.htm, 128k.htm, and 512k.htm). Five individual serverfarms were created for the five policies. Each serverfarm had two real servers associated with it. The test was run several times, first with individual GET requests for the individual pages matched to the URL map, then with all pages associated with the five maps and others that will hit the default policy.

Relevant ACE Configuration

```
!
 map 64K url
  match protocol http url *64k*
!
 map 16K url
  match protocol http url *16k*
!
 map 32K url
  match protocol http url *32k*
!
 map 128K url
  match protocol http url *128k*
!
 map 512K url
  match protocol http url *512k*
!
 policy 16K-FORWARDING
  url-map 16K
  sticky-group 11
  serverfarm URL-MAP-16K
!
 policy 32K-FORWARDING
  url-map 32K
  sticky-group 12
  serverfarm URL-MAP-32K
!
 policy 64K-FORWARDING
  url-map 64K
  sticky-group 13
  serverfarm URL-MAP-64K
!
 policy 128K-FORWARDING
  url-map 128K
  sticky-group 14
  serverfarm URL-MAP-128K
!
 policy 512K-FORWARDING
  url-map 512K
  sticky-group 15
  serverfarm URL-MAP-512K
!
 sticky 11 netmask 255.255.255.255 address both timeout 30
!
 sticky 12 netmask 255.255.255.255 address both timeout 30
!
 sticky 13 netmask 255.255.255.255 address both timeout 30
!
 sticky 14 netmask 255.255.255.255 address both timeout 30
!
 sticky 15 netmask 255.255.255.255 address both timeout 30
!
 vserver URL-MAPS
  virtual 192.168.120.214 tcp www
  serverfarm URL-MAPS
  persistent rebalance
  slb-policy 16K-FORWARDING
  slb-policy 32K-FORWARDING
  slb-policy 64K-FORWARDING
  slb-policy 128K-FORWARDING
  slb-policy 512K-FORWARDING
  inservice
```

```
!
```

Test Procedure

The procedure used to perform the URL Map—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Send HTTP GET requests for the /16k.htm pages.

Step 4 Verify that the connection requests hit the expected serverfarm.

Step 5 Send HTTP GET requests for the /32k.htm pages.

Step 6 Verify that the connection requests hit the expected serverfarm.

Step 7 Send HTTP GET requests for the /64k.htm pages.

Step 8 Verify that the connection requests hit the expected serverfarm.

Step 9 Send HTTP GET requests for the /128k.htm pages.

Step 10 Verify that the connection requests hit the expected serverfarm.

Step 11 Send HTTP GET requests for the /512k.htm pages.

Step 12 Verify that the connection requests hit the expected serverfarm.

Step 13 Clear select counters on the serverfarms.

Step 14 Send HTTP GET requests for all of the matched pages, as well as non-matched pages.

Step 15 Verify that the connection requests are forwarded by the load balancer to the correct serverfarms.

Expected Results

The following test results are anticipated:

- We expect the ACE to select the correct serverfarm when various URL’s are sent to the VIP.
- We expect that the load balancer will not crash or become unresponsive.

Results

URL Map—CSM passed.

Miscellaneous

Traffic Handling tests verify that the load balancer module can manage various types of traffic. Specific types of traffic covered by this category include FTP, inserts, maps, SSL, stickiness, L4 to L7 policy changes, UDP load balancing, persistence rebalance, pipeline, redir policies, URL lengths and TCP normalization.

In this category of testing, L4 to L7 policy changes, UDP load balancing, persistence rebalance, pipeline, redir policies, URL lengths and TCP normalization are verified.

This section contains the following topics:

Persistence Rebalance—ACE

The load balancer supports HTTP 1.1 persistence. This feature allows browsers to send multiple HTTP requests on a single persistent connection. After a persistent connection is established, the server keeps the connection open for a configurable interval to allow for more requests from the same client. This eliminates the overhead involved in establishing a new TCP connection for each request.

The load balancer allows HTTP connections to be switched based on a URL, cookies, or other fields contained in the HTTP header. Persistent connection support in the load balancer allows for each successive HTTP request in a persistent connection to be switched independently. As a new HTTP request arrives on an existing connection, it may be switched to the same server as the prior request, it may be switched to a different server, or it may be reset to the client preventing that request from being completed.

This test verified the behavior of the load balancer to maintain stickiness for connections established with the HTTP v1.1 protocol, which uses a single connection to issue multiple requests. Each request in the connection will match a different policy configured on the load balancer and will be paired with a server in different server farms from the other requests. This pairing should be maintained for the duration of the transaction.

Relevant Load Balancer Configuration

```
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
parameter-map type http PERSIST-REBALANCE
  persistence-rebalance
serverfarm host PERSISTENT
  rserver LOCAL-240
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver RT-154
    inservice
serverfarm host URL-MAP-128K
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
serverfarm host URL-MAP-16K
  rserver BRG-12
    inservice
  rserver LOCAL-242
    inservice
serverfarm host URL-MAP-32K
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
serverfarm host URL-MAP-512K
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
serverfarm host URL-MAP-64K
  rserver LOCAL-242 91
    inservice
  rserver LOCAL-244
    inservice
serverfarm host URL-MAPS
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-244
    inservice
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-11
  timeout 30
  serverfarm URL-MAP-16K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-13
  timeout 30
  serverfarm URL-MAP-64K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-12
  timeout 30
  serverfarm URL-MAP-32K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-14
  timeout 30
  serverfarm URL-MAP-128K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-15
  timeout 30
  serverfarm URL-MAP-512K
class-map type http loadbalance match-all 128K-FORWARDING
  2 match http url .*128k.*
class-map type http loadbalance match-all 16K-FORWARDING
  2 match http url .*16k.*
class-map type http loadbalance match-all 32K-FORWARDING
  2 match http url .*32k.*
class-map type http loadbalance match-all 512K-FORWARDING
  2 match http url .*512k.*
class-map type http loadbalance match-all 64K-FORWARDING
  2 match http url .*64k.*
class-map match-all PERSISTENT-VIP_131:80
  2 match virtual-address 192.168.120.131 tcp eq www
policy-map type loadbalance first-match PLBSF_PERSISTENT
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
```

```
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm PERSISTENT
policy-map multi-match SH-Gold-VIPs2
  class PERSISTENT-VIP_131:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PERSISTENT
    loadbalance vip icmp-reply active
    nat dynamic 1 VLAN 120
    appl-parameter http advanced-options PERSIST-REBALANCE
```

Test Procedure

The procedure used to perform the Persistence Rebalance—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that a policy-map is configured with the appropriate policies.

Step 3 Clear select counters on the load balancer.

Step 4 On the client, launch a test tool that opens up a single TCP connection, issuing a series of HTTP GET requests matching each policy configured.

Step 5 Verify that there is only a single TCP connection shown in the test tool results.

Step 6 Verify that there is only a single TCP connection shown in the total connections counter for a specific VIP and that the GET requests are being sent to the expected serverfarms resulting in a sequential staggered hit count (increments by 100 per policy).

Step 7 Verify that due to sticky only one server in each serverfarm gets all of the connections.

Step 8 Verify that the sticky database has been populated.

Step 9 Clear select counters on the load balancer.

Expected Results

The following test results are anticipated:

- We expect each GET request to match the correct policy.
- We expect sticky to be maintained as each GET is remapped to the correct policy.
- We expect that a policy match to continue to occur and count as a failure if the serverfarm is down.
- We expect that the load balancer will not crash or become unresponsive.

Results

Persistence Rebalance—ACE passed.

Persistence Rebalance—CSM

The load balancer supports HTTP 1.1 persistence. This feature allows browsers to send multiple HTTP requests on a single persistent connection. After a persistent connection is established, the server keeps the connection open for a configurable interval to allow for more requests from the same client. This eliminates the overhead involved in establishing a new TCP connection for each request.

The load balancer allows HTTP connections to be switched based on a URL, cookies, or other fields contained in the HTTP header. Persistent connection support in the load balancer allows for each successive HTTP request in a persistent connection to be switched independently. As a new HTTP request arrives on an existing connection, it may be switched to the same server as the prior request, it may be switched to a different server, or it may be reset to the client preventing that request from being completed.

This test verified the behavior of the load balancer to maintain stickiness for connections established with the HTTP v1.1 protocol, which uses a single connection to issue multiple requests. Each request in the connection will match a different policy configured on the load balancer and will be paired with a server in different server farms from the other requests. This pairing should be maintained for the duration of the transaction.

Relevant Load Balancer Configuration

```
!
 map 64K url
  match protocol http url *64k*
!
 map 16K url
  match protocol http url *16k*
!
 map 32K url
  match protocol http url *32k*
!
 map 128K url
  match protocol http url *128k*
!
 map 512K url
  match protocol http url *512k*
!
 serverfarm PERSISTENT
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm URL-MAP-128K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm URL-MAP-16K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm URL-MAP-32K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
!
 serverfarm URL-MAP-512K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm URL-MAP-64K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-154
   inservice
!
 sticky 11 netmask 255.255.255.255 address both timeout 30
 sticky 12 netmask 255.255.255.255 address both timeout 30
 sticky 13 netmask 255.255.255.255 address both timeout 30
 sticky 14 netmask 255.255.255.255 address both timeout 30
 sticky 15 netmask 255.255.255.255 address both timeout 30
!
 policy 16K-FORWARDING
  url-map 16K
  sticky-group 11
  serverfarm URL-MAP-16K
!
 policy 32K-FORWARDING
  url-map 32K
  sticky-group 12
  serverfarm URL-MAP-32K
!
 policy 64K-FORWARDING
  url-map 64K
  sticky-group 13
  serverfarm URL-MAP-64K
!
 policy 128K-FORWARDING
  url-map 128K
  sticky-group 14
  serverfarm URL-MAP-128K
!
 policy 512K-FORWARDING
  url-map 512K
  sticky-group 15
  serverfarm URL-MAP-512K
!
 vserver PERSISTENT
  virtual 192.168.120.224 tcp www
  serverfarm PERSISTENT
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  slb-policy 16K-FORWARDING
  slb-policy 32K-FORWARDING
  slb-policy 64K-FORWARDING
  slb-policy 128K-FORWARDING
  slb-policy 512K-FORWARDING
  inservice
!
```

Test Procedure

The procedure used to perform the Persistence Rebalance—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that a policy-map is configured with the appropriate policies.

Step 3 Clear select counters on the load balancer.

Step 4 On the client, launch a test tool that opens up a single TCP connection, issuing a series of HTTP GET requests matching each policy configured.

Step 5 Verify that there is only a single TCP connection shown in the test tool results.

Step 6 Verify that there is only a single TCP connection shown in the total connections counter for a specific VIP and that the GET requests are being sent to the expected serverfarms resulting in a sequential staggered hit count (increments by 100 per policy).

Step 7 Verify that due to sticky only one server in each serverfarm gets all of the connections.

Step 8 Verify that the sticky database has been populated.

Step 9 Clear select counters on the load balancer.

Expected Results

The following test results are anticipated:

- We expect each GET request to match the correct policy.
- We expect sticky to be maintained as each GET is remapped to the correct policy.
- We expect that a policy match to continue to occur and count as a failure if the serverfarm is down.
- We expect that the load balancer will not crash or become unresponsive.

Results

Persistence Rebalance—CSM passed.

Redirect Policy—ACE

Redirects are applied to incoming connections that are to be sent on to a server or server farm behind a "hidden" server. An incoming connection destined for server X is redirected to server Y, with a corresponding message being sent back to the requesting client.

A redirect URL policy was used to cause all incoming connections matching URLs ending in /redirect-1k.html, /redirect-10k.html and /redirect-100k.html to the appropriate serverfarms. These serverfarms are configured to redirect all incoming connections to the appropriate Virtual IP Address (VIP).

Relevant Load Balancer Configuration

```
rserver redirect REDIRECT-100K
  webhost-redirection http://192.168.120.132/redirect-100k.html 302
  inservice
rserver redirect REDIRECT-10K
  webhost-redirection http://192.168.120.133/redirect-10k.html 302
  inservice
rserver redirect REDIRECT-1K
  webhost-redirection http://192.168.120.134/redirect-1k.html 302
  inservice
serverfarm host RED-ALL-SVRS
  rserver LOCAL-240
    inservice
  rserver LOCAL-241
    inservice
serverfarm host REDIRECT
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
serverfarm redirect REDIRECT-100K
  rserver REDIRECT-100K
    inservice
serverfarm redirect REDIRECT-10K
  rserver REDIRECT-10K
    inservice
serverfarm redirect REDIRECT-1K
  rserver REDIRECT-1K
    inservice
class-map match-all RED-100K-VIP_132:80
  match virtual-address 192.168.120.132 tcp eq www
class-map match-all RED-10K-VIP_133:80
  match virtual-address 192.168.120.133 tcp eq www
class-map match-all RED-1K-VIP_134:80
  match virtual-address 192.168.120.134 tcp eq www
class-map match-all REDIRECT_135:80
  match virtual-address 192.168.120.135 tcp eq www
class-map type http loadbalance match-all REDIRECT-100K
  match http url .*redirect-100k.html
class-map type http loadbalance match-all REDIRECT-10K
  match http url .*redirect-10k.html
class-map type http loadbalance match-all REDIRECT-1K
  match http url .*redirect-1k.html
policy-map type loadbalance first-match PLBSF_REDIRECT
  class REDIRECT-1K
    serverfarm REDIRECT-1K
  class REDIRECT-10K
    serverfarm REDIRECT-10K
  class REDIRECT-100K
    serverfarm REDIRECT-100K
  class class-default
    serverfarm REDIRECT
```

```
policy-map type loadbalance first-match PLBSF_RED-ALL-SVRS
  class class-default
    serverfarm RED-ALL-SVRS
policy-map multi-match SH-Gold-VIPs2
  class RED-100K-VIP_132:80
    appl-parameter http advanced-options  PERSIST-REBALANCE
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip inservice
    loadbalance vip icmp-reply active
  class RED-10K-VIP_133:80
    appl-parameter http advanced-options  PERSIST-REBALANCE
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip inservice
    loadbalance vip icmp-reply active
  class RED-1K-VIP_134:80
    appl-parameter http advanced-options  PERSIST-REBALANCE
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip inservice
    loadbalance vip icmp-reply active
  class REDIRECT-VIP_135:80
    appl-parameter http advanced-options  PERSIST-REBALANCE
    loadbalance policy PLBSF_REDIRECT
    loadbalance vip inservice
    loadbalance vip icmp-reply active
```

Test Procedure

The procedure used to perform the Redirect Policy—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Issue an HTTP request on a client for /redirect-1k.html. Verify through the client output that the redirect was successful.

Step 4 Verify that the load balancer redirected the the traffic as expected.

Step 5 Clear select counters on the load balancer.

Step 6 Start a series of HTTP GET’s for multiple URLs, matching and not matching the redirect policies.

Step 7 Verify that the load balancer redirected the the traffic as expected.

Expected Results

The following test results are anticipated:

- We expect the load balancer to respond to the client with a 302 redirect for only the applicable traffic
- We expect that the load balancer will not crash or become unresponsive.

Results

Redirect Policy—ACE passed.

Redirect Policy—CSM

Redirects are applied to incoming connections that are to be sent on to a server or server farm behind a "hidden" server. An incoming connection destined for server X is redirected to server Y, with a corresponding message being sent back to the requesting client.

A redirect URL policy was used to cause all incoming connections matching URLs ending in /redirect-1k.html, /redirect-10k.html and /redirect-100k.html to the appropriate serverfarms. These serverfarms are configured to redirect all incoming connections to the appropriate Virtual IP Address (VIP).

Relevant Load Balancer Configuration

```
!
 map REDIRECT-1K url
  match protocol http url *redirect-1k.html
!
 map REDIRECT-10K url
  match protocol http url *redirect-10k.html
!
 map REDIRECT-100K url
  match protocol http url *redirect-100k.html
!
 real LOCAL-LINUX-244
  address 172.29.0.244
  inservice
 real RT-LINUX-154
  address 172.28.0.154
  inservice
 real LOCAL-LINUX-240
  address 172.29.0.240
  inservice
 real LOCAL-LINUX-242
  address 172.29.0.242
  inservice
 real RT-LINUX-153
  address 172.28.0.153
  inservice
!
 serverfarm REDIRECT-100K
  nat server
  no nat client
  redirect-vserver REDIRECT-100K
   webhost relocation 192.168.120.220/redirect-100k.html
   inservice
!
 serverfarm REDIRECT-10K
  nat server
  no nat client
  redirect-vserver REDIRECT-10K
   webhost relocation 192.168.120.221/redirect-10k.html
   inservice
!
 serverfarm REDIRECT-1K
  nat server
  no nat client
  redirect-vserver REDIRECT-1K
   webhost relocation 192.168.120.222/redirect-1k.html
   inservice
!
 serverfarm RED-ALL-SVRS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm REDIRECT
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name RT-LINUX-153
   inservice
!
 policy REDIRECT-1K
  url-map REDIRECT-1K
  serverfarm REDIRECT-1K
!
 policy REDIRECT-10K
  url-map REDIRECT-10K
  serverfarm REDIRECT-10K
!
 policy REDIRECT-100K
  url-map REDIRECT-100K
  serverfarm REDIRECT-100K
!
 vserver RED-100K-VIP
  virtual 192.168.120.220 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver RED-10K-VIP
  virtual 192.168.120.221 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver RED-1K-VIP
  virtual 192.168.120.222 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver REDIRECT
  virtual 192.168.120.213 tcp www
  serverfarm REDIRECT
  persistent rebalance
  slb-policy REDIRECT-1K
  slb-policy REDIRECT-10K
  slb-policy REDIRECT-100K
  inservice
!
```

Test Procedure

The procedure used to perform the Redirect Policy—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Issue an HTTP request on a client for /redirect-1k.html. Verify through the client output that the redirect was successful.

Step 4 Verify that the load balancer redirected the the traffic as expected.

Step 5 Clear select counters on the load balancer.

Step 6 Start a series of HTTP GET's for multiple URLs, matching and not matching the redirect policies.

Step 7 Verify that the load balancer redirected the the traffic as expected.

Expected Results

The following test results are anticipated:

- We expect the load balancer to respond to the client with a 302 redirect for only the applicable traffic
- We expect that the load balancer will not crash or become unresponsive.

Results

Redirect Policy—CSM passed.

Topology Baseline—ACE

With background traffic running, the network is kept in a steady state for approximately sixty minues. Memory and CPU statistics are taken before and after steady state. This test verified that the network is in a known, good, state and ready for testing.

Test Procedure

The procedure used to perform the Topology Baseline—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Capture memory utilization on the Primary and Standby load balancer modules.

Step 3 Capture free memory and CPU utilization on the Primary and Standby load balancer modules.

Step 4 Run a script that will sleep for sixty minutes, while the network remains in a steady state.

Step 5 Capture memory utilization on the Primary and Standby load balancer modules.

Step 6 Capture free memory and CPU utilization on the Primary and Standby load balancer modules.

Expected Results

The following test results are anticipated:

- We expect that the load balancer will not crash or become unresponsive.
- We expect no CPU or memory problems.

Results

Topology Baseline—ACE passed with exception. The following exceptions were noted: CSCsu95887.

Topology Baseline—CSM

With background traffic running, the network is kept in a steady state for approximately sixty minues. Memory and CPU statistics are taken before and after steady state. This test verified that the network is in a known, good, state and ready for testing.

Test Procedure

The procedure used to perform the Topology Baseline—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Capture free memory and CPU utilization on the Primary and Standby load balancer modules.

Step 3 Run a script that will sleep for sixty minutes, while the network remains in a steady state.

Step 4 Capture free memory and CPU utilization on the Primary and Standby load balancer modules.

Step 5 Compare the free memory and CPU utilizations taken before and after the steady state period.

Expected Results

The following test results are anticipated:

- We expect that the load balancer will not crash or become unresponsive.
- We expect no CPU or memory problems.

Results

Topology Baseline—CSM passed.

UDP Load Balancing—ACE

Two types of UDP traffic is tested for basic load balancing. First DNS traffic is used and then a custom application. DNS is tested to see if the responses returned are correct. The custom application sends a large UDP datagram spanned (not fragmented) across 3 packets. This is checked to if the packets are sent and received in the correct order.

This test verified the ability of the Load Balancer to properly load balance DNS traffic and transmit/receive spanned UDP packets.

Relevant Load Balancer Configuration

```
probe udp UDP
  interval 5
  passdetect interval 10
probe udp UDP:2222
  port 2222
  interval 5
  passdetect interval 2
```

```
serverfarm host UDP
  probe UDP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-151
    inservice
policy-map multi-match SH-Gold-VIPs
  class-map match-all UDP-VIP_114:UDP
    2 match virtual-address 192.168.120.114 udp any
  class UDP-VIP_114:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_UDP
    loadbalance vip icmp-reply
    nat dynamic 1 VLAN 120
    connection advanced-options 1SECOND-IDLE
```

Test Procedure

The procedure used to perform the UDP Load Balancing—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Start a traffic stream which will verify basic load balancing for DNS traffic.

Step 4 Let the traffic run for a few minutes, then stop it. Verify in the output that the first octet returned for the IP address matches the number used in the domain name.

Step 5 Start a traffic stream which will verify basic load balancing for DNS traffic, but at a higher rate.

Step 6 Verify that these requests are load balanced in the serverfarm, and have no connection failures.

Step 7 Stop the traffic.

Step 8 Add an additional server to the serverfarm, and then take a server out-of-service.

Step 9 Start a series of generic UDP packets with the data payload spanning 3 packets. Each packet will expect a response.

Step 10 Verify on the client that the expected responses (to the UDP packets) are received in the expected order.

Step 11 Stop the traffic.

Step 12 Change the load balancer configuration back to its original configuration.

Expected Results

The following test results are anticipated:

- We expect that DNS traffic will be load balanced across the servers with the correct responses received.
- We expect that UDP data spanning multiple packets will be transmitted and received in the order it was sent.
- We expect that the load balancer will not crash or become unresponsive.

Results

UDP Load Balancing—ACE passed with exception. The following exceptions were noted: CSCsg80625 and CSCsu54652.

UDP Load Balancing—CSM

Two types of UDP traffic is tested for basic load balancing. First DNS traffic is used and then a custom application. DNS is tested to see if the responses returned are correct. The custom application sends a large UDP datagram spanned (not fragmented) across 3 packets. This is checked to if the packets are sent and received in the correct order.

This test verified the ability of the Load Balancer to properly load balance DNS traffic and transmit/receive spanned UDP packets.

Relevant Load Balancer Configuration

```
 serverfarm UDP
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm IDLE-UDP
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-LINUX-154
   inservice
!
 sticky 30 netmask 255.255.255.255 address source timeout 30
!
 vserver UDP
  virtual 192.168.120.219 udp dns
  serverfarm UDP
  idle 4
  persistent rebalance
  inservice
!
 vserver IDLE-UDP
  virtual 192.168.120.211 udp 0
  serverfarm IDLE-UDP
  idle 60
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the UDP Load Balancing—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Start a traffic stream which will verify basic load balancing for DNS traffic.

Step 4 Let the traffic run for a few minutes, then stop it. Verify in the output that the first octet returned for the IP address matches the number used in the domain name.

Step 5 Start a traffic stream which will verify basic load balancing for DNS traffic, but at a higher rate.

Step 6 Verify that these requests are load balanced in the serverfarm, and have no connection failures.

Step 7 Stop the traffic.

Step 8 Start a series of generic UDP packets with the data payload spanning 3 packets. Each packet will expect a response.

Step 9 Verify on the client that the expected responses (to the UDP packets) are received in the expected order.

Step 10 Stop the traffic.

Expected Results

The following test results are anticipated:

- We expect that DNS traffic will be load balanced across the servers with the correct responses received.
- We expect that UDP data spanning multiple packets will be transmitted and received in the order it was sent.
- We expect that the load balancer will not crash or become unresponsive.

Results

UDP Load Balancing—CSM passed.

SSL

Traffic Handling tests verify that the load balancer module can manage various types of traffic.

Secure Sockets Layer (SSL) is an application-level protocol that encryption technology for the Internet, ensuring secure transactions transmission of credit card numbers for e-commerce websites. SSL secure transaction of data between a client and a server through privacy, authentication, and data integrity. SSL relies upon certificates private-public key exchange pairs for this level of security. Specific types of SSL traffic covered by this category include SSL end-to-end, SSL initiation, and SSL termination.

This section contains the following topics:

- SSL URL Rewrite—ACE, page 183
- SSL URL Rewrite—CSM-SSLM, page 186

Client Authentication—ACE

During the flow of a normal SSL handshake, the server sends its certificate to the client. The client verifies the identity of the server through the certificate. However, the client does not send any identification of its own to the server. When you enable the client authentication feature the load balancer requires that the client sends a certificate to be authenticated.

This test verified that the load balancer will authenticate client certificates based upon the configured parameters.

Relevant ACE Configuration

```
crypto authgroup CAUTH_GRP
  cert init.pem
  cert bigcert.pem
  cert TB2-CA-121.cer
crypto crl TB2-CA-121_CRL http://tb2-ace-vm-3.cisco.com/CertEnroll/TB2-CA-121.crl
probe http GEN_HTTP
  interval 10
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "A2_ctx_ACE_CLEAR"
ip domain-lookup
ip domain-list cisco.com
ip name-server 10.86.83.121
parameter-map type ssl SSL_CAUTH
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 3660
parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow
ssl-proxy service CAUTH
  key term-wc.key
  cert term-wc.cer
  authgroup CAUTH_GRP
  crl TB2-CA-121_CRL
  ssl advanced-options SSL_CAUTH
serverfarm host SSL_CAUTH
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host SSL_CAUTH2
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
sticky http-cookie SSL_CAUTH STKY-CKY_CAUTH
  cookie insert
  timeout 30
  replicate sticky
  serverfarm SSL_CAUTH
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  10 match url .*
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
  31 match request-method rfc get
```

```
class-map match-all SSL_CAUTH-VIP_102:443
  2 match virtual-address 192.168.140.102 tcp eq https
class-map type http loadbalance match-all URL*_L7
  2 match http url .*
class-map type http loadbalance match-all URL_*.JPG
  2 match http url .*.jpg
policy-map type loadbalance first-match PLBSF_SSL_CAUTH
  class URL_*.GIF
    serverfarm SSL_CAUTH2
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_CAUTH.GIF"
  class URL_*.JPG
    serverfarm SSL_CAUTH2
    insert-http I_AM header-value "SSL_CAUTH.JPG"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
  class URL*_L7
    sticky-serverfarm STKY-CKY_CAUTH
    insert-http I_AM header-value "SSL_CAUTH_COOKIE_INS"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit
policy-map multi-match A2-VIPS
  class SSL_CAUTH-VIP_102:443
    loadbalance vip inservice
    loadbalance policy PLBSF_SSL_CAUTH
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server CAUTH
    connection advanced-options TCP_PARAM
```

Test Procedure

The procedure used to perform the Client Authentication—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 The CRL is set to publish updates hourly, so it is likely the CRL loaded on the load balancer is passed its NextUpdate interval. The load balancer will download a new copy if this interval has passed and a client authentication request is received. Turn on SSL debug logging to see load balancer attempts of downloading. Verify that the CRL is downloaded and note the timestamp.

Step 4 On the CA from where the CRL is downloaded from, rename the file, so that a new copy cannot be downloaded.

Step 5 From a Windows client open an IE browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a valid client cert. The browser request should be successful as the default behavior of the load balancer is to not reject requests if the CRL is outdated.

Verify that load balancer attempted and failed to download the CRL (log messages will indicate). That the existing CRL is still loaded into memory and that the browser successfully loaded the page.

Step 6 On the CA from where the CRL is downloaded from, rename the file back to its original name, so that a new copy can be downloaded. Clear the relevant load balancer counters.

Step 7 From a Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a valid client cert. Verify that both browsers successfully load the page.

Step 8 Verify that the serverfarm has successful connections and that the Total SSL client authentication counter increase with none for Failed.

Step 9 From the Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a revoked client cert. Verify that both browsers fail to load the page.

Step 10 Verify that the serverfarm has no new connections and that the Failed SSL client authentication and SSL revoked certificates counters increase.

Step 11 From the Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a client cert that was altered (bad signature). Verify that both browsers fail to load the page.

Step 12 Verify that the serverfarm has no new connections and that the Total and Failed SSL client authentication incremented the same number of time without any increase for SSL revoked certificates.

Step 13 From the Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a client cert that is expired. Verify that both browsers fail to load the page.

Step 14 Verify that the serverfarm has no new connections and that the Total and Failed SSL client authentication incremented the same number of time without any increase for SSL revoked certificates.

Step 15 Configure the load balancer to use an expired CRL and verify that it has been downloaded.

Step 16 By default the ACE will not reject client authentication attempts if the CRL has passed its NextUpdate interval. From a Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a valid client cert. Verify that both browsers successfully load the page.

Step 17 From a Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a revoked client cert. Verify that both browsers fail to load the page.

Step 18 Configure the load balancer to reject client authentication attempts when the CRL has passed its NextUpdate interval.

Step 19 Clear the crypto and serverfarm statistics.

Step 20 Now the load balancer is configured to reject client authentication attempts if the CRL has passed its NextUpdate interval. From a Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a valid client cert. Verify that both browsers fail to load the page.

Step 21 Verify that the serverfarm has no new connections and that the Failed SSL client authentication and SSL revoked certificates counters increase.

Step 22 Configure the load balancer back to the default of allowing expired CRLs to be used.

Step 23 The load balancer can be configured to use the CRL distribution point embedded in the client certificate instead of a predefined location configured on the load balancer. When this is configured the load balancer will download and check the CRL for each client authentication request received. Since we will be checking certs from two different CA's a new root cert needs to be configured.

Step 24 Clear the crypto and serverfarm statistics.

Step 25 With the load balancer configured to look inside the client cert for a CRL distribution point it can query an individual CRL for each cert presented. From a Windows client open two IE and two Firefox browsers. Connect to https://cauth-102.safeharbor.com/ and on each pair, when prompted, select a valid client cert, one from each CA. Verify that both browsers successfully load the page.

Step 26 Verify that the serverfarm has successful connections and that the Total SSL client authentication and SSL best effort CRL lookups increase with none for Failed.

Step 27 From a Windows client open two IE and two Firefox browsers. Connect to https://cauth-102.safeharbor.com/ and on each pair, when prompted, select a revoked client cert, one from each CA. Verify that both browsers fail to load the page.

Step 28 Verify that the serverfarm has no new connections and that the Failed SSL client authentication, SSL revoked certificates, and SSL best effort CRL lookups counters increase.

Step 29 Configure the ACE to use a CRL signed by an untrusted CA and verify that it has been downloaded. Remove the newly added root CA cert in a prior step.

Step 30 From a Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a revoked client cert. Verify that both browsers fail to load the page.

Step 31 Configure the load balancer to use a non-existent CRL and verify that it fails to download.

Step 32 When the load balancer is configured to use a CRL and it cannot be downloaded, it will reject all requests, unless there is an existing copy (not in this case because of the config change). From a Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com/ and when prompted select a good client cert. Verify that both browsers fail to load the page.

Step 33 Configure the load balancer to use the proper CRL and verify that it has been downloaded.

Step 34 From a Windows client open an IE and Firefox browser. Connect to https://cauth-102.safeharbor.com:444/ and when prompted select a revoked client cert. Since this SSL proxy list does not have CRL checking enabled this will not get rejected. Verify that both browsers successfully load the page.

Step 35 From a Windows client use the open IE and Firefox browser windows. Change the URL from https://cauth-102.safeharbor.com:444/ to https://cauth-102.safeharbor.com/ and when prompted select a revoked client cert. Since this SSL proxy list does have CRL checking enabled this will get rejected. Verify that both browsers fail to load the page.

Step 36 Remove the debug logging.

Expected Results

The following test results are anticipated:

- We expect the load balancer, if CRL checking is enabled, to reject client certificates that have been revoked.
- We expect the load balancer to reject client requests if it is signed by an untrusted CA.
- We expect the load balancer, if CRL checking is enabled, to allow client certificates that are not listed as revoked.
- We expect that the load balancer will not crash or become unresponsive.

Results

Client Authentication—ACE failed. The following failures were noted: CSCsv01732.

Client Authentication—CSM-SSLM

During the flow of a normal SSL handshake, the server sends its certificate to the client. The client verifies the identity of the server through the certificate. However, the client does not send any identification of its own to the server. When you enable the client authentication feature the load balancer requires that the client sends a certificate to be authenticated.

This test verified that the load balancer will authenticate client certificates based upon the configured parameters.**Relevant CSM config**

```
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 serverfarm SSLM
  no nat server
  no nat client
  real 192.168.120.199
   inservice
!
 serverfarm WEB_SERVERS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name BRG-LINUX-15
   inservice
  real name RT-IIS-152
   inservice
  probe TCP-GENERIC
!
 vserver CLR-CLIENT-AUTH
  virtual 192.168.120.193 tcp www
  vlan 121
  serverfarm WEB_SERVERS
  persistent rebalance
  inservice
!
 vserver SSL-CLIENT-AUTH
  virtual 192.168.120.193 tcp https
  serverfarm SSLM
  persistent rebalance
  inservice
!
Relevant SSLM config
!
ssl-proxy service client-auth
 virtual ipaddr 192.168.120.193 protocol tcp port 443 secondary
 server ipaddr 192.168.120.7 protocol tcp port 80
 certificate rsa general-purpose trustpoint client-auth
 no nat server
 trusted-ca client-auth
 authenticate verify all
 inservice
!
ssl-proxy vlan 83
 ipaddr 10.86.83.118 255.255.255.0
 gateway 10.86.83.1
 admin
ssl-proxy vlan 121
 ipaddr 192.168.120.199 255.255.255.0
 route 172.28.0.0 255.254.0.0 gateway 192.168.120.254
!
ssl-proxy pool ca client-auth
 ca trustpoint safeharbor-IIS-root
 ca trustpoint TB2-CA-121.cer
!
crypto ca trustpoint client-auth
 enrollment terminal pem
 serial-number
 fqdn www.clientauth.com
 ip-address 10.1.0.204
 subject-name C=US, ST=Massachusetts, L=Boxborough, O=Cisco, OU=Safeharbor,
CN=www.clientauth.com
 rsakeypair client-auth
!
```

Test Procedure

The procedure used to perform the Client Authentication—CSM-SSLM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 From a Windows client open an IE browser and a Firefox browser. Connect to https://www.clientauth.com/ and when prompted select a valid client cert. The browser requests should be successful.

Step 4 Verify that the load balancer handled these requests without error (no fatal alerts).

Step 5 From a Windows client open an IE and Firefox browser. Connect to https://www.clientauth.com/ and when prompted select a revoked client cert. Verify that both browsers fail to load the page.

Step 6 Verify that the load balancer logged fatal errors when handling these requests.

Step 7 Clear the relevant counters.

Step 8 Configure client authentication to only do signature verification.

Step 9 From a Windows client open an IE browser and a Firefox browser. Connect to https://www.clientauth.com/ and when prompted, select a valid client cert. The browser requests should be successful.

Step 10 Verify that the load balancer handled these requests without error.

Step 11 From a Windows client open an IE and Firefox browser. Connect to https://www.clientauth.com/ and when prompted select a revoked client cert. With the SSLM configured to only verify the certificate signature a CRL check is not performed. Verify that both browsers successfully load the page.

Step 12 Verify that the load balancer handled these requests without error.

Step 13 Configure the load balancer to remove the trusted certificate, so that signature verification will fail.

Step 14 Clear the relevant counters.

Step 15 From a Windows client open an IE browser and a Firefox browser. Connect to https://www.clientauth.com/ and when prompted select a valid client cert with a bad signature. The browser requests will fail.

Step 16 Verify that the load balancer logged fatal errors when handling these requests.

Step 17 Configure the load balancer to back to its original config.

Expected Results

The following test results are anticipated:

- We expect the load balancer, if CRL checking is enabled, to reject client certificates that have been revoked.
- We expect the load balancer to reject client requests if it is signed by an untrusted CA.
- We expect the load balancer, if CRL checking is enabled, to allow client certificates that are not listed as revoked.

Results

Client Authentication—CSM-SSLM passed.

End to End SSL—ACE

End-to-end SSL refers to the load balancer establishing and maintaining SSL connections between the client at one end of the connection and the server at the other end of the connection, while briefly the traffic is unencrypted to allow for advanced load balancing methods. With the load balancer configured for end-to-end SSL it terminates an SSL session with the client (front-end connection), initiates an SSL session with the server (back-end connection), and load balances the back-end content.

This test verified that the load balancer can handle a variety of traffic (GET, POST, CHUNKED, Spanned Headers) for a period of time, while maintaining long lived flows.

Relevant ACE Configuration

```
probe https SSL
  interval 10
  passdetect interval 7
  passdetect count 2
  expect status 200 200
  header Via header-value "A2_ctx_SSL_Prb"
probe tcp TCP:443
  port 443
  interval 5
  passdetect interval 10
  connection term forced
  open 3
parameter-map type http HTTP_REBAL_REUSE
  case-insensitive
  persistence-rebalance
  server-conn reuse
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow
```

```
parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type ssl CLIENT_SSL
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 0
  close-protocol disabled
parameter-map type ssl TERM_SSL
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 3660
ssl-proxy service E2E_SSL
  key pkey.pem
  cert end-to-end.pem
  ssl advanced-options TERM_SSL
ssl-proxy service INIT_E2E_SSL
  ssl advanced-options CLIENT_SSL
serverfarm host E2E
  failaction purge
  predictor leastconns
  probe SSL
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-IIS-241 443
    inservice
  rserver LOCAL-IIS-245 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-IIS-152 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host E2E_POST
  failaction purge
  predictor leastconns
  retcode 100 599 check count
  probe TCP:443
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
```

```
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host E2E_CHUNK
  failaction purge
  predictor leastconns
  probe TCP:443
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-LINUX-151 443
    inservice
sticky http-cookie E2E_SSL STKY-CKY_E2E_ANY
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E
sticky http-cookie E2E_SSL STKY-CKY_E2E_POST
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E_POST
sticky http-cookie E2E_SSL STKY-CKY_E2E_CHUNK
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E_CHUNK
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  10 match url .*
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
  31 match request-method rfc get
class-map type http loadbalance match-any POST.PL_URL
  2 match http url .*cgipostform.pl
class-map type http loadbalance match-all CHUNK.PL_URL
```

```
  2 match http url .*.chunk.pl
class-map type http loadbalance match-all URL_*.GIF
  2 match http url .*.gif
class-map match-all E2E-VIP_105:443
  2 match virtual-address 192.168.140.105 tcp eq https
policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit
policy-map type loadbalance first-match PLBSF_LL_E2E
  class POST.PL_URL
    sticky-serverfarm STKY-CKY_E2E_POST
    insert-http I_AM header-value "E2E_SSL_POST"
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    ssl-proxy client INIT_E2E_SSL
  class CHUNK.PL_URL
    sticky-serverfarm STKY-CKY_E2E_CHUNK
    insert-http I_AM header-value "E2E_SSL_CHUNK_XFER"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client INIT_E2E_SSL
  class class-default
    sticky-serverfarm STKY-CKY_E2E_ANY
    insert-http I_AM header-value "E2E_SSL_ANY"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client INIT_E2E_SSL
policy-map multi-match A2-VIPS
  class E2E-VIP_105:443
    loadbalance vip inservice
    loadbalance policy PLBSF_LL-E2E
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_REBAL_REUSE
    ssl-proxy server E2E_SSL
    connection advanced-options TCP_PARAM
```

Test Procedure

The procedure used to perform the End to End SSL—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 Start a series of client emulators on a Linux client that will generate 16 long lived HTTPS flows.

Step 4 Verify that the serverfarm has 2 open connections for each rserver and gather the connection information.

Step 5 On a Linux client change the MTU to 576 to cause the HTTPS requests to span packets.

Step 6 The following steps will send a series of different types of HTTPs traffic getting file of various sizes. Send a series of HTTPS traffic using persistent and non-persistent connections (HTTP 1.1 / 1.0) requesting different sized files, serially. Capture the test tool’s output when complete.

Step 7 Send a series of requests sending a large POST and requested CHUNKED transfer. Capture the test tool’s last paragraph of output when complete.

Step 8 Verify that there were no errors seen on the load balancer.

Step 9 Launch a continuous stream of various types HTTPS traffic (POST, GET, CHUNKED, Size, Persistence(none)) from several Linux clients.

Step 10 Verify that all serverfarms are getting hits, no failures seen and the counters are incrementing.

Step 11 Verify that the original 16 flows setup initially are still running by issuing the **show conn serverfarm E2E | include 10.1.0.235**. Compare these connections to those captured at the beginning of this test.

Step 12 Run a series of commands to verify that the traffic is working and there are no errors. Let traffic continue to run for more than 1 hour before proceeding to the next step.

Step 13 Verify that the original 16 flows are still present, counters are incrementing with no or a small number of errors seen. Single digit errors values can typically be ignored, but if they persist and show up on the delta then they would need to be investigated.

Step 14 Stop the traffic that is being generated from all clients. Including the long-lived flows.

Step 15 The following steps will send a series of different types of HTTPs traffic getting file of various sizes. Send a series of HTTPS traffic using persistent and non-persistent connections (HTTP 1.1 / 1.0) requesting different sized files, serially. Capture the test tool's output when complete.

Step 16 Send a series of requests sending a large POST and requested CHUNKED transfer. Capture the test tool's last paragraph of output when complete.

Step 17 Verify that there were no errors seen on the load balancer.

Step 18 Configure the client to use the default MTU of 1500.

Expected Results

The following test results are anticipated:

- We expect the load balancer to handle long lived HTTPS flows without error.
- We expect the load balancer to handle various types and sizes of traffic generated for a period of time.

Results

End to End SSL—ACE failed. The following failures were noted: CSCsv02360.

End to End SSL—CSM-SSLM

End-to-end SSL refers to the load balancer establishing and maintaining SSL connections between the client at one end of the connection and the server at the other end of the connection, while briefly the traffic is unencrypted to allow for advanced load balancing methods. With the load balancer configured for end-to-end SSL it terminates an SSL session with the client (front-end connection), initiates an SSL session with the server (back-end connection), and load balances the back-end content.

This test verified that the load balancer can handle a variety of traffic (GET, POST, CHUNKED, Spanned Headers) for a period of time, while maintaining long lived flows.

Relevant CSM Configuration

```
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 105 client
  ip address 192.168.105.12 255.255.255.0 alt 192.168.105.13 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
  ip address 10.86.83.13 255.255.255.0 alt 10.86.83.14 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 serverfarm CLEAR-TEXT
  nat server
  nat client CLIENT_NAT
  real 192.168.120.190 83
   inservice
  real 192.168.120.190 84
   inservice
  real 192.168.120.190 85
   inservice
!
 serverfarm ROUTE
  no nat server
  no nat client
  predictor forward
!
 serverfarm SSLM
  no nat server
  no nat client
  real 192.168.120.199
   inservice
!
 vserver CLEAR-TEXT
  virtual 192.168.120.195 tcp 83
  vlan 121
  serverfarm CLEAR-TEXT
  sticky 10 group 203
  persistent rebalance
  inservice
!
 vserver DIRECT-ACCESS
  virtual 0.0.0.0 0.0.0.0 tcp 0
  vlan 121
  serverfarm ROUTE
  persistent rebalance
  inservice
!
 vserver SECURE-WEB
  virtual 192.168.120.194 tcp https
  vlan 120
  serverfarm SSLM
  persistent rebalance
  inservice
```

```
!
Relevant SSLM Configuration
!
ssl-proxy service backend-ssl client
 virtual ipaddr 192.168.120.190 protocol tcp port 83
 server ipaddr 172.29.0.240 protocol tcp port 443
 trusted-ca Safe Harbor
 authenticate verify signature-only
 inservice
!
ssl-proxy service backend-ssl2 client
 virtual ipaddr 192.168.120.190 protocol tcp port 84
 server ipaddr 172.28.0.151 protocol tcp port 443
 trusted-ca Safe Harbor
 authenticate verify signature-only
 inservice
!
ssl-proxy service backend-ssl3 client
 virtual ipaddr 192.168.120.190 protocol tcp port 85
 server ipaddr 192.168.120.11 protocol tcp port 443
 trusted-ca Safe Harbor
 authenticate verify signature-only
 inservice
!
ssl-proxy service secure-web
 virtual ipaddr 192.168.120.194 protocol tcp port 443 secondary
 server ipaddr 192.168.120.195 protocol tcp port 83
 certificate rsa general-purpose trustpoint secure-web
 inservice
!
ssl-proxy vlan 83
 ipaddr 10.86.83.118 255.255.255.0
 gateway 10.86.83.1
 admin
ssl-proxy vlan 121
 ipaddr 192.168.120.199 255.255.255.0
 route 172.28.0.0 255.254.0.0 gateway 192.168.120.254
!
ssl-proxy pool ca Safe Harbor
 ca trustpoint safeharbor-root
 ca trustpoint LOCAL-LINUX-240
 ca trustpoint RT-LINUX-151
 ca trustpoint BRG-LINUX-11
!
```

Test Procedure

The procedure used to perform the End to End SSL—CSM-SSLM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 Start a series of client emulators on a Linux client that will generate 6 long lived HTTPS flows.

Step 4 Verify that the serverfarm has 2 open connections for each rserver and gather the connection information.

Step 5 On a Linux client change the MTU to 576 to cause the HTTPS requests to span packets.

Step 6 The following steps will send a series of different types of HTTPs traffic getting file of various sizes. Send a series of HTTPS traffic using persistent and non-persistent connections (HTTP 1.1 / 1.0) requesting different sized files, serially. Capture the test tool's output when complete.

Step 7 Send a series of requests sending a large POST and requested CHUNKED transfer. Capture the test tool's last paragraph of output when complete.

Step 8 Verify that there were no errors seen on the load balancer.

Step 9 Launch a continuous stream of various types HTTPS traffic (POST, GET, CHUNKED, Size, Persistence(none)) from several Linux clients.

Step 10 Verify that all serverfarms are getting hits, no unusual failures seen and the counters are incrementing.

Step 11 Verify that the original 6 flows setup initially are still running.

Step 12 Run a series of commands to verify that the traffic is working and there are no unusual errors. Let traffic continue to run for more than 1 hour before proceeding to the next step.

Step 13 Verify that the original 6 flows are still present, counters are incrementing with no or a small number of errors seen. Single digit errors values can typically be ignored, but if they persist and show up on the delta then they would need to be investigated.

Step 14 Stop the traffic that is being generated from all clients, including the long-lived flows. Clear the relevant counters.

Step 15 The following steps will send a series of different types of HTTPS traffic getting file of various sizes. Send a series of HTTPS traffic using persistent and non-persistent connections (HTTP 1.1 / 1.0) requesting different sized files, serially. Capture the test tool's output when complete.

Step 16 Send a series of requests sending a large POST and requested CHUNKED transfer. Capture the test tool's last paragraph of output when complete.

Step 17 Verify that there were no unusual errors seen on the load balancer.

Step 18 Configure the client to use the default MTU of 1500.

Expected Results

The following test results are anticipated:

- We expect the load balancer to handle long lived HTTPS flows without error.
- We expect the load balancer to handle various types and sizes of traffic generated for a period of time.

Results

End to End SSL—CSM-SSLM passed.

Header Insert (SSL)—ACE

The HTTP header insert feature provides the load balancer the ability to insert information, such as the client IP, destination IP, custom, and other types into the HTTP header.

The CSM/SSLM provides the ability to insert SSL information such as certificates and session ID. At present the ACE does not perform this function, but is expected to in an upcoming release. This test verified that load balancer inserted the configured headers when GETs and POSTs were issued during HTTPS sessions.

Relevant Load Balancer Configuration

```
parameter-map type http PERSIST-REBAL-4K
  persistence-rebalance
  set header-maxparse-length 4096
parameter-map type ssl TERM_SSL
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
ssl-proxy service ACE_TERM
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options TERM_SSL
serverfarm host HEADER-INSERT
  rserver BRG-13 80
    inservice
  rserver RT-151 80
    inservice
  rserver RT-153 80
    inservice
serverfarm host HEADER-INSERT2
  rserver BRG-12 80
    inservice
  rserver BRG-14 80
    inservice
  rserver LOCAL-240 80
    inservice
  rserver LOCAL-241 80
    inservice
  rserver LOCAL-244 80
    inservice
  rserver LOCAL-245 80
    inservice
  rserver RT-153 80
    inservice
  rserver RT-154 80
    inservice
class-map match-all HEADER-INSERT-VIP_121:443
  match virtual-address 192.168.120.121 tcp eq https
class-map match-all HEADER-INSERT2-VIP_122:443
  match virtual-address 192.168.120.122 tcp eq https
class-map type http loadbalance match-all P-HDR-INSERT
   2 match http url .*
class-map type http loadbalance match-all P-HDR-IXIA
   2 match http url .*
class-map type http loadbalance match-all P-HDR-SRCDST-IP
   2 match http url .*
policy-map type loadbalance first-match PLBSF_HEADER-INSERT
  class P-HDR-INSERT
    serverfarm HEADER-INSERT
    insert-http Source-IP header-value "%is"
    insert-http Accept header-value "anything"
    insert-http Pragma header-value "Pragma no Pragma that is the question"
    insert-http Destination_iP header-value "%id"
    insert-http
Custom-header_name_size_100bytes_abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQR
S_100BYTES header-value "Size of inserted header value is 100 bytes
abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHI_100BYTES"
policy-map type loadbalance first-match PLBSF_HEADER-INSERT2
  class P-HDR-SRCDST-IP
    serverfarm HEADER-INSERT2
    insert-http Destination_iP header-value "%id"
    insert-http Source-IP header-value "%is"
policy-map multi-match SH-Gold-VIPs
    class HEADER-INSERT-VIP_121:443
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
    ssl-proxy server ACE_TERM
  class HEADER-INSERT2-VIP_122:443
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
    ssl-proxy server ACE_TERM
```

Test Procedure

The procedure used to perform the Header Insert (SSL)—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the load balancer make the necessary changes to the config to allow POSTs.

Step 3 Clear the relevant counters.

Step 4 Start a packet capture on the internal portchannel.

Step 5 On a Windows client, set the MTU (Maximum Transmission Unit) to 576 bytes to remote client.

Step 6 Use an IE and Firefox browser to issue a single request to https://192.168.120.121/.

Step 7 Stop the packet capture and view the results.

Isolate requests to vserver IP 192.168.120.121. Verify on the server side of the trace that HTTPS requests hitting this VIP have the client's (src IP) address NATed to 192.168.120.70, while embedding the actual client IP into the Source-IP header, along with all of the other configured headers.

Step 8 Verify that headers were inserted and that no errors were seen.

Step 9 Restart the packet capture on the internal portchannel.

Step 10 On a Linux client, set the interface to an MTU of 576 to simulate a remote user. Send a short series of HTTP POST and GET requests to 192.168.120.121.

Step 11 Stop the packet capture and view the results.

Isolate requests to vserver IP 192.168.120.121. Verify on the server side of the trace that HTTP requests hitting this VIP have the client's (src IP) address NATed to 192.168.120.70, while embedding the actual client IP into the Source-IP header, along with all of the other configured headers.

Step 12 Verify that headers were inserted and that no errors were seen.

Step 13 On a Linux client, send a short series of POST and GET requests with large URLs and POST payload to 192.168.120.121.

Step 14 Stop the packet capture and view the results. Isolate the requests to 192.168.120.121 and verify that all of the configured headers are inserted on the server side of the trace.

Step 15 These steps will test the ability of the load balancer to insert headers in all requests on a persistent connection. Start a packet capture on the internal portchannel.

Step 16 On a Linux client start two long-lived flows to issue a continuous series of POST and GET requests to 192.168.120.121.

Step 17 Let the traffic run for a minimum of 30s. Then stop the client traffic.

Step 18 Stop the packet capture and view the results. Verify that the configured headers are inserted on the server side of the trace for all the GET and POST requests.

Step 19 Verify that the there were no reported header insert errors and then clear the counters.

Step 20 On several Linux clients launch a continuous stream of traffic to the vserver 192.168.120.121.

Step 21 Let the test traffic run to completion (about 20 minutes), while periodically gathering statistics.

Step 22 When the traffic is finished verify that there were no reported header insert errors.

Step 23 Set client mtu back to 1500.

Step 24 Return the config back to its original state.

Expected Results

The following test results are anticipated:

- We expect the load balancer to properly insert the configured headers.
- We expect the load balancer to properly insert the configured headers when requests span multiple packets.
- We expect that the load balancer will not crash or become unresponsive.

Results

Header Insert (SSL)—ACE passed.

Header Insert (SSL)—CSM-SSLM

The HTTP header insert feature provides the load balancer the ability to insert information, such as the client IP, destination IP, custom, and other types into the HTTP header.

The CSM/SSLM provides the ability to insert SSL information such as certificates and session ID. At present the ACE does not perform this function, but is expected to in an upcoming release. This test verified that load balancer inserted the configured headers when GETs and POSTs were issued during HTTPS sessions.

Relevant CSM Configuration

```
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
!
 serverfarm SSLM
  no nat server
  no nat client
  real 192.168.120.199
   inservice
!
 serverfarm WEB_SERVERS2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-LINUX-151
   inservice
  real name BRG-LINUX-11
   inservice
!
 vserver CLR-REWRITE80
  virtual 192.168.120.196 tcp www
  vlan 121
  serverfarm WEB_SERVERS
  persistent rebalance
  inservice
!
 vserver SSL-REWRITE80
  virtual 192.168.120.196 tcp https
  serverfarm SSLM
  persistent rebalance
  inservice
!
Relevant SSLM Configuration
!
ssl-proxy policy http-header backend-ssl
 custom
"240:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
 client-ip-port
 prefix backend-ssl
!
ssl-proxy service urlrewrite80
 virtual ipaddr 192.168.120.196 protocol tcp port 443 secondary
 server ipaddr 192.168.120.7 protocol tcp port 80
 certificate rsa general-purpose trustpoint urlrewrite80
 no nat server
 policy url-rewrite urlrewrite80
 inservice
!
ssl-proxy vlan 83
 ipaddr 10.86.83.118 255.255.255.0
 gateway 10.86.83.1
 admin
ssl-proxy vlan 121
 ipaddr 192.168.120.199 255.255.255.0
 route 172.28.0.0 255.254.0.0 gateway 192.168.120.254
```

```
!
```

Test Procedure

The procedure used to perform the Header Insert (SSL)—CSM-SSLM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the load balancer make the necessary changes to the config to allow POSTs.

Step 3 Clear the relevant counters.

Step 4 Start a packet capture on the internal portchannel.

Step 5 On a Windows client, set the MTU (Maximum Transmission Unit) to 576 bytes to remote client.

Step 6 Use an IE and Firefox browser to issue a single request to https://192.168.120.196/.

Step 7 Stop the packet capture and view the results.

Isolate requests to vserver IP 192.168.120.196. Verify on the server side of the trace that HTTPS requests hitting this VIP have the client's (src IP) address NATed to 192.168.120.209, while embedding the actual client IP of 10.1.0.242 in the Source-IP header, along with all of the other configured headers.

Step 8 Verify that headers were inserted and that no errors were seen.

Step 9 Restart the packet capture on the internal portchannel.

Step 10 On a Linux client, set the interface to an MTU of 576 to simulate a remote user. Send a short series of HTTP POST and GET requests to 192.168.120.196.

Step 11 Stop the packet capture and view the results.

Isolate requests to vserver IP 192.168.120.196. Verify on the server side of the trace that HTTP requests hitting this VIP have the client's (src IP) address NATed to 192.168.120.209, while embedding the actual client IP into the Source-IP header, along with all of the other configured headers.

Step 12 Verify that headers were inserted and that no errors were seen.

Step 13 Restart the packet capture on the internal portchannel.

Step 14 On a Linux client, send a short series of POST and GET requests with large URLs and POST payload to 192.168.120.196.

Step 15 Stop the packet capture and view the results. Isolate the requests to 192.168.120.196 and verify that all of the configured headers are inserted on the server side of the trace.

Step 16 These steps will test the ability of the load balancer to insert headers in all requests on a persistent connection. Start a packet capture on the internal portchannel.

Step 17 On a Linux client start two long-lived flows to issue a continuous series of POST and GET requests to 192.168.120.196.

Step 18 Let the traffic run for a minimum of 30s. Then stop the client traffic.

Step 19 Stop the packet capture and view the results. Verify that the configured headers are inserted on the server side of the trace for all the GET and POST requests.

Step 20 Verify that the there were no reported header insert errors and then clear the counters.

Step 21 On several Linux clients launch a continuous stream of traffic to the vserver 192.168.120.196.

Step 22 Let the test traffic run to completion (about 20 minutes), while periodically gathering statistics.

Step 23 When the traffic is finished verify that there were no reported header insert errors.

Step 24 Set client mtu back to 1500.

Step 25 Return the config back to its original state.

Expected Results

The following test results are anticipated:

- We expect the load balancer to properly insert the configured headers.
- We expect the load balancer to properly insert the configured headers when requests span multiple packets.

Results

Header Insert (SSL)—CSM-SSLM passed.

SSL Termination—ACE

A load balancer acting as a SSL proxy server, will terminate SSL/TLS connections from a client and then established a clear text (unencrypted) TCP connection to an HTTP server.

This test verified SSL/TLS termination using a range of cipher suites.

Relevant ACE Configuration

```
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  4 match request-method rfc get
  5 match request-method rfc head
  6 match request-method rfc options
  7 match request-method rfc post
  8 match request-method rfc put
  9 match request-method rfc trace
  10 match url .*
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
class-map type http loadbalance match-all LB_CLASS_HTTP
  2 match http url .*
  3 match source-address 192.168.130.0 255.255.255.0
class-map match-all SSL_TERM-11
```

```
  description TERM SSL VIP
  2 match virtual-address 192.168.130.11 tcp eq https
class-map type http loadbalance match-all STICK_ME_TO_SERVER
  description STICKY FOR SSL TESTING
  2 match http url .*.jpg
  3 match source-address 192.168.130.0 255.255.255.0
policy-map type loadbalance first-match POLICY_SSL_TERM
  class STICK_ME_TO_SERVER
    insert-http I_AM header-value "SSL_TERM"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "is"
  class LB_CLASS_HTTP
    serverfarm ACE_TERM_SERVERS_CLEAR
    insert-http I_AM header-value "SSL_TERM"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "is"
policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_TERM-11
    loadbalance vip inservice
    loadbalance policy POLICY_SSL_TERM
    loadbalance vip icmp-reply
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server ACE_TERM
    connection advanced-options TCP_PARAM
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5
  version TLS1
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow
ssl-proxy service ACE_TERM
  key pkey.pem
  cert term.pem
  ssl advanced-options PARM_ACE_TERM_EXPORT_CIPHERS
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
```

```
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
```

Test Procedure

The procedure used to perform the SSL Termination—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 Configure the ssl-proxy service for export ciphers.

Step 4 Verify the priority order of the configured ciphers.

Step 5 Verify that the Firefox browser is configured to send both SSL 3.0 and TLS 1.0 record layer/client hello packets.

Step 6 From a WinXP client start a packet capture. Using IE, open a connection to **https://www.ssl-term-ace.com**

Step 7 Stop the packet capture and check the client/server hello packets for cipher suites offered and accepted. Verify that the SSL record layer version sent by the browser is TLS 1.0 and that the server responded with highest priority cipher configured, RSA_EXPORT1024_WITH_DES_CBC_SHA (priority:5).

Step 8 Verify the client traffic is matching on the proper policy/class map and that there were no SSL errors seen.

Step 9 Alter the priority of the ciphers within the parameter map to make a different cipher the highest priority.

Step 10 Start a new packet capture and relaunch the browser making a request to https://www.ssl-term-ace.com. Stop the packet capture and verify that the cipher with the recently modified priority was used.

Step 11 Alter the priority of the ciphers within the parameter map to make a different cipher the highest priority.

Step 12 Start a new packet capture and relaunch the browser making a request to https://www.ssl-term-ace.com. Stop the packet capture and verify the correct cipher was used.

Step 13 Restore parameter-map back to its original settings.

Step 14 Configure the load balancer to use a different parameter-map that has cipher RSA_WITH_RC4_128_MD5 configured with a priority of six. Close all browsers and Launch the Internet Explorer 7 browser.

Step 15 In the Internet Explorer 7 browser open a new connection to https://www.ssl-term-ace.com. Click browse and upload the file "20k-textfile.doc".

This should upload without any errors.

Step 16 Remove the RC4 cipher within the parameter map replacing it with ones that do not use RC4.

Step 17 In the Internet Explorer 7 browser open a new connection to https://www.ssl-term-ace.com. Click browser and upload the file "ACE_Upload_Test_File.doc". While the browser is still POSTing change the priority of a cipher in an unused parameter-map.

Step 18 The upload of the file should complete without error.

Step 19 Change the parameter-maps back to their original configuration.

Expected Results

The following test results are anticipated:

- We expect the load balancer to terminate SSL connections using various ciphers.
- We expect the load balancer to terminate SSL traffic without error.
- We expect the ACE to select the correct cipher based on the configured priority.

Results

SSL Termination—ACE passed.

SSL Termination—CSM-SSLM

A load balancer acting as a SSL proxy server will terminate SSL/TLS connections from a client and then establish a clear text (unencrypted) TCP connection to an HTTP server.

This test verified SSL/TLS termination using a range of cipher suites.

Relevant CSM/SSLM Configuration

```
SSLM
!
ssl-proxy policy ssl RC4_MD5
 cipher rsa-with-rc4-128-md5
ssl-proxy policy ssl DES_SHA
 cipher rsa-with-3des-ede-cbc-sha
!
ssl-proxy service urlrewrite80
 virtual ipaddr 192.168.120.196 protocol tcp port 443 secondary
 server ipaddr 192.168.120.7 protocol tcp port 80
 certificate rsa general-purpose trustpoint urlrewrite80
 no nat server
 policy url-rewrite urlrewrite80
 inservice
!
ssl-proxy vlan 83
 ipaddr 10.86.83.118 255.255.255.0
 gateway 10.86.83.1
 admin
ssl-proxy vlan 121
 ipaddr 192.168.120.199 255.255.255.0
 route 172.28.0.0 255.254.0.0 gateway 192.168.120.254
!
CSM
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
```

```
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
!
 serverfarm SSLM
  no nat server
  no nat client
  real 192.168.120.199
   inservice
!
 serverfarm WEB_SERVERS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name BRG-LINUX-15
   inservice
  real name RT-IIS-152
   inservice
  probe TCP-GENERIC
!
 vserver CLR-REWRITE80
  virtual 192.168.120.196 tcp www
  vlan 121
  serverfarm WEB_SERVERS
  persistent rebalance
  inservice
!
 vserver SSL-REWRITE80
  virtual 192.168.120.196 tcp https
  serverfarm SSLM
  persistent rebalance
  inservice
```

Test Procedure

The procedure used to perform the SSL Termination CSM-SSLM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 From a WinXP client start a packet capture. Using a Firefox browser open a connection to https://192.168.120.196/

Step 4 Stop the packet capture and check the client/server hello packets for cipher suites offered and accepted. Verify that the server responded with highest priority cipher (rsa-with-3des-ede-cbc-sha) configured.

Step 5 Verify the client traffic is matching on the proper policy/class map.

Step 6 Force the browser to not include the previously used cipher, so that another is selected.

Step 7 Start a new packet capture and relaunch the browser making a request to https://192.168.120.196/. Stop the packet capture and verify that a different cipher (rsa_rc4_128_md5) was used.

Step 8 Verify the client traffic is matching on the proper policy/class map.

Step 9 Force the browser to not include the previously used cipher, so that another is selected.

Step 10 Start a new packet capture and relaunch the browser making a request to https://192.168.120.196/. Stop the packet capture and verify the correct cipher was used.

Step 11 Verify the client traffic is matching on the proper policy/class map.

Step 12 Restore the browser to include all ciphers.

Step 13 Configure the load balancer to use a cipher policy that only has RSA_WITH_RC4_128_MD5 configured. Close all browsers and Launch the Internet Explorer 7 browser.

Step 14 In the Internet Explorer 7 browser open a new connection to https://192.168.120.196. Click browse and upload the file "20k-textfile.doc".

This should upload without any errors.

Step 15 Replace the cipher policy with one that does not use RC4.

Step 16 In the Internet Explorer 7 browser open a new connection to https://192.168.120.196. Click browser and upload the file "ACE_Upload_Test_File.doc". While the browser is still POSTing change the cipher in an unused policy.

Step 17 The upload of the file should complete without error.

Step 18 Change the parameter-maps back to their original configuration.

Step 19 Clear the relevant counters.

Step 20 Launch a stream of SSL connections and verify that the test tool finished without error.

Expected Results

The following test results are anticipated:

- We expect the load balancer to terminate SSL connections using various ciphers.
- We expect the load balancer to terminate SSL traffic without error.

Results

SSL Termination CSM-SSLM passed with exception. The following exceptions were noted: CSCsr41176.

SSL URL Rewrite—ACE

SSL URL Rewrite feature allows the load balancer to rewrite HTTP redirects to HTTPS when terminating (proxying) SSL traffic.

This test verified that the load balancer can perform SSL URL rewrite on http 301s and 302s responses.

Relevant ACE Configuration

```
parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type ssl TERM_SSL
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 3660
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow
action-list type modify http HTTP-HTTPS-Rewrite-Term
  header insert response Cache-Control header-value "max-age=901"
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header insert request Pragma header-value "no-cache, no-cache is the only pragma value I 
have ever seen"
  header delete response ETag
  header delete response Set-Cookie
  header rewrite response Keep-Alive header-value "timeout=150" replace "timeout=300"
  header rewrite request User-Agent header-value ".*petescape.*" replace "Mozilla/4.0 
(compatible; Tarantula Client; Runs on Linux)"
  ssl url rewrite location "192.168.130.103"
  ssl url rewrite location "www.ssl-term-ace.com"
ssl-proxy service TERM_SSL
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options TERM_SSL
serverfarm host TERM
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 81 -- change to apache port
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 81 -- change to apache port
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 81 -- change to apache port
    inservice
sticky http-cookie TERM_SSL STKY-CKY_TERM
  cookie insert
  timeout 30
  replicate sticky
  serverfarm TERM
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  4 match request-method rfc get
  5 match request-method rfc head
  6 match request-method rfc options
  7 match request-method rfc post
  8 match request-method rfc put
  9 match request-method rfc trace
  10 match url .*
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
class-map match-all TERM-VIP_103:443
  2 match virtual-address 192.168.140.103 tcp eq https
policy-map type loadbalance first-match PLBSF_TERM
  class URL_*.GIF
    serverfarm TERM2
    action HTTP-HTTPS-Rewrite-Term
    insert-http I_AM header-value "SSL_TERM.GIF"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
  class URL_*.JPG
    serverfarm TERM2
    action HTTP-HTTPS-Rewrite-Term
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM.JPG"
  class URL*_L7
    sticky-serverfarm STKY-CKY_TERM
    action HTTP-HTTPS-Rewrite-Term
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM_COOKIE_INS"
policy-map multi-match A2-VIPS
  class TERM-VIP_103:443
    loadbalance vip inservice
    loadbalance policy PLBSF_TERM
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server TERM_SSL
    connection advanced-options TCP_PARAM
```

Test Procedure

The procedure used to perform the SSL URL Rewrite —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Add the action lists to the load balance policy maps. Take all the Windows servers out of service. They do not gice 301s or 302s ao are useless in this test.

Step 3 On a WinXP client verify that the mtu to 576.

Step 4 Clear the relevant counters.

Step 5 From the NAM, start a capture on the internal portchannel used by the load balancer.

Step 6 From a WinXP client using a Firefox and IE browser, open a connection to **https://www.ssl-term80.com/302redirect-term.html**. Stop the capture on the NAM.

Step 7 From the capture on the NAM, verify that a redirect was returned and the location header is rewritten from HTTP to HTTPS.

Step 8 Verify that the counters incremented indicating successful URL rewrites.

Step 9 From the NAM, start a capture on the internal portchannel used by the load balancer.

Step 10 From a WinXP client using a Firefox and IE browser, open a connection to **https://www.ssl-term80.com/301moved-term.html**. Stop the capture on the NAM.

Step 11 From the capture on the NAM, verify that a redirect was returned and the location header is rewritten from HTTP to HTTPS.

Step 12 Verify that the counters incremented indicating successful URL rewrites.

Step 13 Clear the relevant counters.

Step 14 From the NAM, start a capture on the internal portchannel used by the load balancer.

Step 15 Using curl, send some requests to https://192.168.140.103/302redirect-term.html, https://192.168.140.103/301moved-term.html. In these requests use a large header and cookie that will span multiple packets.

Step 16 Using the curl output and the NAM capture verify that the load balancer is rewriting the location header.

Step 17 Verify that the counters incremented indicating successful URL rewrites.

Step 18 Return the configuration back to its original state.

Expected Results

The following test results are anticipated:

- We expect the load balancer to re-write SSL URLs on 301 and 302 responses from the http server.
- We expect the load balancer to re-write SSL URLs on 301 and 302 responses from the http server when requests span multiple packets.

Results

SSL URL Rewrite —ACE passed.

SSL URL Rewrite—CSM-SSLM

SSL URL Rewrite feature allows the load balancer to rewrite HTTP redirects to HTTPS when terminating (proxying) SSL traffic.

This test verified that the load balancer can perform SSL URL rewrite on http 301s and 302s responses.

Relevant CSM/SSLM config

```
CSM
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
```

```
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 serverfarm SSLM
  no nat server
  no nat client
  real 192.168.120.199
   inservice
!
 serverfarm WEB_SERVERS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name BRG-LINUX-15
   inservice
  real name RT-IIS-152
   inservice
  probe TCP-GENERIC
!
 serverfarm WEB_SERVERS3
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240 81
   inservice
  real name RT-LINUX-151 81
   inservice
  real name BRG-LINUX-11 81
   inservice
!
 vserver CLR-REWRITE80
  virtual 192.168.120.196 tcp www
  vlan 121
  serverfarm WEB_SERVERS
  persistent rebalance
  inservice
!
 vserver SSL-REWRITE80
  virtual 192.168.120.196 tcp https
  serverfarm SSLM
  persistent rebalance
  inservice
SSLM
!
ssl-proxy policy url-rewrite urlrewrite80
 url www.urlrewrite80-1.com
 url www.urlrewrite80-2.com
 url www.urlrewrite80-3.com
 url www.urlrewrite80-4.com
 url www.urlrewrite80-5.com
 url www.urlrewrite80-6.com
 url www.urlrewrite80-7.com
 url www.urlrewrite80-8.com
 url www.urlrewrite80-9.com
 url www.urlrewrite80-10.com
 url www.urlrewrite80-11.com
 url www.urlrewrite80-12.com
 url www.urlrewrite80-13.com
 url www.urlrewrite80-14.com
 url www.urlrewrite80-15.com
 url www.urlrewrite80-16.com
 url www.urlrewrite80-17.com
 url www.urlrewrite80-18.com
 url www.urlrewrite80-19.com
 url www.urlrewrite80-20.com
 url www.urlrewrite21*
 url www.urlrewrite22*
 url www.urlrewrite23*
 url www.urlrewrite24*
 url www.urlrewrite25*
 url www.urlrewrite26*
 url www.urlrewrite27*
 url www.urlrewrite28
 url www.urlrewrite29
 url www.urlrewrite31
 url www.urlrewrite80.com
 url 192.168.* sslport 888
!
ssl-proxy service urlrewrite80
 virtual ipaddr 192.168.120.196 protocol tcp port 443 secondary
 server ipaddr 192.168.120.7 protocol tcp port 80
 certificate rsa general-purpose trustpoint urlrewrite80
 no nat server
 policy url-rewrite urlrewrite80
 inservice
!
ssl-proxy vlan 83
 ipaddr 10.86.83.118 255.255.255.0
 gateway 10.86.83.1
 admin
ssl-proxy vlan 121
 ipaddr 192.168.120.199 255.255.255.0
 route 172.28.0.0 255.254.0.0 gateway 192.168.120.254
!
```

Test Procedure

The procedure used to perform the SSL URL Rewrite—CSM-SSLM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 On the WinXP client verify that the mtu is set to 576.

Step 3 Make a change to the serverfarm used to allow for the redirects to occur.

Step 4 Clear the relevant counters.

Step 5 From the NAM, start a capture on the internal portchannel used by the load balancer.

Step 6 From a WinXP client use a Firefox and IE browser to open a connection to **https://www.ssl-term80.com/302redirect-term.html**. Stop the capture on the NAM.

Step 7 From the capture on the NAM, verify that a redirect was returned and the location header is rewritten from HTTP to HTTPS.

Step 8 Verify that the counters incremented indicating successful URL rewrites.

Step 9 From the NAM, start a capture on the internal portchannel used by the load balancer.

Step 10 From a WinXP client using a Firefox and IE browser, open a connection to **https://www.ssl-term80.com/301moved-term.html**. Stop the capture on the NAM.

Step 11 From the capture on the NAM, verify that a redirect was returned and the location header is rewritten from HTTP to HTTPS.

Step 12 Verify that the counters incremented indicating successful URL rewrites.

Step 13 Clear the relevant counters.

Step 14 From the NAM, start a capture on the internal portchannel used by the load balancer.

Step 15 Using curl, send some requests to https://192.168.120.196/302redirect-term.html, https://192.168.120.196/301moved-term.html. In these requests use a large header and cookie that will span multiple packets.

Step 16 Using the curl output and the NAM capture verify that the load balancer is rewriting the location header.

Step 17 Verify that the counters incremented indicating successful URL rewrites.

Step 18 Return the config back to its original state.

Expected Results

The following test results are anticipated:

- We expect the load balancer to re-write SSL URLs on 301 and 302 responses from the http server.
- We expect the load balancer to re-write SSL URLs on 301 and 302 responses from the http server when requests span multiple packets.

Results

SSL URL Rewrite—CSM-SSLM passed.

Sticky

Stickiness is a load balancer feature that allows the same client to maintain multiple simultaneous or subsequent TCP or IP connections with the same real server for the duration of a session. A session, as used here, is defined as a series of transactions between a client and a server over some finite period of time (from several minutes to several hours). This feature is particularly useful for e-commerce applications where a client needs to maintain multiple connections with the same server while shopping online, especially while building a shopping cart and during the checkout process.

Depending on the configured SLB policy, the load balancer "sticks" a client to an appropriate server after the load balancer has determined which load-balancing method to use. If the load balancer determines that a client is already stuck to a particular server, then the load balancer sends that client request to that server, regardless of the load-balancing criteria specified by the matched policy. If the load balancer determines that the client is not stuck to a particular server, it applies the normal load balancing rules to the content request.

This section contains the following topics:

- Header Sticky—CSM, page 196
- IP Netmask Sticky—ACE, page 199
- IP Netmask Sticky—CSM, page 201
- SSL Session ID Sticky—ACE, page 202
- SSL Session ID Sticky—CSM, page 205

Cookie Sticky—ACE

Cookie sticky allows HTTP connections where a cookie is present to remain stuck to a specific server until the cookie or sticky timer expires. Cookies can be found in the HTTP header or embedded into the URL, with the Load Balancer supporting both methods. When the sticky is used with fault tolerance it can replicate these entries to the standby Load Balancer.

This test verified that HTTP connections remain stuck to a particular server when the same cookie is presented, whether the cookie was presented in the HTTP header or URL. Sticky entry replication and timeout were also checked.

Relevant Load Balancer Configuration

```
parameter-map type http COOKIE-DELIM
  persistence-rebalance
  set secondary-cookie-delimiters @$
serverfarm host STICKY-COOKIE
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver RT-154
    inservice
serverfarm host GEN-80
  probe TCP
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
sticky http-cookie COOKIE_TEST COOKIE-GROUP
  cookie secondary URLCOOKIE
  timeout 40
  replicate sticky
  serverfarm STICKY-COOKIE
  8 static cookie-value "REDSOX0" rserver RT-154
  16 static cookie-value "PATRIOTS0" rserver RT-151
class-map match-all STICKY-COOKIE-VIP_127:80
  2 match virtual-address 192.168.120.127 tcp eq www
class-map type http loadbalance match-all INDEX.HTML
  2 match http url /index.html
class-map type http loadbalance match-all URL*_L7
```

```
  2 match http url .*
policy-map type loadbalance first-match PLBSF_STICKY-COOKIE
  class INDEX.HTML
    sticky-serverfarm COOKIE-GROUP
  class URL*_L7
    serverfarm GEN-80
policy-map multi-match SH-Gold-VIPs
  class STICKY-COOKIE-VIP_127:80
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-COOKIE
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-DELIM
```

Test Procedure

The procedure used to perform the Cookie Sticky—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the primary and standby load balancers.

Step 3 Force a serverfarm to fail by adding a probe that will not become active. Verify the serverfarm state before proceeding.

Step 4 Start a series of HTTP 1.0 requests containing a specific cookie.

Step 5 Verify that the connections are being sent to the correct policy.

Step 6 Start a series of HTTP 1.0 requests with a cookie inserted into the HTTP header and cookies embedded into the URL. This time the cookie in the header will be modified to not match the sticky group forcing the group to look for the secondary cookie (url cookie).

Step 7 The traffic sent from the client has a cookie in the header that doesn't match the sticky group forcing it to look for the secondary cookie. The URLCOOKIE has values of VALUE2 and VALUE1 interchanged using a staggered number of GET requests, two to one. Two additional sticky entries will be created for a total of three and two different servers will get these new requests. Verify that one server gets double the number connections as other.

Step 8 Compare the counters and sticky table on the primary and standby load balancers.

Step 9 Clear select counters on the primary and standby load balancers.

Step 10 Start a series of HTTP 1.0 requests with a cookie inserted into the HTTP header using one of the predefined values and with cookies embedded into the URL.

Step 11 Verify that all of the connections go to a single server.

Step 12 Start a series of HTTP 1.0 requests with a cookie inserted into the HTTP header using one of the predefined values and also with cookies embedded into the URL.

Step 13 All of the connections should stick to an available server that is configured.

Step 14 Clear select counters and the sticky database on the primary and standby load balancers.

Step 15 Allow the serverfarm to become operational by removing the probe FORCED-FAIL. Verify the serverfarm state before proceeding.

Expected Results

The following test results are anticipated:

- We expect the use of default load balancing on HTTP requests that have a cookie that does not match the ACE configuration.
- We expect HTTP requests with a matching cookie to remain stuck to an rserver whether the cookie was found in the HTTP header or URL.
- We expect HTTP requests matching a static cookie to be sent to the configured rserver as configured.
- We expect sticky entries to NOT timeout on active when left to the default setting.
- We expect sticky entries to timeout on an active connection when configured.
- We expect that the load balancer will not crash or become unresponsive.

Results

Cookie Sticky—ACE passed.

Cookie Sticky—CSM

Cookie sticky allows HTTP connections where a cookie is present to remain stuck to a specific server until the cookie or sticky timer expires. Cookies can be found in the HTTP header or embedded into the URL, with the Load Balancer supporting both methods. When the sticky is used with fault tolerance it can replicate these entries to the standby Load Balancer.

This test verified that HTTP connections remain stuck to a particular server when the same cookie is presented, whether the cookie was presented in the HTTP header or URL. Sticky entry replication and timeout were also checked.

Relevant Load Balancer Configuration

```
!
 map COOKIE-MAP cookie
  match protocol http cookie CSM_TEST cookie-value This*is*a*test0
!
 real LOCAL-IIS-241
  address 172.29.0.241
  inservice
 real RT-LINUX-151
  address 172.28.0.151
  inservice
!
 serverfarm STICKY-COOKIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm COOKIE-1
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
!
 sticky 210 cookie CSM_TEST timeout 10
!
 policy STICKY-COOKIE
  cookie-map COOKIE-MAP
  serverfarm STICKY-COOKIE
!
 vserver STICKY-COOKIE
  virtual 192.168.120.227 tcp www
  serverfarm COOKIE-1
  persistent rebalance
  slb-policy STICKY-COOKIE
  inservice
!
```

Test Procedure

The procedure used to perform the Cookie Sticky—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that the load balancer does not have any sticky groups configured.

Step 3 Verify that the Virtual IP Address (VIP) is configured to use a sticky policy.

Step 4 Start a series of HTTP 1.0 requests containing a specific cookie.

Step 5 Verify that the connections are being sent to the correct policy.

Step 6 Verify that all of the connections are being load-balanced properly across all of the servers in serverfarm.

Step 7 Clear select counters on the load balancer.

Step 8 Start a series of HTTP requests containing a specific cookie.

Step 9 Verify that the connections are being sent to the correct policy.

Step 10 Configure the load balancer to use a sticky group.

Step 11 Clear select counters on the load balancer.

Step 12 Start a series of HTTP requests containing a specific cookie.

Step 13 Verify that the connections are being sent to the correct policy.

Step 14 Verify that the connections are stuck to a single real server in the serverfarm.

Step 15 Verify that sticky relationships are being maintained in the sticky database.

Step 16 Remove the sticky group from the load balancer.

Expected Results

The following test results are anticipated:

- We expect the use of default load balancing on HTTP requests that have a cookie that does not match the load balancer configuration.
- We expect HTTP requests matching a static cookie to be sent to the configured server.
- We expect that the load balancer will not crash or become unresponsive.

Results

Cookie Sticky—CSM passed.

Header Sticky—ACE

Server stickiness allows a particular connection to be maintained with a specific rserver of a serverfarm for its duration. If Host A requests a page from http://www.myurl.com, which is being served up by a farm of servers, that host can be paired with a single server for the duration of its connection. Stickiness can be created based on a number of different criteria, including a cookie field, Header, IP addr, etc. The Load Balancer maintains a database of client-to-real server pairings which can then be reused for subsequent connections for the timeout period.

This test verified that the Load Balancer stuck to a particular server based upon a well known HTTP header or a custom one that was user defined.

Relevant Load Balancer Configuration

```
serverfarm host STICKY-HEADER
  probe HTTP
  rserver BRG-12
    inservice
  rserver LOCAL-243
    inservice
  rserver RT-151
    inservice
serverfarm host STICKY-HEADER2
  probe HTTP
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
serverfarm host DEFAULT
  probe ICMP
  rserver BRG-15
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-153
    inservice
sticky http-header MSISDN HEADER-GROUP-42
  timeout 30
  serverfarm STICKY-HEADER
sticky http-header TestHeader HEADER-GROUP-41
  header offset 15 length 7
  timeout 30
  serverfarm STICKY-HEADER2
class-map match-all STICKY-HEADER_129:80
  2 match virtual-address 192.168.120.129 tcp eq www
class-map type http loadbalance match-all MSISDN
  2 match http header MSISDN header-value ".*"
class-map type http loadbalance match-all TestHeader
  2 match http header TestHeader header-value ".*"
policy-map type loadbalance first-match PLBSF_STICKY-HEADER
  class MSISDN
    sticky-serverfarm HEADER-GROUP-42
  class TestHeader
    sticky-serverfarm HEADER-GROUP-41
  class class-default
    serverfarm DEFAULT
policy-map multi-match SH-Gold-VIPs
  class STICKY-HEADER_129:80
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-HEADER
    loadbalance vip icmp-reply active
    nat dynamic 1 VLAN 30
    appl-parameter http advanced-options PERSIST-REBALANCE
```

Test Procedure

The procedure used to perform the Header Sticky —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Start a series of HTTP connections using the same MSISDN headers.

Step 4 Verify that there are two sticky entries in the database for the sticky group that matches the server(s) that is getting the connections.

The traffic started in an earlier step, is sending two unique headers which result in either one or two servers getting all of the connections. Sticky ensures that no matter how many times these GET requests are sent (within the timeout period) the server(s) chosen with the first iteration will always get the connections.

Step 5 Begin another series of HTTP connections this time using a static custom header.

Step 6 Verify that there is a sticky entry in the database for the sticky group that matches the server getting the connections.

The test tool is sending a static header value, so only one server will get all of the connections. Sticky ensures that no matter how many times these GET requests are sent (within the timeout period) the server(s) chosen with the first iteration will always get the connections.

Step 7 Start another series of HTTP connections this time using a static custom header.

Step 8 This time the header value is different, so a new sticky entry will be created. Verify that there is second sticky entry in the database for the sticky group that matches the server getting the connections.

Step 9 Clear select counters on the load balancer.

Step 10 On a Linux client, change the MTU on the client to 272. This will cause the MSISDN header to be split across two packets in the middle of the header name. Generate a series of HTTP connections using the same MSISDN headers.

Step 11 Verify that there are two sticky entries in the database for the sticky group that matches the server(s) getting the connections.

Step 12 Clear select counters on the load balancer.

Step 13 On a Linux client, change the MTU on the client to 276. This will cause the MSISDN header to be split across two packets just before the colon. Begin a series of HTTP connections using the same MSISDN header to the serverfarm.

Step 14 Verify that there are two sticky entries in the database for the sticky group that matches the server(s) getting the connections.

Step 15 With the client MTU still set to 276, the header will be split across two packets in the middle of the header name.

Step 16 Verify that there is a sticky entry in the database for the sticky group that matches the server getting the connections.

Step 17 Clear select counters on the load balancer.

Step 18 On a Linux client, change the MTU on the client to 280. This will cause the header to be split across two packets just before the colon. Begin a series of HTTP connections using a custom header.

Step 19 Verify that there is a sticky entry in the database for the sticky group that matches the server getting the connections in the serverfarm.

Step 20 On the Linux client change the MTU back to the default value of 1500. Clear select counters on the load balancer.

Step 21 Change the sticky timeout on the sticky group to one minute.

Step 22 Begin a series of HTTP connections using the same MSISDN header to the serverfarm STICKY-HEADER. Verify that there are two sticky entries in the database for the sticky group that matches the server(s) getting the connections.

Step 23 Wait for the sticky connection to timeout and be removed from the database.

Step 24 Begin a series of HTTP connections using the same MSISDN header to the serverfarm STICKY-HEADER.

Step 25 Verify that there are two different sticky entries in the database for the sticky group matching the server(s) getting the connections.

Step 26 Change the sticky timeout back to 30 minutes for the sticky group.

Expected Results

The following test results are anticipated:

- We expect a client to remain stuck to a server based upon the sticky configuration.
- We expect a client to remain stuck even if the header being parsed is split across two packets.
- We expect the sticky entry to be purged after the timer expires and to be load balanced and stuck to a new server.
- We expect that the load balancer will not crash or become unresponsive.

Results

Header Sticky —ACE passed.

Header Sticky—CSM

Server stickiness allows a particular connection to be maintained with a specific rserver of a serverfarm for its duration. If Host A requests a page from http://www.myurl.com, which is being served up by a farm of servers, that host can be paired with a single server for the duration of its connection. Stickiness can be created based on a number of different criteria, including a cookie field, Header, IP addr, etc. The Load Balancer maintains a database of client-to-real server pairings which can then be reused for subsequent connections for the timeout period.

This test verified that the Load Balancer stuck to a particular server based upon a well known HTTP header or a custom one that was user defined.

Relevant Load Balancer Configuration

```
!
 serverfarm STICKY-HEADER
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-154
   inservice
  real name LOCAL-IIS-245
   inservice
  real name LOCAL-LINUX-242
   inservice
  probe HTTP
!
 serverfarm STICKY-HEADER2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
  probe HTTP
!
 sticky 40 header MSISDN timeout 30
!
 sticky 41 header TestHeader timeout 30
  header offset 15 length 7
!
 policy STICKY-HEADER
  sticky-group 40
  serverfarm STICKY-HEADER
!
 policy STICKY-HEADER2
  sticky-group 41
  serverfarm STICKY-HEADER2
!
 vserver STICKY-HEADER
  virtual 192.168.120.201 tcp www
  serverfarm DEFAULT
  persistent rebalance
  slb-policy STICKY-HEADER
  inservice
!
```

Test Procedure

The procedure used to perform the Header Sticky —CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Begin a series of HTTP connections using the same MSISDN header.

Step 4 Verify that sticky relationships are being maintained in the sticky database and that the connections are stuck to two servers.

Step 5 Replace the current sticky policy with another sticky policy which uses a different header.

Step 6 Begin another series of HTTP connections this time using a static custom header.

Step 7 Verify that sticky relationships are being maintained in the sticky database and that the connections are stuck to a single server.

Step 8 Clear select counters on the load balancer.

Step 9 Replace the current sticky policy with another sticky policy which uses a different header.

Step 10 On a Linux client, change the MTU on the client to 272. This will cause the MSISDN header to be split across two packets in the middle of the header name. Begin a series of HTTP connections using the same MSISDN header as was used in a previous test step.

Step 11 Verify that sticky relationships are being maintained in the sticky database and that the connections are stuck to two servers.

Step 12 Clear select counters on the load balancer.

Step 13 On a Linux client, change the MTU on the client to 276. This will cause the MSISDN header to be split across two packets just before the colon. Begin a series of HTTP connections using the same MSISDN header to the Virtual IP Address (VIP).

Step 14 Verify that sticky relationships are being maintained in the sticky database and that the connections are stuck to two servers.

Step 15 Replace the current sticky policy with another sticky policy which uses a different header.

Step 16 With the client MTU still set to 276, the header will be split across two packets in the middle of the header name. Begin a series of HTTP connections using a custom header to the VIP.

Step 17 Verify that there is a sticky entry in the database for the sticky group that matches the server getting the connections.

Step 18 Clear select counters on the load balancer.

Step 19 On a Linux client, change the MTU on the client to 280. This will cause the header to be split across two packets just before the colon. Begin a series of HTTP connections using a custom header.

Step 20 Verify that there is a sticky entry in the database for the sticky group that matches the server getting the connections.

Step 21 On the Linux client change the MTU back to the default value of 1500. Clear select counters on the load balancer.

Step 22 Change the sticky timeout on the sticky group one minute.

Step 23 Replace the current sticky policy with another sticky policy which uses a different header.

Step 24 Send an HTTP connection using the same MSISDN header to the VIP.

Step 25 Wait for the sticky connection to timeout and be removed from the database.

Step 26 Start a new connection using the same TestHeader header and verify that the sticky table gets populated with an entry using a different server.

Step 27 Verify that there are two different sticky entries in the database for the sticky group matching the server(s) getting the connections.

Step 28 Change the sticky timeout back to 30 minutes for the sticky group.

Expected Results

The following test results are anticipated:

- We expect a client to remain stuck to a server based upon the sticky configuration.
- We expect a client to remain stuck even if the header being parsed is split across two packets.

- We expect the sticky entry to be purged after the timer expires and to be load balanced and stuck to a new server.
- We expect that the load balancer will not crash or become unresponsive.

Results

Header Sticky —CSM passed.

IP Netmask Sticky—ACE

IP address stickiness allows you to stick a client to the same server for multiple subsequent connections as needed to complete a transaction using the client source IP address, the destination IP address, or both.

This test verified that server stickiness was maintained based upon its configuration.

Relevant Load Balancer Configuration

```
sticky ip-netmask 255.255.255.255 address both STKY-GRP-33
  timeout 20
  replicate sticky
  serverfarm STICKY-NETMASK
serverfarm host STICKY-NETMASK
  probe ICMP
  rserver BRG-12
    inservice
  rserver LOCAL-242
    inservice
  rserver RT-153
    inservice
class-map match-any STICKY-IP_115:UDP-TCP
  2 match virtual-address 192.168.120.115 udp any
  3 match virtual-address 192.168.120.115 tcp any
policy-map type loadbalance first-match PLBSF_STICKY-NETMASK
  class class-default
    sticky-serverfarm STKY-GRP-33
policy-map multi-match SH-Gold-VIPs
  class STICKY-IP_115:UDP-TCP
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-NETMASK
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
```

Test Procedure

The procedure used to perform the IP Netmask Sticky —ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Verify that the sticky group is configured as IP sticky both. This is a netmask sticky using a hash of both the source and destination IP addresses of the incoming connection request.

Step 3 Clear select counters on the load balancer.

Step 4 Begin a series of HTTP connections using a range of seven source IP addresses to the Virtual IP Address (VIP).

Step 5 With three servers configured and seven source IP's used, two servers will get two sticky entries and one server will get three. Verify that seven sticky entries are in the sticky database in this manner and that one server gets the bulk of the connections.

Step 6 Clear select counters on the load balancer.

Step 7 Change the sticky group to use the destination IP of the incoming connection request.

Step 8 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 9 Verify that only one sticky entry is created in the sticky database and that this matches the server getting all of the connections in the serverfarm. With destination IP sticky all traffic is stuck to one server because all traffic is hitting the same destination IP.

Step 10 Clear select counters on the load balancer.

Step 11 Change the IP sticky group to use the source IP of the incoming connection request for the hash.

Step 12 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 13 With three servers configured and seven source IP's used, two servers will get two sticky entries and one server will get three. Verify that seven sticky entries are in the sticky database in this manner and that one server gets the bulk of the connections.

Step 14 Clear only the serverfarm counters.

Step 15 Take one of the servers out of service.

Step 16 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 17 Verify that no sticky entries are associated with the server previously taken out of service and that no connections were load balanced to this server.

Step 18 Bring the server that was taken out of service back to an operational state.

Step 19 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 20 Verify that the sticky entries did not change and this newly activated server did not receive any connections. When a server is no longer available the entry is moved to a server that is able to service the request. When the original server is again available the sticky entry is not moved back to that server. New connections which do not have a sticky entry can be sent to the recently available server.

Step 21 Begin a series of HTTP connections using a different range of seven source IP addresses to the VIP.

Step 22 Verify that the server recently brought back into service now has sticky entries and handled connections.

Step 23 Return the sticky group hash setting back to both.

Expected Results

The following test results are anticipated:

- We expect the connections to remain stuck to the real based on the configured IP source, destination, and source/destination sticky.
- We expect the sticky entry to change only when the original real becomes unavailable.
- We expect that the load balancer will not crash or become unresponsive.

Results

IP Netmask Sticky —ACE passed.

IP Netmask Sticky—CSM

IP address stickiness allows you to stick a client to the same server for multiple subsequent connections as needed to complete a transaction using the client source IP address, the destination IP address, or both.

This test verified that server stickiness was maintained based upon its configuration.

Relevant Load Balancer Configuration

```
!
 real LOCAL-IIS-245
  address 172.29.0.245
  inservice
 real RT-LINUX-154
  address 172.28.0.154
  inservice
!
 serverfarm STICKY-MASK
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-154
   inservice
!
 sticky 30 netmask 255.255.255.255 address source timeout 30
 sticky 10 netmask 255.255.255.255 address both timeout 10
 sticky 20 netmask 255.255.255.255 address destination timeout 30
!
 vserver STICKY-MASK
  virtual 192.168.120.225 tcp www
  serverfarm STICKY-MASK
  sticky 30 group 10
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the IP Netmask Sticky—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear select counters on the load balancer.

Step 3 Modify the load balancer configuration to use a netmask sticky using a hash of both the source IP and destination IP of the incoming connection request. Verify that the sticky group has been applied to the load balancer configuration.

Step 4 Begin a series of HTTP connections using a range of seven source IP addresses to the Virtual IP Address (VIP).

Step 5 With three servers configured and seven source IP's used, two servers will get two sticky entries and one server will get three. Verify that seven sticky entries are in the sticky database in this manner and that one server gets the bulk of the connections.

Step 6 Clear select counters on the load balancer.

Step 7 Change the sticky group to use the destination IP of the incoming connection request.

Step 8 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 9 Verify that only one sticky entry is created in the sticky database and that this matches the server getting all of the connections in the serverfarm. With destination IP sticky all traffic is stuck to one server because all traffic is hitting the same destination IP.

Step 10 Clear select counters on the load balancer.

Step 11 Change the IP sticky group to use the source IP of the incoming connection request for the hash.

Step 12 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 13 With three servers configured and seven source IP's used, two servers will get two sticky entries and one server will get three. Verify that seven sticky entries are in the sticky database in this manner and that one server gets the bulk of the connections.

Step 14 Clear select counters on the load balancer.

Step 15 Take one of the servers out of service.

Step 16 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 17 Verify that no sticky entries are associated with server previously taken out of service and that no connections were load balanced to this server.

Step 18 Bring the server that was taken out of service back to an operational state.

Step 19 Begin a series of HTTP connections using a range of seven source IP addresses to the VIP.

Step 20 Verify that no sticky entries are associated with server previously taken out of service and that no connections were load balanced to this server.

Expected Results

The following test results are anticipated:

- We expect the connections to remain stuck to the real based on the configured IP source, destination, and source/destination sticky.
- We expect the sticky entry to change only when the original real becomes unavailable.
- We expect that the load balancer will not crash or become unresponsive.

Results

IP Netmask Sticky—CSM passed.

SSL Session ID Sticky—ACE

SSL Session-ID Stickiness allows a load balancer to stick the same client to the same SSL server based on the SSL Session ID for SSLv3 and TLS 1.0. If the SSL Session ID remains the same across multiple connections from the same client, you can use this feature to stick clients to a particular load balanced SSL server.

This test verified that the load balancer will stick SSLv3 and TLS1.0 requests to the same server when the same SSL ID is used and that sticky entries were replicated to the standby.

Relevant ACE Configuration

```
probe tcp TCP:443
  port 443
  interval 5
  passdetect interval 10
  connection term forced
  open 3
parameter-map type generic SSL-PARSE
  set max-parse-length 70
serverfarm host SSL-SESSION
  failaction purge
  predictor leastconns
  probe TCP:443
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-11
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-IIS-245
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
sticky layer4-payload SSL-SESSION_STKY
  timeout 600
  serverfarm SSL-SESSION
  response sticky
  layer4-payload offset 43 length 32 begin-pattern "\x20"
class-map type generic match-any SSLID-32_REGEX
  2 match layer4-payload regex "\x16\x03[\x00\x01]..[\x01\x02].*"
  3 match layer4-payload regex "\x80\x4c.*"
class-map match-any SSL-SESSION-VIP_106:443
  match virtual-address 192.168.140.106 tcp eq https
policy-map type loadbalance generic first-match PLBSF_SSL-SESSION
  class SSLID-32_REGEX
  sticky-serverfarm SSL-SESSION_STKY
policy-map multi-match A2-VIPS
  class SSL-SESSION-VIP_106:443
    loadbalance vip advertise active
    loadbalance vip inservice
    loadbalance vip icmp-reply active
    loadbalance policy PLBSF_SSL-SESSION
    appl-parameter generic advanced-options SSL-PARSE
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
```

Test Procedure

The procedure used to perform the SSL Session ID Sticky—ACE test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 From a Linux client send two streams of SSL 3.0 and 3.1 traffic reusing the same SSL session ID for each connection.

Step 4 Verify that there are two sticky entries created and that all of the connections were load balanced to only these servers.

Step 5 Verify on the standby load balancer that the same two sticky entries were created.

Step 6 Clear the relevant counters on the primary ACE module.

Step 7 From a Linux client send two streams of SSL 3.0 and 3.1 traffic using a different SSL session ID for each connection.

Step 8 Verify that there are two hundred new sticky entries are created and that all of the connections were load balanced across all of the servers.

Step 9 Clear the relevant counters.

Step 10 From a Windows client configure an IE browser to use a hybrid SSL hello. A hybrid hello has an SSLv2.0 record format, but requests a SSLv3 or TLS1.0 session.

Step 11 Start a packet capture on the Windows client.

Step 12 On the IE browser connect to https://192.168.140.106/. After a successful browser request refresh the browser several more times.

Step 13 Verify that a sticky entry is created and the requests serviced by a single rserver.

Step 14 Stop the packet capture and verify that the first client hello is in a hybrid format.

```
Secure Socket Layer
    SSLv2 Record Layer: Client Hello
        [Version: SSL 2.0 (0x0002)]
        Length: 76
        Handshake Message Type: Client Hello (1)
        Version: TLS 1.0 (0x0301)
        Cipher Spec Length: 51
        Session ID Length: 0
        Challenge Length: 16
        Cipher Specs (17 specs)
        Challenge
```

Step 15 On the Windows client return the IE browser configuration to stop using a hybrid SSL hello.

Expected Results

The following test results are anticipated:

- We expect the load balancer to stick to an rserver based on the SSL Session ID for SSLv3 and TLS1.0 traffic.
- We expect the load balancer to replicate the sticky entries to the standby.

Results

SSL Session ID Sticky—ACE passed.

SSL Session ID Sticky—CSM

SSL Session-ID Stickiness allows a load balancer to stick the same client to the same SSL server based on the SSL Session ID for SSLv3 and TLS 1.0. If the SSL Session ID remains the same across multiple connections from the same client, you can use this feature to stick clients to a particular load balanced SSL server.

This test verified that the load balancer will stick SSLv3 and TLS1.0 requests to the same server when the same SSL ID is used and that sticky entries were replicated to the standby.**Relevant CSM config**

```
!
 ft group 2 vlan 900
  priority 110 alt 100
  preempt
  track group hsrp-Vl120-120
  track gateway 192.168.16.251
  track interface GigabitEthernet4/37
  track mode any
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 105 client
  ip address 192.168.105.12 255.255.255.0 alt 192.168.105.13 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
  ip address 10.86.83.13 255.255.255.0 alt 10.86.83.14 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 vlan 121 server
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  alias 192.168.120.7 255.255.255.0
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 serverfarm STICKY-SSL
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-IIS-152
   inservice
  real name LOCAL-IIS-245
   inservice
!
sticky 110 ssl timeout 30
!
 vserver STICKY-SSL
  virtual 192.168.120.226 tcp https
  serverfarm STICKY-SSL
  sticky 30 group 110
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
```

Test Procedure

The procedure used to perform the SSL Session ID Sticky—CSM test follows:

Step 1 Connect to the DUT (Device Under Test)

Step 2 Clear the relevant counters.

Step 3 From a Linux client send two streams of SSL 3.0 and 3.1 traffic reusing the same SSL session ID for each connection.

Step 4 Verify that there are two sticky entries created and that all of the connections were load balanced to only these servers.

Step 5 Verify on the standby load balancer that the same two sticky entries were created.

Step 6 Clear the relevant counters.

Step 7 From a Linux client send two streams of SSL 3.0 and 3.1 traffic using a different SSL session ID for each connection.

Step 8 Verify that there are two hundred new sticky entries are created and that all of the connections were load balanced across all of the servers.

Step 9 Verify on the standby load balancer that two hundred sticky entries were created.

Step 10 Clear the relevant counters.

Step 11 From a Windows client configure an IE browser to use a hybrid SSL hello. A hybrid hello has an SSLv2.0 record format, but requests a SSLv3 or TLS1.0 session.

Step 12 Start a packet capture on the Windows client.

Step 13 On the IE browser connect to https://192.168.120.226/. After a successful browser request refresh the browser several more times.

Step 14 Verify that a sticky entry is created and the requests serviced by a single rserver.

Step 15 Stop the packet capture and verify that the first client hello is in a hybrid format.

```
Secure Socket Layer
    SSLv2 Record Layer: Client Hello
        [Version: SSL 2.0 (0x0002)]
        Length: 76
        Handshake Message Type: Client Hello (1)
        Version: TLS 1.0 (0x0301)
        Cipher Spec Length: 51
        Session ID Length: 0
        Challenge Length: 16
        Cipher Specs (17 specs)
        Challenge
```

Step 16 On the Windows client return the IE browser configuration to stop using a hybrid SSL hello.

Expected Results

The following test results are anticipated:

- We expect the load balancer to stick to an rserver based on the SSL Session ID for SSLv3 and TLS1.0 traffic.
- We expect the load balancer to replicate the sticky entries to the standby.

Results

SSL Session ID Sticky—CSM passed.

Topology Configurations

The following configurations are provided for testing considerations. See Test Cases, page 11 for individual testing procedures.

- Basic Topology: Modules, page 208
- Basic Topology: Test Device Configurations, page 212

Basic Topology: Modules

The following devices were used in the Safe Harbor testbed.

Cat 6500 Device Configurations

- SH-ACE2-6K-1, page 208
- SH-ACE2-6K-2, page 209
- SH-ACE2-6K-3, page 210
- SH-ACE2-6K-4, page 211

SH-ACE2-6K-1

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

Show Module

```
sh-ace2-6k-1#show module
Load for five secs: 2%/0%; one minute: 1%; five minutes: 1%
Time source is NTP, 09:52:43.153 EDT Fri Oct 17 2008

Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------
  1    1  Application Control Engine Module      ACE10-6500-K9      SAD110207CN
  2    8  Network Analysis Module                WS-SVC-NAM-2       SAD105000MY
  3    1  SSL Module                             WS-SVC-SSL-1       SAD0716028B
  4   48  CEF720 48 port 10/100/1000mb Ethernet  WS-X6748-GE-TX     SAL10499MLP
  5    2  Supervisor Engine 720 (Hot)            WS-SUP720-3B       SAL1127TJ0E
  6    2  Supervisor Engine 720 (Active)         WS-SUP720-3B       SAL1051BRW0
  7   16  16 port 1000mb GBIC ethernet           WS-X6416-GBIC      SAD04240683
  8    4  SLB Application Processor Complex      WS-X6066-SLB-APC   SAD08120167
  9    8  CEF720 8 port 10GE with DFC            WS-X6708-10GE      SAL1050AZVL

Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  1  001a.6d66.0e3c to 001a.6d66.0e43   1.3   8.7(0.22)ACE A2(1.2)      Ok
  2  0019.e78f.0af4 to 0019.e78f.0afb   4.2   7.2(1)       3.5(1a)      Ok
  3  0003.feab.7e76 to 0003.feab.7e7d   2.0   7.2(1)       2.1(10)      Ok
  4  0019.e8e6.b510 to 0019.e8e6.b53f   2.5   12.2(14r)S5  12.2(18)SXF1 Ok
  5  001a.2f3b.f5a0 to 001a.2f3b.f5a3   5.4   8.4(2)       12.2(18)SXF1 Ok
  6  0013.c347.6a50 to 0013.c347.6a53   5.3   8.4(2)       12.2(18)SXF1 Ok
  7  00d0.c0cf.a32c to 00d0.c0cf.a33b   1.1   5.3(1)       8.5(0.46)RFW Ok
  8  0003.feae.cd2c to 0003.feae.cd33   1.7                4.2(6)       Ok
  9  0018.b966.e110 to 0018.b966.e117   1.3   12.2(18r)S1  12.2(18)SXF1 Ok

Mod  Sub-Module                  Model              Serial       Hw     Status
---- --------------------------- ------------------ ----------- ------- -------
```

```
  4  Distributed Forwarding Card WS-F6700-DFC3B     SAL1049A29M  4.4    Ok
  5  Policy Feature Card 3       WS-F6K-PFC3B       SAL1127TE22  2.3    Ok
  5  MSFC3 Daughterboard         WS-SUP720          SAL1127TJLL  3.0    Ok
  6  Policy Feature Card 3       WS-F6K-PFC3B       SAL1051BNYP  2.3    Ok
  6  MSFC3 Daughterboard         WS-SUP720          SAL1051BTZU  2.6    Ok
  9  Distributed Forwarding Card WS-F6700-DFC3C     SAD10490ARP  1.0    Ok

Mod  Online Diag Status
---- -------------------
  1  Pass
  2  Pass
  3  Pass
  4  Pass
  5  Pass
  6  Pass
  7  Pass
  8  Pass
  9  Pass
sh-ace2-6k-1#
sh-ace2-6k-1#term length 24
sh-ace2-6k-1#term width 80
sh-ace2-6k-1#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k-1(config)#logging console
sh-ace2-6k-1(config)#end
sh-ace2-6k-1#
```

Return to: SH-ACE2-6K-1, page 208

Return to: Test Results Summary, page 5

SH-ACE2-6K-2

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

Show Module

```
sh-ace2-6k-2#show module
Load for five secs: 5%/2%; one minute: 1%; five minutes: 1%
Time source is NTP, 09:52:43.216 EDT Fri Oct 17 2008

Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------
  1    1  Application Control Engine Module      ACE10-6500-K9      SAD1051048L
  2    8  Network Analysis Module                WS-SVC-NAM-2       SAD111101R2
  3    1  SSL Module                             WS-SVC-SSL-1       SAD0909041F
  4   48  CEF720 48 port 10/100/1000mb Ethernet  WS-X6748-GE-TX     SAL1049A456
  5    2  Supervisor Engine 720 (Active)         WS-SUP720-3B       SAL1103EE98
  7   16  16 port 1000mb GBIC ethernet           WS-X6416-GBIC      SAL08384B8E
  8    4  SLB Application Processor Complex      WS-X6066-SLB-APC   SAD08160B4S
  9    8  CEF720 8 port 10GE with DFC            WS-X6708-10GE      SAD1128014C

Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  1  001a.6d65.f760 to 001a.6d65.f767   1.3    8.7(0.22)ACE A2(1.2)      Ok
  2  001b.2a65.5d74 to 001b.2a65.5d7b   4.2    7.2(1)       3.5(1b)      Ok
  3  0013.8054.ed80 to 0013.8054.ed87   3.2    7.2(1)       2.1(10)      Ok
  4  0019.e8e6.dd90 to 0019.e8e6.ddbf   2.5    12.2(14r)S5  12.2(18)SXF1 Ok
  5  0016.46f9.0d78 to 0016.46f9.0d7b   5.3    8.4(2)       12.2(18)SXF1 Ok
  7  0012.0182.1238 to 0012.0182.1247   2.6    5.4(2)       8.5(0.46)RFW Ok
  8  000f.905c.8284 to 000f.905c.828b   1.7                 4.2(6)       Ok
  9  001c.5843.89c0 to 001c.5843.89c7   1.3    12.2(18r)S1  12.2(18)SXF1 Ok
```

```
Mod  Sub-Module                  Model              Serial       Hw     Status
---- --------------------------- ------------------ ----------- ------- -------
  4  Distributed Forwarding Card WS-F6700-DFC3B     SAL10478BF2  4.4    Ok
  5  Policy Feature Card 3       WS-F6K-PFC3B       SAL1103E4X2  2.3    Ok
  5  MSFC3 Daughterboard         WS-SUP720          SAL1103EM4K  2.6    Ok
  9  Distributed Forwarding Card WS-F6700-DFC3C     SAL1127TEKL  1.0    Ok

Mod  Online Diag Status
---- -------------------
  1  Pass
  2  Pass
  3  Pass
  4  Pass
  5  Pass
  7  Pass
  8  Pass
  9  Pass
sh-ace2-6k-2#
sh-ace2-6k-2#term length 24
sh-ace2-6k-2#term width 80
sh-ace2-6k-2#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k-2(config)#logging console
sh-ace2-6k-2(config)#end
sh-ace2-6k-2#
```

Return to: SH-ACE2-6K-2, page 209

Return to: Test Results Summary, page 5

SH-ACE2-6K-3

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

Show Module

```
sh-ace2-6k3#show module
Load for five secs: 1%/1%; one minute: 1%; five minutes: 1%
Time source is hardware calendar, *13:25:46.608 UTC Fri Oct 17 2008

Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------
  2    8  Network Analysis Module                WS-SVC-NAM-2       SAD105103W9
  4   48  CEF720 48 port 10/100/1000mb Ethernet  WS-X6748-GE-TX     SAL104891HR
  5   16  16 port 1000mb GBIC ethernet           WS-X6416-GBIC      SAL08384B4W
  6    2  Supervisor Engine 720 (Active)         WS-SUP720-3B       SAL1051BRVG

Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  2  001a.6c03.95ce to 001a.6c03.95d5   4.2   7.2(1)       3.5(1a)      Ok
  4  001a.2f08.9f90 to 001a.2f08.9fbf   2.5   12.2(14r)S5  12.2(18)SXF1 Ok
  5  0012.016a.e5c0 to 0012.016a.e5cf   2.6   5.4(2)       8.5(0.46)RFW Ok
  6  0017.9444.3378 to 0017.9444.337b   5.3   8.4(2)       12.2(18)SXF1 Ok

Mod  Sub-Module                  Model              Serial       Hw     Status
---- --------------------------- ------------------ ----------- ------- -------
  4  Distributed Forwarding Card WS-F6700-DFC3B     SAL10478GZU  4.4    Ok
  6  Policy Feature Card 3       WS-F6K-PFC3B       SAL1051BQVS  2.3    Ok
  6  MSFC3 Daughterboard         WS-SUP720          SAL1051BTTD  2.6    Ok

Mod  Online Diag Status
```

```
---- -------------------
   2  Pass
   4  Pass
   5  Pass
   6  Pass
sh-ace2-6k3#
sh-ace2-6k3#term length 24
sh-ace2-6k3#term width 80
sh-ace2-6k3#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k3(config)#logging console
sh-ace2-6k3(config)#end
sh-ace2-6k3#
```

Return to: SH-ACE2-6K-3, page 210

Return to: Test Results Summary, page 5

SH-ACE2-6K-4

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

Show Module

```
sh-ace2-6k4#show module
Load for five secs: 0%/0%; one minute: 0%; five minutes: 0%
Time source is hardware calendar, *13:38:45.892 UTC Fri Oct 17 2008

Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------
  2    8  Network Analysis Module                WS-SVC-NAM-2       SAD111101T9
  4   48  SFM-capable 48 port 10/100/1000mb RJ45 WS-X6548-GE-TX     SAD074304R7
  5    4  CEF720 4 port 10-Gigabit Ethernet      WS-X6704-10GE      SAL10392HNQ
  6    2  Supervisor Engine 720 (Active)         WS-SUP720-3B       SAL1052C1L6

Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  2  001b.2a65.5f3c to 001b.2a65.5f43   4.2   7.2(1)       3.5(1b)      Ok
  4  000d.edb5.1268 to 000d.edb5.1297   7.2   7.2(1)       8.5(0.46)RFW Ok
  5  0019.06da.cc64 to 0019.06da.cc67   2.5   12.2(14r)S5  12.2(18)SXF1 Ok
  6  0017.9444.3524 to 0017.9444.3527   5.3   8.4(2)       12.2(18)SXF1 Ok

Mod  Sub-Module                  Model              Serial       Hw     Status
---- --------------------------- ------------------ ----------- ------- -------
  5  Centralized Forwarding Card WS-F6700-CFC       SAL10360TAT  3.0    Ok
  6  Policy Feature Card 3       WS-F6K-PFC3B       SAL1051BSA9  2.3    Ok
  6  MSFC3 Daughterboard         WS-SUP720          SAL1052BXUJ  2.6    Ok

Mod  Online Diag Status
---- -------------------
  2  Pass
  4  Pass
  5  Pass
  6  Pass
sh-ace2-6k4#
sh-ace2-6k4#term length 24
sh-ace2-6k4#term width 80
sh-ace2-6k4#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k4(config)#logging console
sh-ace2-6k4(config)#end
sh-ace2-6k4#
```

Return to: SH-ACE2-6K-4, page 211

Return to: Test Results Summary, page 5

Basic Topology: Test Device Configurations

The following configuration files were used in the Safe Harbor testbed for the following devices tested.

Cat 6500 Device Configurations

- SH-ACE2-6K-1, page 212
- SH-ACE2-6K-2, page 247
- SH-ACE2-6K-3, page 280
- SH-ACE2-6K-4, page 290

ACE Module Configurations

- 6K-1 ACE1 All Context, page 298
- 6K-2 ACE1 All Context, page 416

SH-ACE2-6K-1

Go to Topology Configurations, page 208

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

```
sh-ace2-6k-1#show running-config
Load for five secs: 2%/1%; one minute: 1%; five minutes: 1%
Time source is NTP, 09:59:11.192 EDT Fri Oct 17 2008

Building configuration...

Current configuration : 42889 bytes
!
! Last configuration change at 09:58:41 EDT Fri Oct 17 2008 by cisco
!
upgrade fpd auto
version 12.2
service timestamps debug datetime msec localtime
service timestamps log datetime msec localtime
no service password-encryption
service internal
service counters max age 5
!
hostname sh-ace2-6k-1
!
boot system flash disk0:s72033-adventerprisek9_wan-mz.122-18.SXF13.bin
logging buffered 256000 debugging
!
username cisco secret 5 $1$R3AT$TCtBy3t1AfYQ7gR3mMMnS0
aaa new-model
!
aaa session-id common
clock timezone EST -5
clock summer-time EDT recurring
clock calendar-valid
```

```
svclc autostate
svclc multiple-vlan-interfaces
svclc module 1 vlan-group 11111
svclc vlan-group 11111  2,16,29,31,83,97-99,105,106,120,130,160,281,283,320
svclc vlan-group 11111  329,900,2830
analysis module 2 management-port access-vlan 83
ssl-proxy module 3 allowed-vlan 29,83,121
ip subnet-zero
!
!
!
ip ftp username lab
ip ftp password labtest1
ip ssh time-out 60
ip ssh authentication-retries 2
ip domain-name cisco.com
ipv6 mfib hardware-switching replication-mode ingress
vtp mode transparent
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
no mls acl tcam share-global
mls cef error action freeze
!
!
!
!
!
!
!
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
!
spanning-tree mode rapid-pvst
spanning-tree extend system-id
spanning-tree vlan 1-79,90-900 priority 24576
module ContentSwitchingModule 8
 variable GSLB_LICENSE_KEY Q1NDT0dTTEJCRU9XVUxG
 variable REAL_SLOW_START_ENABLE 3
 variable SASP_FIRST_BIND_ID 1000
 variable SASP_GWM_BIND_ID_MAX 8
!
 ft group 2 vlan 900
  priority 110 alt 100
  preempt
  track gateway 192.168.16.251
  track interface GigabitEthernet4/37
  track mode any
!
 vlan 120 client
  ip address 192.168.120.8 255.255.255.0 alt 192.168.120.9 255.255.255.0
  gateway 192.168.120.254
  alias 192.168.120.7 255.255.255.0
!
 vlan 29 server
  ip address 172.29.0.12 255.255.255.0 alt 172.29.0.13 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 105 client
  ip address 192.168.105.12 255.255.255.0 alt 192.168.105.13 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
  ip address 10.86.83.13 255.255.255.0 alt 10.86.83.14 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 script file disk0:c6slb-apc.tcl
!
 probe ICMP icmp
  interval 10
  retries 2
  failed 10
  receive 3
!
 probe HTTP-PROBE http
  interval 5
  failed 10
  open 3
  receive 5
  port 80
!
 probe TCP tcp
  interval 10
  retries 2
  failed 10
  open 3
  port 53
!
 probe TELNET telnet
  interval 10
  retries 2
  failed 10
  open 3
  receive 5
  port 23
!
 probe FTP ftp
  interval 10
  retries 2
  failed 10
  open 3
  receive 5
  port 21
!
 probe SSLPROBE script
  script SSL_PROBE_SCRIPT 0
  interval 5
  failed 10
!
 probe TCP-GENERIC tcp
  interval 5
  failed 10
!
 probe HTTP http
  interval 5
  failed 10
  open 3
  receive 5
!
 probe PRED-PING icmp
  address 172.29.0.1 routed
  interval 5
  retries 2
  failed 3
!
 probe ICMP-RTR icmp
  address 172.29.0.1 routed
  interval 5
  failed 5
!
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 map REDIRECT-1K url
  match protocol http url *redirect-1k.html
!
 map REDIRECT-10K url
  match protocol http url *redirect-10k.html
!
 map REDIRECT-100K url
  match protocol http url *redirect-100k.html
!
 map 64K url
  match protocol http url *64k*
!
 map 16K url
  match protocol http url *16k*
!
 map 32K url
  match protocol http url *32k*
!
 map 128K url
  match protocol http url *128k*
!
 map 512K url
  match protocol http url *512k*
!
 map COOKIE-MAP cookie
  match protocol http cookie CSM_TEST cookie-value This*is*a*test0
!
 map BROWSER_MOZILLA header
  match protocol http header User-Agent header-value *Mozilla*
!
 map BROWSER_MSIE header
  match protocol http header User-Agent header-value *MSIE*
!
 map PARSE-LENGTH header
  match protocol http header Accept-Charset header-value *utf-8
!
 map M-ORDER1 header
  match protocol http header TestHeader header-value *p-order1*
!
 map M-ORDER2 header
  match protocol http header TestHeader header-value *p-order2*
!
 map M-ORDER3 header
  match protocol http header TestHeader header-value *p-order3*
!
 map M-ORDER4 header
```

```
  match protocol http header TestHeader header-value *p-order4*
!
 map M-ORDER5 header
  match protocol http header TestHeader header-value *p-order5*
!
 map M-ORDER6 header
  match protocol http header User-Agent header-value MSIE
!
 map M-ORDER7 header
  match protocol http header TestHeader header-value *p-order7*
!
 map M-MAX-CONN url
  match protocol http method GET url /index.html
!
 map M-GSLB-SH dns
  match protocol dns domain www.sh1.com
  match protocol dns domain www.sh2.com
  match protocol dns domain www.sh3.com
!
 map M-GSLB-NS dns
  match protocol dns domain .*
!
 map INDEX.HTML url
  match protocol http method GET url /index.html*
!
 map M-HDR-INSERT header
  insert protocol http header Source-IP header-value %is
  insert protocol http header Accept header-value anything
  insert protocol http header Pragma header-value "Pragma no Pragma that is the question"
  insert protocol http header Destination_IP header-value %id
!
 map M-HDR-SRCDST-IP header
  insert protocol http header Source-IP header-value %is
  insert protocol http header Destination_IP header-value %id
!
 real 6K-3
  address 192.168.105.253
  location "needed for arp entry, CSCej81417"
  inservice
 real BRG-LINUX-11
  address 192.168.120.11
  inservice
 real BRG-LINUX-12
  address 192.168.120.12
  inservice
 real BRG-LINUX-13
  address 192.168.120.13
  inservice
 real BRG-LINUX-14
  address 192.168.120.14
  inservice
 real BRG-LINUX-15
  address 192.168.120.15
  inservice
 real IXIA-25
  address 172.31.20.250
  inservice
 real IXIA-26
  address 172.31.101.250
  inservice
 real LOCAL-IIS-241
  address 172.29.0.241
  inservice
 real LOCAL-IIS-243
```

```
  address 172.29.0.243
  inservice
 real LOCAL-IIS-245
  address 172.29.0.245
  inservice
 real LOCAL-LINUX-240
  address 172.29.0.240
  inservice
 real LOCAL-LINUX-242
  address 172.29.0.242
  inservice
 real LOCAL-LINUX-244
  address 172.29.0.244
  inservice
 real RT-IIS-152
  address 172.28.0.152
  inservice
 real RT-LINUX-151
  address 172.28.0.151
  inservice
 real RT-LINUX-153
  address 172.28.0.153
  inservice
 real RT-LINUX-154
  address 172.28.0.154
  inservice
 real RT-LINUX-21
  address 172.28.1.21
  inservice
 real RT-LINUX-25
  address 172.28.1.25
  inservice
!
 serverfarm COOKIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm COOKIE-1
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm COOKIE-HASH
  nat server
  nat client CLIENT_NAT
  real 10.96.130.18
   inservice
  real 10.96.130.19
   inservice
  real 10.96.130.26
   inservice
  real 10.96.130.32
   inservice
  real 10.96.130.18 90
   inservice
  real 10.96.130.19 90
   inservice
  real 10.96.130.26 90
   inservice
  real 10.96.130.32 90
   inservice
!
 serverfarm COOKIE-INSERT
  nat server
  no nat client
  real name RT-IIS-152
   inservice
  real name BRG-LINUX-11
   inservice
  real name LOCAL-LINUX-242
   inservice
  probe HTTP-PROBE
!
 serverfarm CS-COOKIES
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm CS-MOZILLA
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm CS-MSIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm CS-SERVERS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm CS-SERVERS-80
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-IIS-152
```

```
    inservice
   real name RT-LINUX-154
    inservice
!
  serverfarm DEFAULT
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-244
    inservice
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-151
    inservice
   probe HTTP-PROBE
   probe ICMP
!
  serverfarm FTP
   nat server
   nat client CLIENT_NAT
   real name RT-IIS-152 21
    inservice
   real name BRG-LINUX-11 21
    inservice
   real name LOCAL-IIS-245 21
    inservice
   probe FTP
!
  serverfarm GEN
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-151
    inservice
   probe ICMP
!
  serverfarm GEN-443
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-244
    inservice
   real name RT-IIS-152
    inservice
   probe SSLPROBE
!
  serverfarm GEN-80
   nat server
   nat client CLIENT_NAT
   real name RT-IIS-152
    inservice
   real name BRG-LINUX-12
    inservice
   real name LOCAL-IIS-241
    inservice
   probe TCP-GENERIC
!
  serverfarm GEN-UDP
   nat server
   nat client CLIENT_NAT
   real name RT-LINUX-154
    inservice
   real name BRG-LINUX-13
    inservice
   probe ICMP
```

```
!
 serverfarm HDR-IXIA
  nat server
  no nat client
  real name IXIA-25
   inservice
  real name IXIA-26
   inservice
  probe HTTP-PROBE
!
 serverfarm HEADER
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm HEADER-INSERT
  nat server
  no nat client
  real name RT-LINUX-153
   inservice
  real name BRG-LINUX-14
   inservice
  real name LOCAL-IIS-245
   inservice
  probe HTTP-PROBE
!
 serverfarm HEADER-INSERT2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-IIS-241
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   inservice
  probe HTTP-PROBE
!
 serverfarm ICMP
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  probe ICMP
!
 serverfarm IDLE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   no inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm IDLE-UDP
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm IXIA
  nat server
  nat client CLIENT_NAT
  real name IXIA-25
   inservice
  real name IXIA-26
   inservice
  probe ICMP
!
 serverfarm LENGTHS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-IIS-152
   inservice
!
 serverfarm MAX-CONN
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 3
   minconns 2
   inservice
  probe HTTP
!
 serverfarm MAX-CONN2
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 500
   inservice
  real name LOCAL-LINUX-240
   maxconns 500
   inservice
  real name LOCAL-IIS-241
   maxconns 500
   inservice
  real name RT-IIS-152
   maxconns 500
   inservice
  real name RT-LINUX-151 90
   maxconns 500
   inservice
  real name RT-LINUX-151 91
   maxconns 500
   inservice
  real name RT-LINUX-151 92
   maxconns 500
   inservice
  real name RT-LINUX-151 93
   maxconns 500
   inservice
  real name RT-LINUX-151 94
   maxconns 500
   inservice
  real name RT-LINUX-151 95
   maxconns 500
   inservice
  probe HTTP
!
 serverfarm PERSISTENT
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm PRED-CONNS
  nat server
  nat client CLIENT_NAT
  predictor leastconns
  real name BRG-LINUX-11
   inservice
  real name BRG-LINUX-12
   inservice
  real name BRG-LINUX-13
   inservice
  real name BRG-LINUX-14
   inservice
  real name BRG-LINUX-15
   inservice
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-IIS-241
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-243
   inservice
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-151
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   health probe PRED-PING
   inservice
  probe HTTP
!
 serverfarm PRED-CONNS-UDP
  nat server
  nat client CLIENT_NAT
  predictor leastconns
  failaction purge
  real name BRG-LINUX-11 2222
   inservice
  real name BRG-LINUX-12 2222
   inservice
  real name LOCAL-LINUX-240 2222
   inservice
  real name LOCAL-LINUX-242 2222
   inservice
  real name LOCAL-LINUX-244 2222
   inservice
  real name RT-LINUX-151 2222
   inservice
  real name RT-LINUX-153 2222
   inservice
  real name RT-LINUX-154 2222
   health probe PRED-PING
   inservice
  probe ICMP
!
 serverfarm PREDICTOR
  nat server
  nat client CLIENT_NAT
  predictor hash url
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm PROBES
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   inservice
  real name LOCAL-LINUX-244
   inservice
  real name RT-LINUX-151
   inservice
  probe HTTP-PROBE
  probe ICMP
  probe TELNET
  probe FTP
  probe TCP
!
 serverfarm RED-ALL-SVRS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm REDIRECT
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm REDIRECT-100K
  nat server
  no nat client
  redirect-vserver REDIRECT-100K
   webhost relocation 192.168.120.120/redirect-100k.html
   inservice
!
 serverfarm REDIRECT-10K
  nat server
  no nat client
  redirect-vserver REDIRECT-10K
   webhost relocation 192.168.120.221/redirect-10k.html
   inservice
!
 serverfarm REDIRECT-1K
  nat server
  no nat client
  redirect-vserver REDIRECT-1K
   webhost relocation 192.168.120.222/redirect-1k.html
   inservice
!
 serverfarm RHI
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
  probe ICMP-RTR
!
 serverfarm SF-ORDER1
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
  probe HTTP
!
 serverfarm SF-ORDER2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
  probe HTTP
!
 serverfarm SF-ORDER3
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-IIS-152
   inservice
  probe HTTP
!
 serverfarm SF-ORDER4
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-153
   inservice
  probe HTTP
!
 serverfarm SF-ORDER5
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-154
   inservice
  probe HTTP
!
 serverfarm SF-ORDER6
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
  probe HTTP
!
 serverfarm SORRY
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
!
 serverfarm SORRY-BACK
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
!
 serverfarm SPAN-TEST
  nat server
  nat client CLIENT_NAT
  real name IXIA-25
   inservice
  real name IXIA-26
   inservice
  probe ICMP
!
 serverfarm STICKY-COOKIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm STICKY-HEADER
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-154
   inservice
  real name LOCAL-IIS-245
   inservice
  real name LOCAL-LINUX-242
```

```
    inservice
   probe HTTP
 !
  serverfarm STICKY-HEADER2
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-240
    inservice
   real name RT-IIS-152
    inservice
   real name RT-LINUX-153
    inservice
   probe HTTP
 !
  serverfarm STICKY-MASK
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-245
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm STICKY-SSL
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-241
    inservice
   real name RT-LINUX-153
    inservice
 !
  serverfarm UDP
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-241
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm URL-MAP-128K
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm URL-MAP-16K
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-245
    inservice
   real name RT-LINUX-151
    inservice
 !
  serverfarm URL-MAP-32K
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-240
    inservice
   real name RT-IIS-152
    inservice
 !
  serverfarm URL-MAP-512K
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-241
    inservice
   real name RT-LINUX-153
    inservice
 !
  serverfarm URL-MAP-64K
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm URL-MAPS
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-244
    inservice
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-151
    inservice
 !
  serverfarm VIP-DEPEND1
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-240 80
    inservice
   real name LOCAL-IIS-241 80
    inservice
   real name RT-LINUX-151 80
    inservice
   probe HTTP
 !
  serverfarm VIP-DEPEND2
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-242 80
    inservice
   real name LOCAL-IIS-243 80
    inservice
   real name RT-LINUX-153 80
    inservice
   probe HTTP
 !
  serverfarm VIP-DEPEND3
   nat server
   nat client CLIENT_NAT
   real name BRG-LINUX-15 80
    inservice
   health retries 3 failed 100
 !
  serverfarm WEIGHT
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-240
    weight 10
    inservice
   real name LOCAL-IIS-241
    weight 20
    inservice
   real name RT-IIS-152
    weight 30
    inservice
   real name RT-LINUX-153
    weight 40
    inservice
 !
  sticky 10 netmask 255.255.255.255 address both timeout 10
 !
  sticky 11 netmask 255.255.255.255 address both timeout 30
 !
  sticky 12 netmask 255.255.255.255 address both timeout 30
 !
  sticky 13 netmask 255.255.255.255 address both timeout 30
 !
  sticky 14 netmask 255.255.255.255 address both timeout 30
 !
  sticky 15 netmask 255.255.255.255 address both timeout 30
 !
  sticky 20 netmask 255.255.255.255 address destination timeout 30
 !
  sticky 30 netmask 255.255.255.255 address source timeout 30
 !
  sticky 31 netmask 255.255.255.255 timeout 30
 !
  sticky 33 netmask 255.255.255.255 address source timeout 1
 !
  sticky 40 header MSISDN timeout 30
 !
  sticky 41 header TestHeader timeout 30
   header offset 15 length 7
 !
  sticky 42 cookie Safeharbor-Cookie1 insert timeout 30
 !
  sticky 43 cookie Sticky_Cookie_Group_43 insert
 !
  sticky 44 cookie Sticky_Cookie_Group_44 insert
 !
  sticky 45 cookie Sticky_Cookie_Group_45 insert timeout 30
 !
  sticky 46 cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes insert
 timeout 1
 !
  sticky 110 ssl timeout 30
 !
  sticky 113 ssl timeout 1
 !
  sticky 210 cookie CSM_TEST timeout 10
 !
  policy 16K-FORWARDING
   url-map 16K
   sticky-group 11
   serverfarm URL-MAP-16K
 !
  policy 32K-FORWARDING
   url-map 32K
   sticky-group 12
   serverfarm URL-MAP-32K
 !
  policy 64K-FORWARDING
   url-map 64K
   sticky-group 13
   serverfarm URL-MAP-64K
 !
  policy BROWSER_MSIE
   header-map BROWSER_MSIE
   serverfarm CS-MSIE
 !
  policy REDIRECT-1K
   url-map REDIRECT-1K
   serverfarm REDIRECT-1K
 !
  policy REDIRECT-10K
   url-map REDIRECT-10K
   serverfarm REDIRECT-10K
 !
  policy REDIRECT-100K
   url-map REDIRECT-100K
   serverfarm REDIRECT-100K
 !
  policy COOKIE-POLICY
   cookie-map COOKIE-MAP
   serverfarm CS-COOKIES
 !
  policy STICKY-COOKIE
   cookie-map COOKIE-MAP
   serverfarm STICKY-COOKIE
 !
  policy STICKY-HEADER
   sticky-group 40
   serverfarm STICKY-HEADER
 !
  policy STICKY-HEADER2
   sticky-group 41
   serverfarm STICKY-HEADER2
 !
  policy P-ORDER1
   header-map M-ORDER1
   serverfarm SF-ORDER1
 !
  policy P-ORDER2
   header-map M-ORDER2
   serverfarm SF-ORDER2
 !
  policy P-ORDER3
   header-map M-ORDER3
   serverfarm SF-ORDER3
 !
  policy P-ORDER4
   header-map M-ORDER4
   serverfarm SF-ORDER4
 !
  policy P-ORDER5
   header-map M-ORDER5
   serverfarm SF-ORDER5
 !
  policy P-ORDER6
   header-map M-ORDER6
   serverfarm SF-ORDER6
 !
  policy P-ORDER7
   header-map M-ORDER7
   serverfarm SF-ORDER6
 !
  policy P-MAX-CONN
   url-map M-MAX-CONN
   sticky-group 31
   serverfarm MAX-CONN
 !
  policy 128K-FORWARDING
```

```
  url-map 128K
  sticky-group 14
  serverfarm URL-MAP-128K
!
 policy 512K-FORWARDING
  url-map 512K
  sticky-group 15
  serverfarm URL-MAP-512K
!
 policy BROWSER_MOZILLA
  header-map BROWSER_MOZILLA
  serverfarm CS-MOZILLA
!
 policy P-GSLB-SH dns
  dns map M-GSLB-SH
!
 policy P-GSLB-NS dns
  dns map M-GSLB-NS
!
 policy RHI
  url-map 64K
  serverfarm RHI
!
 policy SASP2
  url-map INDEX.HTML
  nat client CLIENT_NAT
!
 policy P-INDEX
  url-map INDEX.HTML
  serverfarm GEN-80
!
 policy P-HDR-INSERT
  nat client CLIENT_NAT
  header-map M-HDR-INSERT
  sticky-group 42
  serverfarm HEADER-INSERT
!
 policy P-HDR-SRCDST-IP
  header-map M-HDR-SRCDST-IP
  sticky-group 43
  serverfarm HEADER-INSERT2
!
 policy P-HDR-IXIA
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 44
  serverfarm HDR-IXIA
!
 policy P-COOKIE-INS
  url-map INDEX.HTML
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 46
  serverfarm COOKIE-INSERT
!
 policy P-COOKIE-INS2
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 46
  serverfarm COOKIE-INSERT
!
 vserver COOKIE
  virtual 192.168.120.215 tcp www
  serverfarm COOKIE
  persistent rebalance
  inservice
!
 vserver COOKIE-HASH
  virtual 10.20.30.40 tcp www
  serverfarm COOKIE-HASH
  sticky 30 group 45
  persistent rebalance
  inservice
!
 vserver COOKIE-INSERT
  virtual 192.168.120.233 tcp www
  persistent rebalance
  parse-length 4000
  slb-policy P-COOKIE-INS
  slb-policy P-COOKIE-INS2
  inservice
!
 vserver FTP
  virtual 192.168.120.219 tcp 1111 service ftp
  serverfarm FTP
  no persistent rebalance
  inservice
!
 vserver FTP2
  virtual 192.168.120.219 tcp ftp service ftp
  serverfarm FTP
  no persistent rebalance
  inservice
!
 vserver GEN
  virtual 192.168.120.200 any
  serverfarm GEN
  idle 4
  persistent rebalance
  inservice
!
 vserver GEN-443
  virtual 192.168.120.200 tcp https
  serverfarm GEN-443
  sticky 1 group 113
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-80
  virtual 192.168.120.200 tcp www
  serverfarm GEN-80
  sticky 1 group 33
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-UDP
  virtual 192.168.120.200 udp 0
  serverfarm GEN-UDP
  idle 2
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver HDR-IXIA
```

```
  virtual 172.31.105.240 tcp www
  persistent rebalance
  slb-policy P-HDR-IXIA
  inservice
!
 vserver HEADER
  virtual 192.168.120.216 tcp www
  serverfarm HEADER
  persistent rebalance
  slb-policy BROWSER_MSIE
  slb-policy BROWSER_MOZILLA
  inservice
!
 vserver HEADER-INSERT
  virtual 192.168.120.231 tcp www
  serverfarm DEFAULT
  persistent rebalance
  parse-length 3000
  slb-policy P-HDR-INSERT
  inservice
!
 vserver HEADER-INSERT2
  virtual 192.168.120.232 tcp www
  persistent rebalance
  parse-length 3000
  slb-policy P-HDR-SRCDST-IP
  inservice
!
 vserver IDLE
  virtual 192.168.120.211 tcp 0
  serverfarm IDLE
  idle 60
  persistent rebalance
  inservice
!
 vserver IDLE-UDP
  virtual 192.168.120.211 udp 0
  serverfarm IDLE-UDP
  idle 60
  persistent rebalance
  inservice
!
 vserver IXIA
  virtual 172.31.111.150 any
  vlan 105
  serverfarm IXIA
  advertise active
  persistent rebalance
  inservice
!
 vserver L3
  virtual 192.168.120.240 any
  serverfarm ICMP
  persistent rebalance
  inservice
!
 vserver L4
  virtual 192.168.120.241 tcp 4444 service termination
  serverfarm ICMP
  persistent rebalance
  inservice
!
 vserver L7
  virtual 192.168.120.242 tcp www
  serverfarm ICMP
  persistent rebalance
  slb-policy 64K-FORWARDING
  inservice
!
 vserver LENGTHS
  virtual 192.168.120.218 tcp www
  serverfarm LENGTHS
  replicate csrp connection
  persistent rebalance
  slb-policy P-INDEX
  inservice
!
 vserver MAX-CONN
  virtual 192.168.120.205 tcp www
  serverfarm MAX-CONN2
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  slb-policy P-MAX-CONN
  inservice
!
 vserver PERSISTENT
  virtual 192.168.120.224 tcp www
  serverfarm PERSISTENT
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  slb-policy 16K-FORWARDING
  slb-policy 32K-FORWARDING
  slb-policy 64K-FORWARDING
  slb-policy 128K-FORWARDING
  slb-policy 512K-FORWARDING
  inservice
!
 vserver PRED-CONNS
  virtual 192.168.120.204 tcp www
  serverfarm PRED-CONNS
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver PRED-CONNS-UDP
  virtual 192.168.120.204 udp 0
  unidirectional
  serverfarm PRED-CONNS-UDP
  idle 300
  pending 60
  persistent rebalance
  inservice
!
 vserver PREDICTOR
  virtual 192.168.120.217 tcp www
  serverfarm PREDICTOR
  persistent rebalance
  inservice
!
 vserver PROBES
  virtual 192.168.120.229 tcp www
  serverfarm PROBES
  persistent rebalance
  inservice
!
 vserver REBALANCE
```

```
  virtual 192.168.120.235 tcp www
  serverfarm LENGTHS
  replicate csrp connection
  persistent rebalance
  parse-length default-policy
  slb-policy P-INDEX
  inservice
!
 vserver RED-100K-VIP
  virtual 192.168.120.220 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver RED-10K-VIP
  virtual 192.168.120.221 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver RED-1K-VIP
  virtual 192.168.120.222 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver REDIRECT
  virtual 192.168.120.213 tcp www
  serverfarm REDIRECT
  persistent rebalance
  slb-policy REDIRECT-1K
  slb-policy REDIRECT-10K
  slb-policy REDIRECT-100K
  inservice
!
 vserver RHI-21
  virtual 172.31.111.201 tcp 0 service ftp
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  inservice
!
 vserver RHI-80
  virtual 172.31.111.200 tcp www
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  slb-policy RHI
  inservice
!
 vserver RHI-TERM
  virtual 172.31.111.202 tcp 0 service termination
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  inservice
!
 vserver SORRY
  virtual 192.168.120.228 tcp www
  serverfarm SORRY backup SORRY-BACK
  persistent rebalance
  inservice
!
 vserver SPAN-TEST
  virtual 192.168.120.230 tcp www
  serverfarm SPAN-TEST
  persistent rebalance
  inservice
!
 vserver STICKY-COOKIE
  virtual 192.168.120.227 tcp www
  serverfarm COOKIE-1
  persistent rebalance
  slb-policy STICKY-COOKIE
  inservice
!
 vserver STICKY-HEADER
  virtual 192.168.120.201 tcp www
  serverfarm DEFAULT
  persistent rebalance
  slb-policy STICKY-HEADER
  inservice
!
 vserver STICKY-MASK
  virtual 192.168.120.225 tcp www
  serverfarm STICKY-MASK
  sticky 30 group 30
  persistent rebalance
  inservice
!
 vserver STICKY-SSL
  virtual 192.168.120.226 tcp https
  serverfarm STICKY-SSL
  persistent rebalance
  inservice
!
 vserver UDP
  virtual 192.168.120.219 udp dns
  serverfarm UDP
  idle 4
  persistent rebalance
  inservice
!
 vserver URL-MAPS
  virtual 192.168.120.214 tcp www
  serverfarm URL-MAPS
  persistent rebalance
  slb-policy 16K-FORWARDING
  slb-policy 32K-FORWARDING
  slb-policy 64K-FORWARDING
  slb-policy 128K-FORWARDING
  slb-policy 512K-FORWARDING
  inservice
!
 vserver VIP-DEPEND1
  virtual 192.168.120.202 tcp www
  status-tracking VIP-DEPEND2
  serverfarm VIP-DEPEND1
  persistent rebalance
  inservice
!
 vserver VIP-DEPEND2
  virtual 192.168.120.202 tcp 81
  status-tracking VIP-DEPEND3
  serverfarm VIP-DEPEND2
  persistent rebalance
  inservice
!
 vserver VIP-DEPEND3
  virtual 192.168.120.202 tcp 83
  serverfarm VIP-DEPEND3
  persistent rebalance
  inservice
!
 vserver VS-ORDER
  virtual 192.168.120.203 tcp www
  serverfarm DEFAULT
  persistent rebalance
  slb-policy P-ORDER1
  slb-policy P-ORDER2
  slb-policy P-ORDER3
  slb-policy P-ORDER4
  slb-policy P-ORDER5
  inservice
!
 vserver WEB
  virtual 192.168.120.210 tcp https
  serverfarm CS-SERVERS
  sticky 30 group 110
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver WEB-80
  virtual 192.168.120.210 tcp www
  serverfarm CS-SERVERS-80
  sticky 30 group 30
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver WEIGHT
  virtual 192.168.120.212 tcp www
  serverfarm WEIGHT
  persistent rebalance
  inservice
!
 dfp
  agent 172.31.101.41 5555 1000 37 45
  agent 172.31.101.41 7777 1001
!
 xml-config
  port 8002
  vlan 120
  credentials admin password system
  inservice
!
module ContentSwitchingModule 9
 policy STICKY-COOKIE
!
 vserver COOKIE
  persistent rebalance
  no inservice
!
 vserver STICKY-HEADER
  persistent rebalance
  no inservice
```

```
!
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
port-channel per-module load-balance
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
vlan 2,16,29,31,83
!
vlan 88
 remote-span
!
vlan 97-99,105-106,120-121,130,141,160,281,283,320,329,555,900,2830
!
!
!
!
interface Port-channel1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel5
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel6
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel99
```

```
 switchport
 switchport access vlan 99
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 99
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface GigabitEthernet4/1
 switchport
 switchport access vlan 83
 switchport mode access
 no ip address
!
interface GigabitEthernet4/2
 no ip address
 shutdown
!
interface GigabitEthernet4/3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet4/4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet4/5
 no ip address
 shutdown
!
interface GigabitEthernet4/6
 no ip address
 shutdown
!
interface GigabitEthernet4/7
 no ip address
 shutdown
!
interface GigabitEthernet4/8
 no ip address
 shutdown
!
interface GigabitEthernet4/9
 no ip address
 shutdown
!
interface GigabitEthernet4/10
 no ip address
 shutdown
!
interface GigabitEthernet4/11
 no ip address
!
interface GigabitEthernet4/12
 no ip address
```

```
!
interface GigabitEthernet4/13
 switchport
 switchport access vlan 160
 switchport mode access
 no ip address
!
interface GigabitEthernet4/14
 switchport
 switchport access vlan 160
 switchport mode access
 no ip address
!
interface GigabitEthernet4/15
 switchport
 switchport access vlan 120
 switchport mode access
 no ip address
!
interface GigabitEthernet4/16
 switchport
 switchport access vlan 120
 switchport mode access
 no ip address
!
interface GigabitEthernet4/17
 no ip address
 shutdown
!
interface GigabitEthernet4/18
 no ip address
 shutdown
!
interface GigabitEthernet4/19
 no ip address
 shutdown
!
interface GigabitEthernet4/20
 no ip address
 shutdown
!
interface GigabitEthernet4/21
 switchport
 switchport access vlan 320
 switchport mode access
 no ip address
 speed 100
!
interface GigabitEthernet4/22
 switchport
 switchport access vlan 329
 switchport mode access
 no ip address
 speed 100
!
interface GigabitEthernet4/23
 no ip address
 shutdown
!
interface GigabitEthernet4/24
 switchport
 switchport access vlan 130
 switchport mode access
 no ip address
```

```
!
interface GigabitEthernet4/25
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet4/26
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet4/27
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 7 mode desirable
!
interface GigabitEthernet4/28
 no ip address
 shutdown
!
interface GigabitEthernet4/29
 no ip address
 shutdown
!
interface GigabitEthernet4/30
 no ip address
 shutdown
!
interface GigabitEthernet4/31
 no ip address
 shutdown
!
interface GigabitEthernet4/32
 no ip address
 shutdown
!
interface GigabitEthernet4/33
 no ip address
 shutdown
!
interface GigabitEthernet4/34
 no ip address
 speed 100
!
interface GigabitEthernet4/35
 switchport
 switchport access vlan 555
 switchport mode access
 no ip address
!
interface GigabitEthernet4/36
 switchport
 switchport access vlan 16
 switchport mode access
 no ip address
```

```
!
interface GigabitEthernet4/37
 switchport
 switchport access vlan 99
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 99
 switchport mode trunk
 no ip address
 channel-group 99 mode desirable
!
interface GigabitEthernet4/38
 no ip address
 speed 100
!
interface GigabitEthernet4/39
 no ip address
 shutdown
 speed 100
!
interface GigabitEthernet4/40
 switchport
 no ip address
!
interface GigabitEthernet4/41
 no ip address
 shutdown
 speed 100
!
interface GigabitEthernet4/42
 no ip address
 shutdown
 speed 100
!
interface GigabitEthernet4/43
 no ip address
 shutdown
 speed 100
!
interface GigabitEthernet4/44
 no ip address
 shutdown
 speed 100
!
interface GigabitEthernet4/45
 no ip address
 shutdown
 speed 100
!
interface GigabitEthernet4/46
 no ip address
 shutdown
 speed 100
!
interface GigabitEthernet4/47
 no ip address
 speed 100
!
interface GigabitEthernet4/48
 switchport
 switchport access vlan 105
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 105,106
 switchport mode trunk
 no ip address
```

```
!
interface GigabitEthernet5/1
 no ip address
 shutdown
!
interface GigabitEthernet5/2
 no ip address
 shutdown
!
interface GigabitEthernet6/1
 no ip address
 shutdown
!
interface GigabitEthernet6/2
 no ip address
 shutdown
!
interface GigabitEthernet7/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet7/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet7/3
 no ip address
 shutdown
!
interface GigabitEthernet7/4
 no ip address
 shutdown
!
interface GigabitEthernet7/5
 no ip address
 shutdown
!
interface GigabitEthernet7/6
 no ip address
 shutdown
!
interface GigabitEthernet7/7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 7 mode desirable
!
interface GigabitEthernet7/8
 no ip address
 shutdown
!
interface GigabitEthernet7/9
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet7/10
 no ip address
 shutdown
!
interface GigabitEthernet7/11
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet7/12
 no ip address
 shutdown
!
interface GigabitEthernet7/13
 switchport
 switchport access vlan 99
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 99
 switchport mode trunk
 no ip address
 channel-group 99 mode desirable
!
interface GigabitEthernet7/14
 no ip address
 shutdown
!
interface GigabitEthernet7/15
 no ip address
 shutdown
!
interface GigabitEthernet7/16
 no ip address
 shutdown
!
interface TenGigabitEthernet9/1
 no ip address
 shutdown
!
interface TenGigabitEthernet9/2
 no ip address
 shutdown
!
interface TenGigabitEthernet9/3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 6 mode desirable
!
interface TenGigabitEthernet9/4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 6 mode desirable
!
interface TenGigabitEthernet9/5
 no ip address
 shutdown
!
interface TenGigabitEthernet9/6
 no ip address
 shutdown
!
interface TenGigabitEthernet9/7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 2 mode desirable
!
interface TenGigabitEthernet9/8
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 2 mode desirable
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan2
 ip address 10.2.0.251 255.255.255.0
 standby 2 ip 10.2.0.254
 standby 2 priority 200
 standby 2 preempt
!
interface Vlan16
 ip address 192.168.16.251 255.255.255.0
 no ip redirects
 no ipv6 mld snooping
 standby 16 ip 192.168.16.254
 standby 16 priority 200
 standby 16 preempt
!
interface Vlan28
 no ip address
 shutdown
!
interface Vlan83
 ip address 10.86.83.151 255.255.255.0
!
interface Vlan105
 ip address 192.168.105.251 255.255.255.0
 no ip proxy-arp
 standby 105 ip 192.168.105.254
 standby 105 priority 200
 standby 105 preempt
!
interface Vlan106
 ip address 192.168.106.251 255.255.255.0
 no ip proxy-arp
 standby 106 ip 192.168.106.254
 standby 106 priority 200
 standby 106 preempt
```

```
!
interface Vlan120
 ip address 192.168.120.251 255.255.255.0
 no ip redirects
 standby name ACE_120
 standby 120 ip 192.168.120.254
 standby 120 priority 200
 standby 120 preempt
 standby 120 track GigabitEthernet4/35 110
!
interface Vlan130
 ip address 192.168.130.251 255.255.255.0
 standby priority 200
 standby name ACE_130
 standby 130 ip 192.168.130.254
 standby 130 priority 200
 standby 130 preempt
!
interface Vlan141
 description "Reserved for future use"
 ip address 192.168.141.251 255.255.255.0
 shutdown
 standby priority 200
 standby name ACE_141
 standby 141 ip 192.168.141.254
 standby 141 priority 200
 standby 141 preempt
!
interface Vlan2830
 ip address 172.28.3.251 255.255.255.0
 no ip redirects
 standby name ACE_Bridged_V2830-V283
 standby 28 ip 172.28.3.254
 standby 28 priority 200
 standby 28 preempt
!
router ospf 105
 router-id 192.168.105.251
 log-adjacency-changes
 nsf
 redistribute static subnets route-map rhi-vip
 network 192.168.105.0 0.0.0.255 area 0
 network 192.168.106.0 0.0.0.255 area 0
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.86.83.1
ip route 10.1.0.0 255.255.128.0 10.2.0.253
ip route 10.3.0.0 255.255.255.0 10.2.0.253
ip route 172.28.0.0 255.255.255.0 192.168.120.1
ip route 172.28.4.0 255.255.255.0 172.28.3.1
ip route 172.29.0.0 255.255.255.0 192.168.120.1
ip route 172.29.0.200 255.255.255.255 Null0
!
no ip http server
!
access-list 104 permit ip 172.31.111.0 0.0.0.255 any
access-list 105 permit ip 192.168.125.0 0.0.0.255 any
access-list 106 permit ip 172.28.125.0 0.0.0.255 any
access-list 107 permit ip 192.168.140.0 0.0.0.255 any
access-list 109 permit ip 192.168.120.0 0.0.0.255 any
!
route-map rhi-vip permit 10
 match ip address 105 106 107 104
!
```

```
snmp-server community public RW
snmp-server trap-source Vlan83
snmp-server enable traps slb real virtual csrp
snmp-server host 10.86.83.124 public  casa slb
tftp-server disk0:1_ace-6k-1.cfg
tftp-server disk0:disk0:c6ace-t1k9-mz.3.0.0_A1_2.bin
tftp-server disk0:c6ace-t1k9-mz.3.0.0_A1_2.bin
tftp-server disk0:
tftp-server disk0:pkey.pem
tftp-server disk0:init.pem
tftp-server disk0:term.pem
tftp-server disk0:end-to-end.pem
tftp-server disk0:c6slb-apc.4-2-9.bin
tacacs-server host 172.29.0.236 key cisco
tacacs-server host 172.29.0.237 key cisco
tacacs-server host 172.29.0.235 key safeharbor
tacacs-server directed-request
tacacs-server key safeharbor
!
radius-server source-ports 1645-1646
!
control-plane
!
!
!
dial-peer cor custom
!
!
!
alias exec csm8 show mod csm 8
!
line con 0
 exec-timeout 0 0
 privilege level 15
 password bxb-safeharbor
 history size 256
 stopbits 1
line vty 0 4
 session-timeout 180
 exec-timeout 720 0
 privilege level 15
 password bxb-safeharbor
 length 0
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 5 9
 exec-timeout 720 0
 password bxb-safeharbor
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 10 15
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
!
!
monitor session 1 destination analysis-module 2 data-port 1
monitor session 2 source interface Te1/1
monitor session 2 destination analysis-module 2 data-port 2
scheduler runtime netinput 300
ntp clock-period 17179863
ntp server 10.86.214.4
ntp server 10.86.208.4
ntp server 10.86.210.4
no cns aaa enable
end
```

```
sh-ace2-6k-1#
sh-ace2-6k-1#term length 24
sh-ace2-6k-1#term width 80
sh-ace2-6k-1#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k-1(config)#logging console
sh-ace2-6k-1(config)#end
sh-ace2-6k-1#
```

Return to SH-ACE2-6K-1, page 212

SH-ACE2-6K-2

Go to Basic Topology: Test Device Configurations, page 212

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

```
sh-ace2-6k-2#show running-config
Load for five secs: 2%/1%; one minute: 1%; five minutes: 1%
Time source is NTP, 09:59:12.836 EDT Fri Oct 17 2008

Building configuration...

Current configuration : 41242 bytes
!
! Last configuration change at 09:58:51 EDT Fri Oct 17 2008 by cisco
!
upgrade fpd auto
version 12.2
service timestamps debug datetime msec localtime
service timestamps log datetime msec localtime
no service password-encryption
service internal
service counters max age 5
!
hostname sh-ace2-6k-2
!
boot system flash disk0:s72033-adventerprisek9_wan-mz.122-18.SXF13.bin
logging buffered 256000 debugging
!
username cisco secret 5 $1$T8kv$hKXYnm0ZbKsivQETK9YeH0
no aaa new-model
clock timezone EST -5
clock summer-time EDT recurring
clock calendar-valid
svclc multiple-vlan-interfaces
svclc module 1 vlan-group 11111
svclc vlan-group 11111  2,16,29,83,97-99,105,106,120,130,160,281,283,320,329
svclc vlan-group 11111  900,2830
analysis module 2 management-port access-vlan 83
ip subnet-zero
!
!
!
ip ssh time-out 60
ip ssh authentication-retries 2
ip domain-name cisco.com
ipv6 mfib hardware-switching replication-mode ingress
vtp mode transparent
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
no mls acl tcam share-global
mls cef error action freeze
!
!
!
!
!
!
!
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
!
spanning-tree mode rapid-pvst
spanning-tree extend system-id
spanning-tree vlan 1-79,90-900 priority 28672
module ContentSwitchingModule 8
 variable GSLB_LICENSE_KEY Q1NDT0dTTEJCRU9XVUxG
 variable REAL_SLOW_START_ENABLE 3
 variable SASP_FIRST_BIND_ID 1000
 variable SASP_GWM_BIND_ID_MAX 8
!
 ft group 2 vlan 900
  priority 100 alt 110
  preempt
  track gateway 192.168.16.252
  track interface GigabitEthernet4/38
  track mode any
!
 vlan 120 client
  ip address 192.168.120.9 255.255.255.0 alt 192.168.120.8 255.255.255.0
  gateway 192.168.120.254
  alias 192.168.120.7 255.255.255.0
!
 vlan 29 server
  ip address 172.29.0.13 255.255.255.0 alt 172.29.0.12 255.255.255.0
  route 172.28.0.0 255.255.0.0 gateway 172.29.0.253
  alias 172.29.0.11 255.255.255.0
!
 vlan 105 client
  ip address 192.168.105.13 255.255.255.0 alt 192.168.105.12 255.255.255.0
  route 192.168.16.0 255.255.255.0 gateway 192.168.105.251
!
 vlan 83 client
  ip address 10.86.83.14 255.255.255.0 alt 10.86.83.13 255.255.255.0
  route 161.44.0.0 255.255.0.0 gateway 10.86.83.1
  route 10.80.0.0 255.248.0.0 gateway 10.86.83.1
!
 natpool CLIENT_NAT 192.168.120.209 192.168.120.210 netmask 255.255.255.252
!
 script file disk0:c6slb-apc.tcl
!
 probe ICMP icmp
  interval 10
  retries 2
  failed 10
  receive 3
!
 probe HTTP-PROBE http
  interval 5
  failed 10
  open 3
  receive 5
  port 80
!
 probe TCP tcp
  interval 10
  retries 2
  failed 10
  open 3
  port 22
!
 probe TELNET telnet
  interval 10
  retries 2
  failed 10
  open 3
  receive 5
  port 23
!
 probe FTP ftp
  interval 10
  retries 2
  failed 10
  open 3
  receive 5
  port 21
!
 probe SSLPROBE script
  script SSL_PROBE_SCRIPT 0
  interval 5
  failed 10
!
 probe TCP-GENERIC tcp
  interval 5
  failed 10
!
 probe HTTP http
  interval 5
  failed 10
  open 3
  receive 5
!
 probe PRED-PING icmp
  address 172.29.0.1 routed
  interval 5
  retries 2
  failed 3
!
 probe FORCED-FAIL http
  request method get url /notthere.html
  expect status 200  299
  interval 10
  retries 2
  failed 5
  open 3
  receive 5
!
 probe ICMP-RTR icmp
  address 172.29.0.1 routed
  interval 5
  failed 5
!
 map REDIRECT-1K url
  match protocol http url *redirect-1k.html
```

```
!
 map REDIRECT-10K url
  match protocol http url *redirect-10k.html
!
 map REDIRECT-100K url
  match protocol http url *redirect-100k.html
!
 map 64K url
  match protocol http url *64k*
!
 map 16K url
  match protocol http url *16k*
!
 map 32K url
  match protocol http url *32k*
!
 map 128K url
  match protocol http url *128k*
!
 map 512K url
  match protocol http url *512k*
!
 map COOKIE-MAP cookie
  match protocol http cookie CSM_TEST cookie-value This*is*a*test0
!
 map BROWSER_MOZILLA header
  match protocol http header User-Agent header-value *Mozilla*
!
 map BROWSER_MSIE header
  match protocol http header User-Agent header-value *MSIE*
!
 map PARSE-LENGTH header
  match protocol http header Accept-Charset header-value *utf-8
!
 map M-ORDER1 header
  match protocol http header TestHeader header-value *p-order1*
!
 map M-ORDER2 header
  match protocol http header TestHeader header-value *p-order2*
!
 map M-ORDER3 header
  match protocol http header TestHeader header-value *p-order3*
!
 map M-ORDER4 header
  match protocol http header TestHeader header-value *p-order4*
!
 map M-ORDER5 header
  match protocol http header TestHeader header-value *p-order5*
!
 map M-ORDER6 header
  match protocol http header User-Agent header-value MSIE
!
 map M-ORDER7 header
  match protocol http header TestHeader header-value *p-order7*
!
 map M-MAX-CONN url
  match protocol http method GET url /index.html
!
 map M-GSLB-SH dns
  match protocol dns domain www.sh1.com
  match protocol dns domain www.sh2.com
  match protocol dns domain www.sh3.com
!
 map M-GSLB-NS dns
```

```
  match protocol dns domain .*
!
 map INDEX.HTML url
  match protocol http method GET url /index.html*
!
 map M-HDR-INSERT header
  insert protocol http header Source-IP header-value %is
  insert protocol http header Accept header-value anything
  insert protocol http header Pragma header-value "Pragma no Pragma that is the question"
  insert protocol http header Destination_IP header-value %id
!
 map M-HDR-SRCDST-IP header
  insert protocol http header Source-IP header-value %is
  insert protocol http header Destination_IP header-value %id
!
 real 6K-3
  address 192.168.105.253
  location "needed for arp entry, CSCej81417"
  inservice
 real BRG-LINUX-11
  address 192.168.120.11
  inservice
 real BRG-LINUX-12
  address 192.168.120.12
  inservice
 real BRG-LINUX-13
  address 192.168.120.13
  inservice
 real BRG-LINUX-14
  address 192.168.120.14
  inservice
 real BRG-LINUX-15
  address 192.168.120.15
  inservice
 real IXIA-25
  address 172.31.20.250
  inservice
 real IXIA-26
  address 172.31.101.250
  inservice
 real LOCAL-IIS-241
  address 172.29.0.241
  inservice
 real LOCAL-IIS-243
  address 172.29.0.243
  inservice
 real LOCAL-IIS-245
  address 172.29.0.245
  inservice
 real LOCAL-LINUX-240
  address 172.29.0.240
  inservice
 real LOCAL-LINUX-242
  address 172.29.0.242
  inservice
 real LOCAL-LINUX-244
  address 172.29.0.244
  inservice
 real RT-IIS-152
  address 172.28.0.152
  inservice
 real RT-LINUX-151
  address 172.28.0.151
  inservice
 real RT-LINUX-153
  address 172.28.0.153
  inservice
 real RT-LINUX-154
  address 172.28.0.154
  inservice
 real RT-LINUX-21
  address 172.28.1.21
  inservice
 real RT-LINUX-25
  address 172.28.1.25
  inservice
!
 serverfarm COOKIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm COOKIE-1
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm COOKIE-HASH
  nat server
  nat client CLIENT_NAT
  real 10.96.130.18
   inservice
  real 10.96.130.19
   inservice
  real 10.96.130.26
   inservice
  real 10.96.130.32
   inservice
  real 10.96.130.18 90
   inservice
  real 10.96.130.19 90
   inservice
  real 10.96.130.26 90
   inservice
  real 10.96.130.32 90
   inservice
!
 serverfarm COOKIE-INSERT
  nat server
  no nat client
  real name RT-IIS-152
   inservice
  real name BRG-LINUX-11
   inservice
  real name LOCAL-LINUX-242
   inservice
  probe HTTP-PROBE
!
 serverfarm CS-COOKIES
  nat server
```

```
   nat client CLIENT_NAT
   real name LOCAL-LINUX-244
    inservice
   real name RT-IIS-152
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm CS-MOZILLA
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-245
    inservice
   real name RT-LINUX-153
    inservice
 !
  serverfarm CS-MSIE
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-151
    inservice
 !
  serverfarm CS-SERVERS
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-242
    inservice
   real name LOCAL-IIS-241
    inservice
   real name RT-IIS-152
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm CS-SERVERS-80
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-242
    inservice
   real name LOCAL-IIS-241
    inservice
   real name RT-IIS-152
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm DEFAULT
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-244
    inservice
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-151
    inservice
   probe HTTP-PROBE
   probe ICMP
 !
  serverfarm FTP
   nat server
   nat client CLIENT_NAT
   real name RT-IIS-152 21
```

```
    inservice
   real name BRG-LINUX-11 21
    inservice
   real name LOCAL-IIS-245 21
    inservice
   probe FTP
 !
  serverfarm GEN
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-151
    inservice
   probe ICMP
 !
  serverfarm GEN-443
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-244
    inservice
   real name RT-IIS-152
    inservice
   probe SSLPROBE
 !
  serverfarm GEN-80
   nat server
   nat client CLIENT_NAT
   real name RT-IIS-152
    inservice
   real name BRG-LINUX-12
    inservice
   real name LOCAL-IIS-241
    inservice
   probe TCP-GENERIC
 !
  serverfarm GEN-UDP
   nat server
   nat client CLIENT_NAT
   real name RT-LINUX-154
    inservice
   real name BRG-LINUX-13
    inservice
   probe ICMP
 !
  serverfarm HDR-IXIA
   nat server
   no nat client
   real name IXIA-25
    inservice
   real name IXIA-26
    inservice
   probe HTTP-PROBE
 !
  serverfarm HEADER
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-242
    inservice
   real name LOCAL-IIS-241
    inservice
   real name RT-LINUX-153
    inservice
 !
  serverfarm HEADER-INSERT
   nat server
   no nat client
   real name RT-LINUX-153
    inservice
   real name BRG-LINUX-14
    inservice
   real name LOCAL-IIS-245
    inservice
   probe HTTP-PROBE
 !
  serverfarm HEADER-INSERT2
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-240
    inservice
   real name LOCAL-IIS-241
    inservice
   real name LOCAL-LINUX-242
    inservice
   real name LOCAL-IIS-243
    inservice
   real name RT-LINUX-151
    inservice
   real name RT-IIS-152
    inservice
   real name RT-LINUX-153
    inservice
   real name RT-LINUX-154
    inservice
   probe HTTP-PROBE
 !
  serverfarm ICMP
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-244
    inservice
   probe ICMP
 !
  serverfarm IDLE
   nat server
   nat client CLIENT_NAT
   real name LOCAL-IIS-241
    inservice
   real name RT-LINUX-153
    inservice
 !
  serverfarm IDLE-UDP
   nat server
   nat client CLIENT_NAT
   real name LOCAL-LINUX-240
    inservice
   real name RT-LINUX-154
    inservice
 !
  serverfarm IXIA
   nat server
   nat client CLIENT_NAT
   real name IXIA-25
    inservice
   real name IXIA-26
    inservice
   probe ICMP
 !
```

```
 serverfarm LENGTHS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-IIS-152
   inservice
!
 serverfarm MAX-CONN
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 3
   minconns 2
   inservice
  probe HTTP
!
 serverfarm MAX-CONN2
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-151
   maxconns 500
   inservice
  real name LOCAL-LINUX-240
   maxconns 500
   inservice
  real name LOCAL-IIS-241
   maxconns 500
   inservice
  real name RT-IIS-152
   maxconns 500
   inservice
  real name RT-LINUX-151 90
   maxconns 500
   inservice
  real name RT-LINUX-151 91
   maxconns 500
   inservice
  real name RT-LINUX-151 92
   maxconns 500
   inservice
  real name RT-LINUX-151 93
   maxconns 500
   inservice
  real name RT-LINUX-151 94
   maxconns 500
   inservice
  real name RT-LINUX-151 95
   maxconns 500
   inservice
  probe HTTP
!
 serverfarm PERSISTENT
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
!
```

```
 serverfarm PRED-CONNS
  nat server
  nat client CLIENT_NAT
  predictor leastconns
  real name BRG-LINUX-11
   inservice
  real name BRG-LINUX-12
   inservice
  real name BRG-LINUX-13
   inservice
  real name BRG-LINUX-14
   inservice
  real name BRG-LINUX-15
   inservice
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-IIS-241
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-243
   inservice
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-151
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   health probe PRED-PING
   inservice
  probe HTTP
!
 serverfarm PRED-CONNS-UDP
  nat server
  nat client CLIENT_NAT
  predictor leastconns
  failaction purge
  real name BRG-LINUX-11 2222
   inservice
  real name BRG-LINUX-12 2222
   inservice
  real name LOCAL-LINUX-240 2222
   inservice
  real name LOCAL-LINUX-242 2222
   inservice
  real name LOCAL-LINUX-244 2222
   inservice
  real name RT-LINUX-151 2222
   inservice
  real name RT-LINUX-153 2222
   inservice
  real name RT-LINUX-154 2222
   health probe PRED-PING
   inservice
  probe ICMP
!
 serverfarm PREDICTOR
  nat server
  nat client CLIENT_NAT
  predictor hash url
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm PROBES
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-LINUX-153
   inservice
  real name RT-LINUX-154
   inservice
  real name LOCAL-LINUX-244
   inservice
  real name RT-LINUX-151
   inservice
  probe ICMP
  probe HTTP-PROBE
  probe TELNET
  probe FTP
  probe TCP
!
 serverfarm RED-ALL-SVRS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm REDIRECT
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name LOCAL-LINUX-242
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm REDIRECT-100K
  nat server
  no nat client
  redirect-vserver REDIRECT-100K
   webhost relocation 192.168.120.120/redirect-100k.html
   inservice
!
 serverfarm REDIRECT-10K
  nat server
  no nat client
  redirect-vserver REDIRECT-10K
   webhost relocation 192.168.120.221/redirect-10k.html
   inservice
!
 serverfarm REDIRECT-1K
  nat server
  no nat client
  redirect-vserver REDIRECT-1K
   webhost relocation 192.168.120.222/redirect-1k.html
   inservice
!
 serverfarm RHI
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
  probe ICMP-RTR
!
 serverfarm SF-ORDER1
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
  probe HTTP
!
 serverfarm SF-ORDER2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-151
   inservice
  probe HTTP
!
 serverfarm SF-ORDER3
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242
   inservice
  real name RT-IIS-152
   inservice
  probe HTTP
!
 serverfarm SF-ORDER4
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-153
   inservice
  probe HTTP
!
 serverfarm SF-ORDER5
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-154
   inservice
  probe HTTP
!
 serverfarm SF-ORDER6
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
```

```
   inservice
  real name RT-LINUX-153
   inservice
  probe HTTP
!
 serverfarm SORRY
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
!
 serverfarm SORRY-BACK
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
!
 serverfarm SPAN-TEST
  nat server
  nat client CLIENT_NAT
  real name IXIA-25
   inservice
  real name IXIA-26
   inservice
  probe ICMP
!
 serverfarm STICKY-COOKIE
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm STICKY-HEADER
  nat server
  nat client CLIENT_NAT
  real name RT-LINUX-154
   inservice
  real name LOCAL-IIS-245
   inservice
  real name LOCAL-LINUX-242
   inservice
  probe HTTP
!
 serverfarm STICKY-HEADER2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
  real name RT-LINUX-153
   inservice
  probe HTTP
!
 serverfarm STICKY-MASK
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-154
```

```
   inservice
!
 serverfarm STICKY-SSL
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm UDP
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm URL-MAP-128K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm URL-MAP-16K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-245
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm URL-MAP-32K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   inservice
  real name RT-IIS-152
   inservice
!
 serverfarm URL-MAP-512K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-241
   inservice
  real name RT-LINUX-153
   inservice
!
 serverfarm URL-MAP-64K
  nat server
  nat client CLIENT_NAT
  real name LOCAL-IIS-243
   inservice
  real name RT-LINUX-154
   inservice
!
 serverfarm URL-MAPS
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-244
   inservice
  real name LOCAL-IIS-243
```

```
   inservice
  real name RT-LINUX-151
   inservice
!
 serverfarm VIP-DEPEND1
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240 80
   inservice
  real name LOCAL-IIS-241 80
   inservice
  real name RT-LINUX-151 80
   inservice
  probe HTTP
!
 serverfarm VIP-DEPEND2
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-242 80
   inservice
  real name LOCAL-IIS-243 80
   inservice
  real name RT-LINUX-153 80
   inservice
  probe HTTP
!
 serverfarm VIP-DEPEND3
  nat server
  nat client CLIENT_NAT
  real name BRG-LINUX-15 80
   inservice
  health retries 3 failed 100
!
 serverfarm WEIGHT
  nat server
  nat client CLIENT_NAT
  real name LOCAL-LINUX-240
   weight 10
   inservice
  real name LOCAL-IIS-241
   weight 20
   inservice
  real name RT-IIS-152
   weight 30
   inservice
  real name RT-LINUX-153
   weight 40
   inservice
!
 sticky 10 netmask 255.255.255.255 address both timeout 10
!
 sticky 11 netmask 255.255.255.255 address both timeout 30
!
 sticky 12 netmask 255.255.255.255 address both timeout 30
!
 sticky 13 netmask 255.255.255.255 address both timeout 30
!
 sticky 14 netmask 255.255.255.255 address both timeout 30
!
 sticky 15 netmask 255.255.255.255 address both timeout 30
!
 sticky 20 netmask 255.255.255.255 address destination timeout 30
!
 sticky 30 netmask 255.255.255.255 address source timeout 30
```

```
!
 sticky 31 netmask 255.255.255.255 timeout 30
!
 sticky 33 netmask 255.255.255.255 address source timeout 1
!
 sticky 40 header MSISDN timeout 30
!
 sticky 41 header TestHeader timeout 30
  header offset 15 length 7
!
 sticky 42 cookie Safeharbor-Cookie1 insert timeout 30
!
 sticky 43 cookie Sticky_Cookie_Group_43 insert
!
 sticky 44 cookie Sticky_Cookie_Group_44 insert
!
 sticky 45 cookie Sticky_Cookie_Group_45 insert timeout 30
!
 sticky 46 cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes insert
timeout 1
!
 sticky 110 ssl timeout 30
!
 sticky 113 ssl timeout 1
!
 sticky 210 cookie CSM_TEST timeout 10
!
 policy 16K-FORWARDING
  url-map 16K
  sticky-group 11
  serverfarm URL-MAP-16K
!
 policy 32K-FORWARDING
  url-map 32K
  sticky-group 12
  serverfarm URL-MAP-32K
!
 policy 64K-FORWARDING
  url-map 64K
  sticky-group 13
  serverfarm URL-MAP-64K
!
 policy BROWSER_MSIE
  header-map BROWSER_MSIE
  serverfarm CS-MSIE
!
 policy REDIRECT-1K
  url-map REDIRECT-1K
  serverfarm REDIRECT-1K
!
 policy REDIRECT-10K
  url-map REDIRECT-10K
  serverfarm REDIRECT-10K
!
 policy REDIRECT-100K
  url-map REDIRECT-100K
  serverfarm REDIRECT-100K
!
 policy COOKIE-POLICY
  cookie-map COOKIE-MAP
  serverfarm CS-COOKIES
!
 policy STICKY-COOKIE
  cookie-map COOKIE-MAP
  serverfarm STICKY-COOKIE
!
 policy STICKY-HEADER
  sticky-group 40
  serverfarm STICKY-HEADER
!
 policy STICKY-HEADER2
  sticky-group 41
  serverfarm STICKY-HEADER2
!
 policy P-ORDER1
  header-map M-ORDER1
  serverfarm SF-ORDER1
!
 policy P-ORDER2
  header-map M-ORDER2
  serverfarm SF-ORDER2
!
 policy P-ORDER3
  header-map M-ORDER3
  serverfarm SF-ORDER3
!
 policy P-ORDER4
  header-map M-ORDER4
  serverfarm SF-ORDER4
!
 policy P-ORDER5
  header-map M-ORDER5
  serverfarm SF-ORDER5
!
 policy P-ORDER6
  header-map M-ORDER6
  serverfarm SF-ORDER6
!
 policy P-ORDER7
  header-map M-ORDER7
  serverfarm SF-ORDER6
!
 policy P-MAX-CONN
  url-map M-MAX-CONN
  sticky-group 31
  serverfarm MAX-CONN
!
 policy 128K-FORWARDING
  url-map 128K
  sticky-group 14
  serverfarm URL-MAP-128K
!
 policy 512K-FORWARDING
  url-map 512K
  sticky-group 15
  serverfarm URL-MAP-512K
!
 policy BROWSER_MOZILLA
  header-map BROWSER_MOZILLA
  serverfarm CS-MOZILLA
!
 policy P-GSLB-SH dns
  dns map M-GSLB-SH
!
 policy P-GSLB-NS dns
  dns map M-GSLB-NS
!
 policy RHI
```

```
  url-map 64K
  serverfarm RHI
!
 policy SASP2
  url-map INDEX.HTML
  nat client CLIENT_NAT
!
 policy P-INDEX
  url-map INDEX.HTML
  serverfarm GEN-80
!
 policy P-HDR-INSERT
  nat client CLIENT_NAT
  header-map M-HDR-INSERT
  sticky-group 42
  serverfarm HEADER-INSERT
!
 policy P-HDR-SRCDST-IP
  header-map M-HDR-SRCDST-IP
  sticky-group 43
  serverfarm HEADER-INSERT2
!
 policy P-HDR-IXIA
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 44
  serverfarm HDR-IXIA
!
 policy P-COOKIE-INS
  url-map INDEX.HTML
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 46
  serverfarm COOKIE-INSERT
!
 policy P-COOKIE-INS2
  nat client CLIENT_NAT
  header-map M-HDR-SRCDST-IP
  sticky-group 46
  serverfarm COOKIE-INSERT
!
 vserver COOKIE
  virtual 192.168.120.215 tcp www
  serverfarm COOKIE
  persistent rebalance
  inservice
!
 vserver COOKIE-HASH
  virtual 10.20.30.40 tcp www
  serverfarm COOKIE-HASH
  sticky 30 group 45
  persistent rebalance
  inservice
!
 vserver COOKIE-INSERT
  virtual 192.168.120.233 tcp www
  persistent rebalance
  parse-length 4000
  slb-policy P-COOKIE-INS
  slb-policy P-COOKIE-INS2
  inservice
!
 vserver FTP
  virtual 192.168.120.219 tcp 1111 service ftp
  serverfarm FTP
  no persistent rebalance
  inservice
!
 vserver FTP2
  virtual 192.168.120.219 tcp ftp service ftp
  serverfarm FTP
  no persistent rebalance
  inservice
!
 vserver GEN
  virtual 192.168.120.200 any
  serverfarm GEN
  idle 4
  persistent rebalance
  inservice
!
 vserver GEN-443
  virtual 192.168.120.200 tcp https
  serverfarm GEN-443
  sticky 1 group 113
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-80
  virtual 192.168.120.200 tcp www
  serverfarm GEN-80
  sticky 1 group 33
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver GEN-UDP
  virtual 192.168.120.200 udp 0
  serverfarm GEN-UDP
  idle 2
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver HDR-IXIA
  virtual 172.31.105.240 tcp www
  persistent rebalance
  slb-policy P-HDR-IXIA
  inservice
!
 vserver HEADER
  virtual 192.168.120.216 tcp www
  serverfarm HEADER
  persistent rebalance
  slb-policy BROWSER_MSIE
  slb-policy BROWSER_MOZILLA
  inservice
!
 vserver HEADER-INSERT
  virtual 192.168.120.231 tcp www
  serverfarm DEFAULT
  persistent rebalance
  parse-length 3000
  slb-policy P-HDR-INSERT
  inservice
```

```
!
 vserver HEADER-INSERT2
  virtual 192.168.120.232 tcp www
  persistent rebalance
  parse-length 3000
  slb-policy P-HDR-SRCDST-IP
  inservice
!
 vserver IDLE
  virtual 192.168.120.211 tcp 0
  serverfarm IDLE
  idle 60
  persistent rebalance
  inservice
!
 vserver IDLE-UDP
  virtual 192.168.120.211 udp 0
  serverfarm IDLE-UDP
  idle 60
  persistent rebalance
  inservice
!
 vserver IXIA
  virtual 172.31.111.150 any
  vlan 105
  serverfarm IXIA
  advertise active
  persistent rebalance
  inservice
!
 vserver L3
  virtual 192.168.120.240 any
  serverfarm ICMP
  persistent rebalance
  inservice
!
 vserver L4
  virtual 192.168.120.241 tcp 4444 service termination
  serverfarm ICMP
  persistent rebalance
  inservice
!
 vserver L7
  virtual 192.168.120.242 tcp www
  serverfarm ICMP
  persistent rebalance
  slb-policy 64K-FORWARDING
  inservice
!
 vserver LENGTHS
  virtual 192.168.120.218 tcp www
  serverfarm LENGTHS
  replicate csrp connection
  persistent rebalance
  slb-policy P-INDEX
  inservice
!
 vserver MAX-CONN
  virtual 192.168.120.205 tcp www
  serverfarm MAX-CONN2
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  slb-policy P-MAX-CONN
  inservice
!
 vserver PERSISTENT
  virtual 192.168.120.224 tcp www
  serverfarm PERSISTENT
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  slb-policy 16K-FORWARDING
  slb-policy 32K-FORWARDING
  slb-policy 64K-FORWARDING
  slb-policy 128K-FORWARDING
  slb-policy 512K-FORWARDING
  inservice
!
 vserver PRED-CONNS
  virtual 192.168.120.204 tcp www
  serverfarm PRED-CONNS
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver PRED-CONNS-UDP
  virtual 192.168.120.204 udp 0
  unidirectional
  serverfarm PRED-CONNS-UDP
  idle 300
  pending 60
  persistent rebalance
  inservice
!
 vserver PREDICTOR
  virtual 192.168.120.217 tcp www
  serverfarm PREDICTOR
  persistent rebalance
  inservice
!
 vserver PROBES
  virtual 192.168.120.229 tcp www
  serverfarm PROBES
  persistent rebalance
  inservice
!
 vserver REBALANCE
  virtual 192.168.120.235 tcp www
  serverfarm LENGTHS
  replicate csrp connection
  persistent rebalance
  parse-length default-policy
  slb-policy P-INDEX
  inservice
!
 vserver RED-100K-VIP
  virtual 192.168.120.220 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver RED-10K-VIP
  virtual 192.168.120.221 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
```

```
 vserver RED-1K-VIP
  virtual 192.168.120.222 tcp www
  serverfarm RED-ALL-SVRS
  persistent rebalance
  inservice
!
 vserver REDIRECT
  virtual 192.168.120.213 tcp www
  serverfarm REDIRECT
  persistent rebalance
  slb-policy REDIRECT-1K
  slb-policy REDIRECT-10K
  slb-policy REDIRECT-100K
  inservice
!
 vserver RHI-21
  virtual 172.31.111.201 tcp 0 service ftp
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  inservice
!
 vserver RHI-80
  virtual 172.31.111.200 tcp www
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  slb-policy RHI
  inservice
!
 vserver RHI-TERM
  virtual 172.31.111.202 tcp 0 service termination
  vlan 105
  serverfarm RHI
  advertise active
  persistent rebalance
  inservice
!
 vserver SORRY
  virtual 192.168.120.228 tcp www
  serverfarm SORRY backup SORRY-BACK
  persistent rebalance
  inservice
!
 vserver SPAN-TEST
  virtual 192.168.120.230 tcp www
  serverfarm SPAN-TEST
  persistent rebalance
  inservice
!
 vserver STICKY-COOKIE
  virtual 192.168.120.227 tcp www
  serverfarm COOKIE-1
  persistent rebalance
  slb-policy STICKY-COOKIE
  inservice
!
 vserver STICKY-HEADER
  virtual 192.168.120.201 tcp www
  serverfarm DEFAULT
  persistent rebalance
  slb-policy STICKY-HEADER
  inservice
!
 vserver STICKY-MASK
  virtual 192.168.120.225 tcp www
  serverfarm STICKY-MASK
  sticky 30 group 30
  persistent rebalance
  inservice
!
 vserver STICKY-SSL
  virtual 192.168.120.226 tcp https
  serverfarm STICKY-SSL
  persistent rebalance
  inservice
!
 vserver UDP
  virtual 192.168.120.219 udp dns
  serverfarm UDP
  idle 4
  persistent rebalance
  inservice
!
 vserver URL-MAPS
  virtual 192.168.120.214 tcp www
  serverfarm URL-MAPS
  persistent rebalance
  slb-policy 16K-FORWARDING
  slb-policy 32K-FORWARDING
  slb-policy 64K-FORWARDING
  slb-policy 128K-FORWARDING
  slb-policy 512K-FORWARDING
  inservice
!
 vserver VIP-DEPEND1
  virtual 192.168.120.202 tcp www
  status-tracking VIP-DEPEND2
  serverfarm VIP-DEPEND1
  persistent rebalance
  inservice
!
 vserver VIP-DEPEND2
  virtual 192.168.120.202 tcp 81
  status-tracking VIP-DEPEND3
  serverfarm VIP-DEPEND2
  persistent rebalance
  inservice
!
 vserver VIP-DEPEND3
  virtual 192.168.120.202 tcp 83
  serverfarm VIP-DEPEND3
  persistent rebalance
  inservice
!
 vserver VS-ORDER
  virtual 192.168.120.203 tcp www
  serverfarm DEFAULT
  persistent rebalance
  slb-policy P-ORDER1
  slb-policy P-ORDER2
  slb-policy P-ORDER3
  slb-policy P-ORDER4
  slb-policy P-ORDER5
  inservice
!
```

```
 vserver WEB
  virtual 192.168.120.210 tcp https
  serverfarm CS-SERVERS
  sticky 30 group 110
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver WEB-80
  virtual 192.168.120.210 tcp www
  serverfarm CS-SERVERS-80
  sticky 30 group 30
  replicate csrp sticky
  replicate csrp connection
  persistent rebalance
  inservice
!
 vserver WEIGHT
  virtual 192.168.120.212 tcp www
  serverfarm WEIGHT
  persistent rebalance
  inservice
!
 dfp
  agent 172.31.101.41 5555 1000 37 45
  agent 172.31.101.41 7777 1001
!
 xml-config
  port 8002
  vlan 120
  credentials admin password system
  inservice
!
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
port-channel per-module load-balance
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
vlan 2,16,29,83,97-99,105-106,120,130,141,160,281,283,320,329,900,2830
!
!
!
!
interface Port-channel3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
```

```
interface Port-channel5
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel6
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
 spanning-tree cost 10000
!
interface Port-channel8
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel98
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface GigabitEthernet4/1
 switchport
 switchport access vlan 83
 switchport mode access
 no ip address
!
interface GigabitEthernet4/2
 no ip address
 shutdown
!
interface GigabitEthernet4/3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet4/4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet4/5
```

```
 no ip address
 shutdown
!
interface GigabitEthernet4/6
 no ip address
 shutdown
!
interface GigabitEthernet4/7
 no ip address
 shutdown
!
interface GigabitEthernet4/8
 no ip address
 shutdown
!
interface GigabitEthernet4/9
 no ip address
 shutdown
!
interface GigabitEthernet4/10
 no ip address
 shutdown
!
interface GigabitEthernet4/11
 no ip address
 shutdown
!
interface GigabitEthernet4/12
 no ip address
 shutdown
!
interface GigabitEthernet4/13
 switchport
 switchport access vlan 160
 switchport mode access
 no ip address
!
interface GigabitEthernet4/14
 switchport
 switchport access vlan 160
 switchport mode access
 no ip address
!
interface GigabitEthernet4/15
 no ip address
 shutdown
!
interface GigabitEthernet4/16
 no ip address
 shutdown
!
interface GigabitEthernet4/17
 no ip address
 shutdown
!
interface GigabitEthernet4/18
 no ip address
 shutdown
!
interface GigabitEthernet4/19
 no ip address
 shutdown
!
interface GigabitEthernet4/20
```

```
 no ip address
 shutdown
!
interface GigabitEthernet4/21
 switchport
 switchport access vlan 320
 switchport mode access
 no ip address
 speed 100
!
interface GigabitEthernet4/22
 switchport
 switchport access vlan 329
 switchport mode access
 no ip address
 speed 100
!
interface GigabitEthernet4/23
 no ip address
 shutdown
!
interface GigabitEthernet4/24
 no ip address
 shutdown
!
interface GigabitEthernet4/25
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet4/26
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet4/27
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 8 mode desirable
!
interface GigabitEthernet4/28
 no ip address
 shutdown
!
interface GigabitEthernet4/29
 no ip address
 shutdown
!
interface GigabitEthernet4/30
 no ip address
 shutdown
!
interface GigabitEthernet4/31
 no ip address
 shutdown
```

```
!
interface GigabitEthernet4/32
 no ip address
 shutdown
!
interface GigabitEthernet4/33
 no ip address
 shutdown
!
interface GigabitEthernet4/34
 no ip address
 speed 100
!
interface GigabitEthernet4/35
 switchport
 switchport access vlan 555
 switchport mode access
 no ip address
!
interface GigabitEthernet4/36
 switchport
 switchport access vlan 16
 switchport mode access
 no ip address
!
interface GigabitEthernet4/37
 no ip address
 shutdown
!
interface GigabitEthernet4/38
 switchport
 switchport access vlan 99
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 channel-group 98 mode desirable
!
interface GigabitEthernet4/39
 no ip address
 shutdown
!
interface GigabitEthernet4/40
 no ip address
 shutdown
!
interface GigabitEthernet4/41
 no ip address
 shutdown
!
interface GigabitEthernet4/42
 no ip address
 shutdown
!
interface GigabitEthernet4/43
 no ip address
 shutdown
!
interface GigabitEthernet4/44
 no ip address
 shutdown
!
interface GigabitEthernet4/45
 no ip address
 shutdown
!
interface GigabitEthernet4/46
 no ip address
 shutdown
!
interface GigabitEthernet4/47
 no ip address
 shutdown
!
interface GigabitEthernet4/48
 switchport
 switchport access vlan 105
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 105,106
 switchport mode trunk
 no ip address
!
interface GigabitEthernet5/1
 no ip address
 shutdown
!
interface GigabitEthernet5/2
 no ip address
 shutdown
!
interface GigabitEthernet7/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet7/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet7/3
 no ip address
 shutdown
!
interface GigabitEthernet7/4
 no ip address
 shutdown
!
interface GigabitEthernet7/5
 no ip address
 shutdown
!
interface GigabitEthernet7/6
 no ip address
 shutdown
!
interface GigabitEthernet7/7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 8 mode desirable
!
interface GigabitEthernet7/8
 no ip address
 shutdown
!
interface GigabitEthernet7/9
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet7/10
 no ip address
 shutdown
!
interface GigabitEthernet7/11
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
 channel-group 5 mode desirable
!
interface GigabitEthernet7/12
 no ip address
 shutdown
!
interface GigabitEthernet7/13
 no ip address
 shutdown
!
interface GigabitEthernet7/14
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 channel-group 98 mode desirable
!
interface GigabitEthernet7/15
 no ip address
 shutdown
!
interface GigabitEthernet7/16
 no ip address
 shutdown
!
interface TenGigabitEthernet9/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 900
 switchport mode trunk
 no ip address
!
interface TenGigabitEthernet9/2
 no ip address
 shutdown
!
interface TenGigabitEthernet9/3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 6 mode desirable
!
interface TenGigabitEthernet9/4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 6 mode desirable
!
interface TenGigabitEthernet9/5
 no ip address
 shutdown
!
interface TenGigabitEthernet9/6
 no ip address
 shutdown
!
interface TenGigabitEthernet9/7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 4 mode desirable
!
interface TenGigabitEthernet9/8
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 4 mode desirable
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan2
 ip address 10.2.0.252 255.255.255.0
 standby 2 ip 10.2.0.254
 standby 2 preempt
!
interface Vlan16
 ip address 192.168.16.252 255.255.255.0
 no ip redirects
 no ipv6 mld snooping
 standby 16 ip 192.168.16.254
 standby 16 preempt
!
interface Vlan83
 ip address 10.86.83.153 255.255.255.0
!
interface Vlan105
 ip address 192.168.105.252 255.255.255.0
 no ip proxy-arp
 standby 105 ip 192.168.105.254
 standby 105 preempt
!
interface Vlan106
 ip address 192.168.106.252 255.255.255.0
 no ip proxy-arp
 standby 106 ip 192.168.106.254
 standby 106 preempt
!
interface Vlan120
 ip address 192.168.120.252 255.255.255.0
 standby 120 ip 192.168.120.254
 standby 120 preempt
 standby 120 track GigabitEthernet4/35 20
!
interface Vlan130
 ip address 192.168.130.252 255.255.255.0
 standby name ACE_130
 standby 130 ip 192.168.130.254
 standby 130 preempt
!
interface Vlan141
 description "Reserved for future use"
 ip address 192.168.141.252 255.255.255.0
 shutdown
 standby priority 200
 standby name ACE_141
 standby 141 ip 192.168.141.254
 standby 141 priority 200
 standby 141 preempt
!
interface Vlan2830
 ip address 172.28.3.252 255.255.255.0
 no ip redirects
 standby name ACE_Bridged_V2830-V283
 standby 28 ip 172.28.3.254
 standby 28 preempt
!
router ospf 105
 router-id 192.168.105.252
 log-adjacency-changes
 nsf
 redistribute static subnets route-map rhi-vip
 network 192.168.105.0 0.0.0.255 area 0
 network 192.168.106.0 0.0.0.255 area 0
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.86.83.1
ip route 10.1.0.0 255.255.255.0 10.2.0.253
ip route 10.3.0.0 255.255.255.0 10.2.0.253
ip route 172.28.0.0 255.255.255.0 192.168.120.1
ip route 172.28.4.0 255.255.255.0 172.28.3.1
ip route 172.29.0.0 255.255.255.0 192.168.120.1
ip route 172.29.0.200 255.255.255.255 Null0
!
no ip http server
!
access-list 104 permit ip 172.31.111.0 0.0.0.255 any
access-list 105 permit ip 192.168.125.0 0.0.0.255 any
access-list 106 permit ip 172.28.125.0 0.0.0.255 any
access-list 107 permit ip 192.168.140.0 0.0.0.255 any
!
route-map rhi-vip permit 10
 match ip address 105 106 107 104
!
snmp-server community public RW
tftp-server disk0:c6slb-apc.4-2-9.bin
!
!
```

```
control-plane
!
!
!
dial-peer cor custom
!
!
!
alias exec csm8 show mod csm 8
!
line con 0
 exec-timeout 0 0
 password bxb-safeharbor
 login local
 history size 256
 stopbits 1
line vty 0 4
 exec-timeout 720 0
 password bxb-safeharbor
 login local
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 5 9
 exec-timeout 720 0
 password bxb-safeharbor
 login local
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 10 15
 login
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
!
!
monitor session 1 source interface Te1/1
monitor session 1 destination analysis-module 2 data-port 1
scheduler runtime netinput 300
ntp clock-period 17180000
ntp server 10.86.214.4
ntp server 10.86.208.4
ntp server 10.86.210.4
no cns aaa enable
end

sh-ace2-6k-2#
sh-ace2-6k-2#term length 24
sh-ace2-6k-2#term width 80
sh-ace2-6k-2#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k-2(config)#logging console
sh-ace2-6k-2(config)#end
sh-ace2-6k-2#
```

Return to SH-ACE2-6K-2, page 247

SH-ACE2-6K-3

Go to Basic Topology: Test Device Configurations, page 212

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

```
sh-ace2-6k3#show running-config
Load for five secs: 2%/1%; one minute: 1%; five minutes: 1%
```

```
Time source is hardware calendar, *13:32:17.880 UTC Fri Oct 17 2008

Building configuration...

Current configuration : 11134 bytes
!
upgrade fpd auto
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service counters max age 10
!
hostname sh-ace2-6k3
!
boot system flash disk0:s72033-adventerprisek9_wan-mz.122-18.SXF13.bin
enable password bxb-safeharbor
!
username cisco secret 5 $1$iSvx$pqbTrDQj4UrKmfpvQPGJx1
no aaa new-model
svclc autostate
svclc multiple-vlan-interfaces
svclc module 1 vlan-group 23456
svclc vlan-group 23456  83
firewall module 1 vlan-group 23456
analysis module 2 management-port access-vlan 83
ip subnet-zero
!
!
!
ipv6 mfib hardware-switching replication-mode ingress
vtp mode transparent
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
no mls acl tcam share-global
mls cef error action freeze
!
!
!
!
!
!
!
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
!
spanning-tree mode rapid-pvst
spanning-tree extend system-id
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
port-channel per-module load-balance
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
vlan 2-6,10,16,21,31,83
!
vlan 88
 remote-span
!
```

```
vlan 97,99,105-106,120,130,2830
!
!
!
!
interface Port-channel1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel8
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel98
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel99
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface GigabitEthernet4/1
 switchport
 switchport access vlan 83
 switchport mode access
 no ip address
```

```
!
interface GigabitEthernet4/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 spanning-tree portfast
!
interface GigabitEthernet4/3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet4/4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet4/5
 no ip address
 shutdown
!
interface GigabitEthernet4/6
 no ip address
 shutdown
!
interface GigabitEthernet4/7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet4/8
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet4/9
 no ip address
 shutdown
!
interface GigabitEthernet4/10
 no ip address
 shutdown
!
interface GigabitEthernet4/11
 no ip address
 shutdown
!
interface GigabitEthernet4/12
 no ip address
 shutdown
```

```
!
interface GigabitEthernet4/13
 switchport
 switchport access vlan 10
 no ip address
!
interface GigabitEthernet4/14
 switchport
 switchport access vlan 10
 no ip address
!
interface GigabitEthernet4/15
 no ip address
 shutdown
!
interface GigabitEthernet4/16
 no ip address
 shutdown
!
interface GigabitEthernet4/17
 no ip address
 shutdown
!
interface GigabitEthernet4/18
 no ip address
 shutdown
!
interface GigabitEthernet4/19
 switchport
 switchport access vlan 10
 no ip address
!
interface GigabitEthernet4/20
 no ip address
 shutdown
!
interface GigabitEthernet4/21
 no ip address
 shutdown
!
interface GigabitEthernet4/22
 no ip address
 shutdown
!
interface GigabitEthernet4/23
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 10,83,2830
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/24
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 10,83,2830
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/25
 no ip address
 shutdown
!
interface GigabitEthernet4/26
 no ip address
 shutdown
!
interface GigabitEthernet4/27
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 7 mode desirable
!
interface GigabitEthernet4/28
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 8 mode desirable
!
interface GigabitEthernet4/29
 no ip address
 shutdown
!
interface GigabitEthernet4/30
 no ip address
 shutdown
!
interface GigabitEthernet4/31
 no ip address
 shutdown
!
interface GigabitEthernet4/32
 no ip address
 shutdown
!
interface GigabitEthernet4/33
 no ip address
 shutdown
!
interface GigabitEthernet4/34
 no ip address
 shutdown
!
interface GigabitEthernet4/35
 switchport
 switchport access vlan 16
 switchport mode access
 no ip address
!
interface GigabitEthernet4/36
 switchport
 switchport access vlan 16
 switchport mode access
 no ip address
!
interface GigabitEthernet4/37
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 channel-group 99 mode desirable
!
interface GigabitEthernet4/38
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 channel-group 98 mode desirable
!
interface GigabitEthernet4/39
 no ip address
 shutdown
!
interface GigabitEthernet4/40
 switchport
 switchport access vlan 555
 switchport mode access
 no ip address
!
interface GigabitEthernet4/41
 switchport
 switchport access vlan 555
 switchport mode access
 no ip address
!
interface GigabitEthernet4/42
 no ip address
 shutdown
!
interface GigabitEthernet4/43
 no ip address
 shutdown
!
interface GigabitEthernet4/44
 no ip address
 shutdown
!
interface GigabitEthernet4/45
 no ip address
 shutdown
!
interface GigabitEthernet4/46
 switchport
 switchport access vlan 10
 no ip address
!
interface GigabitEthernet4/47
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 105,106
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/48
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 105,106
 switchport mode trunk
 no ip address
!
interface GigabitEthernet5/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
```

```
!
interface GigabitEthernet5/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet5/3
 no ip address
 shutdown
!
interface GigabitEthernet5/4
 no ip address
 shutdown
!
interface GigabitEthernet5/5
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet5/6
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2,88
 switchport mode trunk
 no ip address
 channel-group 3 mode desirable
!
interface GigabitEthernet5/7
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 7 mode desirable
!
interface GigabitEthernet5/8
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 120,130,2830
 switchport mode trunk
 no ip address
 channel-group 8 mode desirable
!
interface GigabitEthernet5/9
 no ip address
 shutdown
!
interface GigabitEthernet5/10
 no ip address
 shutdown
!
interface GigabitEthernet5/11
 no ip address
 shutdown
!
interface GigabitEthernet5/12
 no ip address
 shutdown
```

```
!
interface GigabitEthernet5/13
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 channel-group 99 mode desirable
!
interface GigabitEthernet5/14
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 97,99
 switchport mode trunk
 no ip address
 channel-group 98 mode desirable
!
interface GigabitEthernet5/15
 no ip address
 shutdown
!
interface GigabitEthernet5/16
 no ip address
 shutdown
!
interface GigabitEthernet6/1
 no ip address
 shutdown
!
interface GigabitEthernet6/2
 switchport
 switchport access vlan 83
 switchport mode access
 no ip address
 shutdown
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan2
 ip address 10.2.0.253 255.255.255.0
!
interface Vlan4
 ip address 10.4.0.1 255.255.255.0
!
interface Vlan5
 ip address 10.5.0.1 255.255.255.0
!
interface Vlan6
 ip address 10.6.0.1 255.255.255.0
!
interface Vlan10
 ip address 10.1.0.1 255.255.255.0
!
interface Vlan16
 ip address 192.168.16.253 255.255.255.0
!
interface Vlan21
 ip address 193.0.14.1 255.255.255.0
!
interface Vlan83
 ip address 10.86.83.156 255.255.255.0
!
```

```
interface Vlan105
 ip address 192.168.105.253 255.255.255.0
!
interface Vlan106
 ip address 192.168.106.253 255.255.255.0
!
router ospf 105
 router-id 192.168.105.253
 log-adjacency-changes
 network 192.168.105.0 0.0.0.255 area 0
 network 192.168.106.0 0.0.0.255 area 0
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.86.83.1
ip route 10.3.0.0 255.255.255.0 10.1.0.111
ip route 172.28.0.0 255.252.0.0 10.2.0.254
ip route 192.168.120.0 255.255.255.0 10.2.0.254
ip route 192.168.130.0 255.255.255.0 10.2.0.254
!
no ip http server
!
!
snmp-server community public RW
!
!
control-plane
!
!
!
dial-peer cor custom
!
!
!
!
line con 0
 exec-timeout 0 0
line vty 0 4
 session-timeout 180
 exec-timeout 720 0
 privilege level 15
 password bxb-safeharbor
 login local
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 5 9
 exec-timeout 720 0
 password bxb-safeharbor
 login local
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 10 15
 login
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
!
!
monitor session 1 destination analysis-module 2 data-port 1
monitor session 1 source remote vlan 88
monitor session 2 source interface Po1
monitor session 2 destination analysis-module 2 data-port 2
no cns aaa enable
end

sh-ace2-6k3#
sh-ace2-6k3#term length 24
```

```
sh-ace2-6k3#term width 80
sh-ace2-6k3#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k3(config)#logging console
sh-ace2-6k3(config)#end
sh-ace2-6k3#
```

Return to SH-ACE2-6K-3, page 280

SH-ACE2-6K-4

Go to Basic Topology: Test Device Configurations, page 212

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

```
sh-ace2-6k4#show running-config
Load for five secs: 0%/0%; one minute: 0%; five minutes: 0%
Time source is hardware calendar, *13:45:18.344 UTC Fri Oct 17 2008

Building configuration...

Current configuration : 9764 bytes
!
upgrade fpd auto
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service counters max age 10
!
hostname sh-ace2-6k4
!
boot system flash disk0:s72033-adventerprisek9_wan-mz.122-18.SXF13.bin
!
username cisco secret 5 $1$R3AT$TCtBy3t1AfYQ7gR3mMMnS0
no aaa new-model
analysis module 2 management-port access-vlan 83
ip subnet-zero
!
!
!
ipv6 mfib hardware-switching replication-mode ingress
vtp mode transparent
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
no mls acl tcam share-global
mls cef error action freeze
!
!
!
!
!
!
!
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
!
spanning-tree mode rapid-pvst
```

```
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
port-channel per-module load-balance
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
vlan 9-12,16,20,28-29,50,83,98,160,281-289,2810-2820
!
!
!
!
interface Port-channel2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface Port-channel4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 logging event link-status
 logging event bundle-status
!
interface GigabitEthernet4/1
 switchport
 switchport access vlan 83
 switchport mode access
 no ip address
!
interface GigabitEthernet4/2
 no ip address
 shutdown
!
interface GigabitEthernet4/3
 no ip address
 shutdown
!
interface GigabitEthernet4/4
 no ip address
 shutdown
!
interface GigabitEthernet4/5
 no ip address
 shutdown
!
interface GigabitEthernet4/6
 no ip address
 shutdown
!
interface GigabitEthernet4/7
 no ip address
 shutdown
!
interface GigabitEthernet4/8
 no ip address
 shutdown
!
interface GigabitEthernet4/9
 no ip address
 shutdown
!
interface GigabitEthernet4/10
 no ip address
 shutdown
!
interface GigabitEthernet4/11
 no ip address
 shutdown
!
interface GigabitEthernet4/12
 no ip address
 shutdown
!
interface GigabitEthernet4/13
 switchport
 switchport access vlan 29
 switchport mode access
 no ip address
!
interface GigabitEthernet4/14
 switchport
 switchport access vlan 29
 switchport mode access
 no ip address
!
interface GigabitEthernet4/15
 no ip address
 shutdown
!
interface GigabitEthernet4/16
 no ip address
 shutdown
!
interface GigabitEthernet4/17
 no ip address
 shutdown
!
interface GigabitEthernet4/18
 no ip address
 shutdown
!
interface GigabitEthernet4/19
 no ip address
 shutdown
!
interface GigabitEthernet4/20
 no ip address
 shutdown
!
interface GigabitEthernet4/21
 no ip address
 shutdown
!
interface GigabitEthernet4/22
 no ip address
 shutdown
!
interface GigabitEthernet4/23
 no ip address
 shutdown
!
interface GigabitEthernet4/24
 no ip address
 shutdown
!
interface GigabitEthernet4/25
 switchport
 switchport trunk allowed vlan 83,283,284
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/26
 switchport
 switchport trunk allowed vlan 83,283,284
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/27
 switchport
 switchport trunk allowed vlan 83,283,284
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/28
 switchport
 switchport trunk allowed vlan 83,283,284
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/29
 no ip address
 shutdown
!
interface GigabitEthernet4/30
 no ip address
 shutdown
!
interface GigabitEthernet4/31
 no ip address
 shutdown
!
interface GigabitEthernet4/32
 no ip address
 shutdown
!
interface GigabitEthernet4/33
 no ip address
 shutdown
!
interface GigabitEthernet4/34
 no ip address
 shutdown
!
interface GigabitEthernet4/35
 no ip address
 shutdown
!
interface GigabitEthernet4/36
 no ip address
 shutdown
!
interface GigabitEthernet4/37
 switchport
 switchport trunk allowed vlan 28,281
 switchport mode trunk
 no ip address
!
interface GigabitEthernet4/38
 no ip address
 shutdown
!
interface GigabitEthernet4/39
 no ip address
 shutdown
!
interface GigabitEthernet4/40
 no ip address
 shutdown
!
interface GigabitEthernet4/41
 no ip address
 shutdown
!
interface GigabitEthernet4/42
 no ip address
 shutdown
!
interface GigabitEthernet4/43
 switchport
 switchport access vlan 29
 switchport mode access
 no ip address
!
interface GigabitEthernet4/44
 no ip address
 shutdown
!
interface GigabitEthernet4/45
 switchport
 switchport access vlan 29
 switchport mode access
 no ip address
!
interface GigabitEthernet4/46
 no ip address
 shutdown
!
interface GigabitEthernet4/47
 switchport
 switchport access vlan 29
 switchport mode access
 no ip address
!
interface GigabitEthernet4/48
 no ip address
 shutdown
!
interface TenGigabitEthernet5/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 2 mode desirable
!
interface TenGigabitEthernet5/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 2 mode desirable
!
interface TenGigabitEthernet5/3
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 4 mode desirable
!
interface TenGigabitEthernet5/4
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 29,98,160,281,283
 switchport mode trunk
 no ip address
 channel-group 4 mode desirable
!
interface GigabitEthernet6/1
 no ip address
 shutdown
!
interface GigabitEthernet6/2
 no ip address
 shutdown
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan28
 ip address 172.28.0.253 255.255.255.0
!
interface Vlan29
 ip address 172.29.0.253 255.255.255.0
!
interface Vlan83
 ip address 10.86.83.158 255.255.255.0
!
interface Vlan281
 ip address 172.28.1.253 255.255.255.0
!
interface Vlan282
 ip address 172.28.2.253 255.255.255.0
!
interface Vlan283
 ip address 172.28.3.253 255.255.255.0
!
interface Vlan284
 ip address 172.28.4.253 255.255.255.0
 standby 84 ip 172.28.4.254
 standby 84 priority 200
 standby 84 preempt
!
interface Vlan285
 ip address 172.28.5.253 255.255.255.0
!
interface Vlan286
 ip address 172.28.6.253 255.255.255.0
!
```

```
interface Vlan287
 ip address 172.28.7.253 255.255.255.0
!
interface Vlan288
 ip address 172.28.8.253 255.255.255.0
!
interface Vlan289
 ip address 172.28.9.253 255.255.255.0
!
interface Vlan2810
 ip address 172.28.10.253 255.255.255.0
!
interface Vlan2811
 ip address 172.28.11.253 255.255.255.0
!
interface Vlan2812
 ip address 172.28.12.253 255.255.255.0
!
interface Vlan2813
 ip address 172.28.13.253 255.255.255.0
!
interface Vlan2814
 ip address 172.28.14.253 255.255.255.0
!
interface Vlan2815
 ip address 172.28.15.253 255.255.255.0
!
interface Vlan2816
 ip address 172.28.16.253 255.255.255.0
!
interface Vlan2817
 ip address 172.28.17.253 255.255.255.0
!
interface Vlan2818
 ip address 172.28.18.253 255.255.255.0
!
interface Vlan2819
 ip address 172.28.19.253 255.255.255.0
!
interface Vlan2820
 ip address 172.28.20.253 255.255.255.0
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.86.83.1
ip route 10.1.0.0 255.255.128.0 172.29.0.1
ip route 10.1.0.240 255.255.255.252 172.28.3.1
ip route 10.128.0.0 255.128.0.0 Null0
ip route 172.19.1.0 255.255.255.0 Null0
ip route 172.25.0.0 255.255.0.0 Null0
ip route 192.168.106.0 255.255.255.0 172.29.0.4
ip route 192.168.120.0 255.255.255.0 172.29.0.1
ip route 192.168.120.209 255.255.255.255 172.29.0.11
ip route 192.168.120.210 255.255.255.255 172.29.0.11
ip route 192.168.130.0 255.255.255.0 172.28.1.4
ip route 192.168.130.70 255.255.255.254 172.28.1.7
ip route 192.168.130.74 255.255.255.254 172.28.1.13
ip route 192.168.130.78 255.255.255.254 172.28.1.10
ip route 192.168.130.82 255.255.255.254 172.28.1.16
!
no ip http server
!
access-list 100 deny   tcp any host 172.28.0.154 eq 84
access-list 100 permit ip any any
access-list 100 deny   tcp any host 172.28.4.244 eq 84
```

```
access-list 101 deny   tcp any host 172.28.0.154 eq ftp
access-list 101 permit ip any any
access-list 101 deny   tcp any host 172.28.4.244 eq ftp
access-list 102 deny   udp any host 172.28.0.154 eq tftp
access-list 102 permit ip any any
access-list 102 deny   udp any host 172.28.4.244 eq tftp
access-list 103 deny   tcp any host 172.28.0.154 eq 84
access-list 103 permit ip any any
access-list 104 deny   tcp any host 172.28.0.154 eq ftp
access-list 104 permit ip any any
access-list 105 deny   udp any host 172.28.0.154 eq tftp
access-list 105 permit ip any any
access-list 106 deny   tcp any host 172.28.0.153 eq www
access-list 106 permit ip any any
access-list 107 deny   tcp any host 172.29.0.244 eq 443
access-list 107 permit ip any any
!
snmp-server community public RW
!
!
control-plane
!
!
!
dial-peer cor custom
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 password bxb-safeharbor
 login local
 history size 256
 stopbits 1
line vty 0 4
 exec-timeout 720 0
 privilege level 15
 password bxb-safeharbor
 login local
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 5 9
 exec-timeout 720 0
 password bxb-safeharbor
 login local
 history size 256
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
line vty 10 15
 login
 transport input lat pad mop udptn telnet rlogin ssh nasi acercon
!
!
monitor session 1 source interface Po4 rx
monitor session 1 destination analysis-module 2 data-port 1
monitor session 2 source interface Gi4/36
monitor session 2 destination analysis-module 2 data-port 2
no cns aaa enable
end

sh-ace2-6k4#
sh-ace2-6k4#term length 24
sh-ace2-6k4#term width 80
```

```
sh-ace2-6k4#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sh-ace2-6k4(config)#logging console
sh-ace2-6k4(config)#end
sh-ace2-6k4#
```

Return to SH-ACE2-6K-4, page 290

6K-1 ACE1 All Context

Go to Basic Topology: Test Device Configurations, page 212

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

```
--- 09:56:17 ---

+++ 09:56:17 6K-1_ACE2-1 logging +++

--- 09:56:17 ---

+++ 09:57:12 6K-1_ACE2-1 ctxExec +++
show context


Number of Contexts = 10

Name: Admin , Id: 0
Config count: 307
Description:
Resource-class: default
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: 130-VRT-1 , Id: 1
Config count: 672
Description:
Resource-class: POINT_FIVE
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: 130-VRT-2 , Id: 2
Config count: 482
Description:
Resource-class: POINT_FIVE
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: 130-VRT-3 , Id: 3
Config count: 676
Description:
Resource-class: POINT_FIVE
```

```
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: A2 , Id: 4
Config count: 3036
Description:
Resource-class: 15_PERCENT
Vlans:  Vlan29, Vlan83, Vlan99, Vlan106, Vlan120, Vlan160
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: LARGE , Id: 5
Config count: 1
Description:
Resource-class: default
Vlans:  Vlan83, Vlan130, Vlan281, Vlan320, Vlan329
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: SH-Bridge , Id: 6
Config count: 2694
Description:
Resource-class: 10_PERCENT
Vlans:  Vlan99, Vlan106, Vlan283, Vlan2830
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: SH-Gold , Id: 7
Config count: 4287
Description:
Resource-class: 15_PERCENT
Vlans:  Vlan29, Vlan99, Vlan105, Vlan120
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: SH-LOAD , Id: 8
Config count: 847
Description:
Resource-class: 5_PERCENT
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: SH-Silver , Id: 9
Config count: 677
```

```
Description:
Resource-class: 10_PERCENT
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


6K-1_ACE2-1/Admin#
--- 09:57:12 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++



6K-1_ACE2-1/Admin#
--- 09:57:13 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++
show running-config

Generating configuration....


logging enable
logging standby
logging timestamp
logging trap 5
logging buffered 5
logging monitor 5
logging queue 150
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514

ldap-server host 172.29.0.244
aaa group server ldap SH_LDAP
  baseDN "dc=bxb-safeharbor,dc=com"

peer hostname 6K-2_ACE2-1

login timeout 0
hostname 6K-1_ACE2-1
boot system image:c6ace-t1k9-mz.A2_1_2.bin
shared-vlan-hostid 16
peer shared-vlan-hostid 15

resource-class 10_PERCENT
  limit-resource all minimum 10.00 maximum unlimited
  limit-resource buffer syslog minimum 10.00 maximum equal-to-min
  limit-resource sticky minimum 10.00 maximum equal-to-min
resource-class 15_PERCENT
  limit-resource all minimum 15.00 maximum unlimited
  limit-resource buffer syslog minimum 15.00 maximum equal-to-min
  limit-resource sticky minimum 15.00 maximum equal-to-min
resource-class 1PERCENT
  limit-resource all minimum 1.00 maximum equal-to-min
  limit-resource sticky minimum 1.00 maximum equal-to-min
resource-class 5_PERCENT
  limit-resource all minimum 5.00 maximum unlimited
  limit-resource buffer syslog minimum 5.00 maximum equal-to-min
  limit-resource sticky minimum 5.00 maximum equal-to-min
resource-class JUST_STICKY
```

```
  limit-resource all minimum 0.00 maximum unlimited
  limit-resource sticky minimum 1.00 maximum equal-to-min
resource-class MIN-ALL
  limit-resource all minimum 0.01 maximum equal-to-min
  limit-resource mgmt-connections minimum 1.00 maximum equal-to-min
  limit-resource rate syslog minimum 1.00 maximum equal-to-min
  limit-resource sticky minimum 0.01 maximum equal-to-min
resource-class OVER-LIMIT
  limit-resource all minimum 90.00 maximum equal-to-min
resource-class POINT_FIVE
  limit-resource all minimum 0.50 maximum equal-to-min
tacacs-server key 7 "vwjjzamggu"
tacacs-server host 10.86.83.215 key 7 "vwjjzamggu"
aaa group server tacacs+ SafeHarbor-Tacacs
  server 10.86.83.215
aaa group server tacacs+ sh-admin
  server 10.86.83.215

clock timezone standard EST
clock summer-time standard EDT
aaa authentication login default group sh-admin local
aaa accounting default group sh-admin
aaa authentication login error-enable

access-list anyone line 10 extended permit ip any any



class-map type management match-any REMOTE-ACCESS_ALL
  description "Remote Management for ALL"
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
class-map type management match-any REMOTE-ACCESS_LOCAL
  description "Remote Management for BXB only"
  2 match protocol telnet source-address 10.80.0.0 255.248.0.0
  4 match protocol ssh source-address 10.80.0.0 255.248.0.0
  5 match protocol icmp source-address 10.80.0.0 255.248.0.0
  6 match protocol http source-address 10.80.0.0 255.248.0.0
  7 match protocol snmp source-address 10.80.0.0 255.248.0.0
  8 match protocol https source-address 10.80.0.0 255.248.0.0
  9 match protocol ssh source-address 161.44.64.0 255.255.252.0
  10 match protocol icmp source-address 161.44.64.0 255.255.252.0
  11 match protocol http source-address 161.44.64.0 255.255.252.0
  12 match protocol snmp source-address 161.44.64.0 255.255.252.0
  13 match protocol https source-address 161.44.64.0 255.255.252.0
  14 match protocol telnet source-address 161.44.64.0 255.255.252.0

policy-map type management first-match REMOTE-MGMT
  class REMOTE-ACCESS_ALL
    permit

interface vlan 83
  ip address 10.86.83.160 255.255.255.0
  peer ip address 10.86.83.161 255.255.255.0
  service-policy input REMOTE-MGMT
  no shutdown

ft interface vlan 900
  ip address 192.168.1.1 255.255.255.0
  peer ip address 192.168.1.2 255.255.255.0
  no shutdown

ft peer 1
  heartbeat interval 300
  heartbeat count 10
  ft-interface vlan 900
ft group 1
  peer 1
  priority 200
  associate-context Admin
  inservice

domain SH-Admin-Domain
  add-object all
domain art
  add-object all

role CiscoRole
  rule 1 permit modify
role SH-Role
  rule 1 permit create
  rule 2 permit modify

ip route 0.0.0.0 0.0.0.0 10.86.83.1

context 130-VRT-1
  allocate-interface vlan 130
  allocate-interface vlan 281
  member POINT_FIVE
context 130-VRT-2
  allocate-interface vlan 130
  allocate-interface vlan 281
  member POINT_FIVE
context 130-VRT-3
  allocate-interface vlan 130
  allocate-interface vlan 281
  member POINT_FIVE
context A2
  allocate-interface vlan 29
  allocate-interface vlan 83
  allocate-interface vlan 99
  allocate-interface vlan 106
  allocate-interface vlan 120
  allocate-interface vlan 160
  member 15_PERCENT
context LARGE
  allocate-interface vlan 83
  allocate-interface vlan 130
  allocate-interface vlan 281
  allocate-interface vlan 320
  allocate-interface vlan 329
context SH-Bridge
  allocate-interface vlan 99
  allocate-interface vlan 106
  allocate-interface vlan 283
  allocate-interface vlan 2830
  member 10_PERCENT
context SH-Gold
  allocate-interface vlan 29
  allocate-interface vlan 99
  allocate-interface vlan 105
  allocate-interface vlan 120
  member 15_PERCENT
context SH-LOAD
```

```
  allocate-interface vlan 130
  allocate-interface vlan 281
  member 5_PERCENT
context SH-Silver
  allocate-interface vlan 130
  allocate-interface vlan 281
  member 10_PERCENT

snmp-server community ACE-private group Network-Monitor
snmp-server community ACE-public group Network-Monitor

snmp-server host 10.86.83.236 traps version 2c ACE-public

snmp-server enable traps snmp coldstart
snmp-server enable traps virtual-context
snmp-server enable traps license
snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown

ft group 2
  peer 1
  priority 200
  associate-context SH-Gold
  inservice
ft group 3
  peer 1
  priority 200
  associate-context SH-Silver
  inservice
ft group 4
  peer 1
  priority 200
  associate-context SH-LOAD
  inservice
ft group 5
  peer 1
  priority 200
  associate-context LARGE
  inservice
ft group 6
  peer 1
  priority 200
  associate-context 130-VRT-1
  inservice
ft group 7
  peer 1
  priority 200
  associate-context 130-VRT-2
  inservice
ft group 8
  peer 1
  priority 200
  associate-context 130-VRT-3
  inservice
ft group 10
  peer 1
  priority 200
  associate-context SH-Bridge
  inservice
ft group 11
```

```
  peer 1
  priority 200
  associate-context A2
  inservice
username admin password 5 $1$YBxN8VJN$KqLLEtXDnrXm6M9vozWI40  role Admin domain
default-domain
username www password 5 $1$UZIiwUk7$QMVYN1JASaycabrHkhGcS/  role Admin domain
default-domain
username netmon password 5 $1$rYHZjgGm$iWMZNUv/9oHEUsPxZM.tq.  role Network-Monitor domain
default-domain
username sh password 5 $1$NZk8T1Xf$uhYawy7j7m9Q2/EUBx5Ti/  role Admin domain
default-domain
username art password 5 $1$pzEPn3Gk$tuSFSxtxQuuB3O6NzIGLr0  role Admin domain
default-domain
username bxb-safeharbor password 5 $1$IvCJDwOx$eanTW5mtGQYWB47SPhqZx.  role
Network-Monitor domain default-domain
username root password 5 $1$mbLJQdlS$mQAerSjxe0E4qxlx8VKO4.  role Admin domain
default-domain
ssh key rsa 4096 force
ssh key rsa1 4096 force


6K-1_ACE2-1/Admin#
--- 09:57:13 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++



6K-1_ACE2-1/Admin#
--- 09:57:13 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-1


6K-1_ACE2-1/130-VRT-1#


6K-1_ACE2-1/130-VRT-1# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:13 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-1


6K-1_ACE2-1/130-VRT-1# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging buffered 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7
```

```
crypto chaingroup CHAIN1
aaa authentication login error-enable

access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-1_probe_GEN-HTTP"
  connection term forced
probe https GEN_HTTPS
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-1_probe_GEN-HTTPS"
  connection term forced


parameter-map type http HTTP_PARAM
  server-conn reuse
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 20
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version TLS1
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  nagle
  syn-data drop
  exceed-mss allow
  urgent-flag clear

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  ssl advanced-options PARM_ACE_TERM

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host ONE-IIS-SERVER
  rserver BRG-IIS-1
```

```
    inservice
serverfarm host TCP-NORM-FARM
  predictor leastconns
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-IIS-3
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver BRG-LINUX-3
    inservice
serverfarm host TELNET-NORM-TEST
  rserver BRG-LINUX-1
    inservice

sticky http-cookie SSL_TERM_COOKIE GROUP_10
  cookie insert browser-expire
  serverfarm ACE_TERM_SERVERS_CLEAR
sticky http-cookie SSL_INIT_COOKIE GROUP_20
  cookie insert browser-expire
  serverfarm ACE_INIT_SERVERS_SSL
sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
  cookie insert browser-expire
  serverfarm ACE_END_TO_END_SERVERS_SSL
sticky http-cookie NON_SSL_TESTING GROUP_40
  cookie insert browser-expire
  serverfarm NON-SSL-TEST
sticky ip-netmask 255.255.255.0 address both NEW_GROUP
  serverfarm NON-SSL-TEST

class-map match-all GENERIC
class-map type http loadbalance match-all LB_CLASS_HTTP
  2 match http url .*
class-map match-all NON-SSL_TEST
  description NON-SSL_TEST
class-map match-all SSL_END_TO_END_13
  description STICKY FOR SSL TESTING
  3 match source-address 192.168.130.0 255.255.255.0
class-map match-all TCP-NORM-TEST
  description TCP NORM TEST
  2 match virtual-address 192.168.130.137 tcp eq 22

policy-map type management first-match POLICY_MGMT

policy-map type loadbalance first-match GENERIC
  class LB_CLASS_HTTP
    serverfarm ONE-IIS-SERVER
  class class-default
    serverfarm ONE-IIS-SERVER
policy-map type loadbalance first-match NON_SSL_TESTING
  class LB_CLASS_HTTP
    serverfarm NON-SSL-TEST
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "NON_SSL"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
  class LB_CLASS_HTTP
    serverfarm ACE_END_TO_END_SERVERS_SSL
    insert-http SRC_IP header-value "%is"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_END_TO_END"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_INIT
  class LB_CLASS_HTTP
    serverfarm ACE_INIT_SERVERS_SSL
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "SSL_INIT"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_TERM
  class LB_CLASS_HTTP
    serverfarm ACE_TERM_SERVERS_CLEAR
    insert-http I_AM header-value "SSL_TERM"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "is"
policy-map type loadbalance first-match TCP-NORM-TESTING
  class LB_CLASS_HTTP
    serverfarm TCP-NORM-FARM
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "TCP-NORM-TEST"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
policy-map type loadbalance first-match TELNET-NORM_TESTING
  class class-default
    serverfarm TELNET-NORM-TEST

policy-map type inspect http all-match INSPECT_GOOD_HTTP

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_END_TO_END_13
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class TCP-NORM-TEST
    loadbalance vip inservice
    loadbalance policy TCP-NORM-TESTING
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
  class GENERIC
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM

interface vlan 130
  ip address 192.168.130.32 255.255.255.0
  alias 192.168.130.31 255.255.255.0
  peer ip address 192.168.130.33 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.74 192.168.130.74 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.14 255.255.255.0
  alias 172.28.1.13 255.255.255.0
  peer ip address 172.28.1.15 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.75 192.168.130.75 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown

ip route 10.1.0.0 255.255.255.0 192.168.130.254
ip route 192.168.120.0 255.255.255.0 192.168.130.254
username admin password 5 $1$Dyc2pgwi$j.ib5ZiBo.7Xm7J7kEmrW/  role Admin domain
default-domain
username netmon password 5 $1$G6bSxh54$7Jop/aBTWNrdN2iS18JX2.  role Admin domain
default-domain




6K-1_ACE2-1/130-VRT-1# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:13 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-1


6K-1_ACE2-1/130-VRT-1#


6K-1_ACE2-1/130-VRT-1# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:13 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-2


6K-1_ACE2-1/130-VRT-2#


6K-1_ACE2-1/130-VRT-2# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:13 ---

+++ 09:57:13 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-2


6K-1_ACE2-1/130-VRT-2# show running-config

Generating configuration....
```

```
logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging buffered 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7



crypto chaingroup CHAIN1
aaa authentication login error-enable

access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-2_probe_GEN-HTTP"
  connection term forced
probe https GEN_HTTPS
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 407
  header Via header-value "130-VRT-2_probe_GEN-HTTPS"
  connection term forced


parameter-map type http HTTP_PARAM
  server-conn reuse
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 20
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version TLS1
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  urgent-flag clear
parameter-map type connection TEST

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice
```

```
ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  ssl advanced-options PARM_ACE_TERM

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host ONE-IIS-SERVER
  rserver BRG-IIS-1
    inservice
serverfarm host TCP-NORM-FARM
  predictor leastconns
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-IIS-3
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver BRG-LINUX-3
    inservice
serverfarm host TELNET-NORM-TEST
  rserver BRG-LINUX-1
    inservice

class-map match-all GENERIC
class-map type http loadbalance match-all LB_CLASS_HTTP
  2 match http url .*
  3 match source-address 192.168.130.0 255.255.255.0
class-map match-all NON-SSL_TEST
  description NON-SSL_TEST
class-map match-all SSL_END_TO_END_13
  description STICKY FOR SSL TESTING
  3 match source-address 192.168.130.0 255.255.255.0
class-map match-all TCP-NORM-TEST
  description TCP NORM TEST
  2 match virtual-address 192.168.130.171 tcp eq 22

policy-map type management first-match POLICY_MGMT

policy-map type inspect http all-match INSPECT_GOOD_HTTP

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_END_TO_END_13
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class TCP-NORM-TEST
    loadbalance vip inservice
    loadbalance vip icmp-reply
    connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
  class GENERIC
    appl-parameter http advanced-options HTTP_PARAM

interface vlan 130
  ip address 192.168.130.35 255.255.255.0
  alias 192.168.130.34 255.255.255.0
  peer ip address 192.168.130.36 255.255.255.0
  access-group input eveyone
  service-policy input POLICY_MGMT
  no shutdown
interface vlan 281
  ip address 172.28.1.11 255.255.255.0
  alias 172.28.1.10 255.255.255.0
  peer ip address 172.28.1.12 255.255.255.0
  access-group input eveyone
  service-policy input POLICY_MGMT
  no shutdown

ip route 10.1.0.0 255.255.255.0 192.168.130.254
username admin password 5 $1$Ot5XpO4e$KlAT/S/YguBVfXRjCgGZZ1  role Admin domain
default-domain
username netmon password 5 $1$RaQYCihb$n9azptTRpeFHrmOBsFDaB1  role Network-Monitor domain
default-domain




6K-1_ACE2-1/130-VRT-2# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:14 ---

+++ 09:57:14 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-2


6K-1_ACE2-1/130-VRT-2#


6K-1_ACE2-1/130-VRT-2# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:14 ---

+++ 09:57:14 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-3


6K-1_ACE2-1/130-VRT-3#


6K-1_ACE2-1/130-VRT-3# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:14 ---

+++ 09:57:14 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-3


6K-1_ACE2-1/130-VRT-3# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging buffered 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7



crypto chaingroup CHAIN1
aaa authentication login error-enable

access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-3_probe_GEN-HTTP"
  connection term forced
probe https GEN_HTTPS
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-3_probe_GEN-HTTPS"
  connection term forced


parameter-map type http HTTP_PARAM
  server-conn reuse
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 20
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version TLS1
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  nagle
  syn-data drop
  exceed-mss allow
  urgent-flag clear

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
```

```
ssl-proxy service ACE_END_TO_END
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  ssl advanced-options PARM_ACE_TERM

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host ONE-IIS-SERVER
  rserver BRG-IIS-1
    inservice
serverfarm host TCP-NORM-FARM
  predictor leastconns
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-IIS-3
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver BRG-LINUX-3
    inservice
serverfarm host TELNET-NORM-TEST
  rserver BRG-LINUX-1
    inservice

sticky http-cookie SSL_TERM_COOKIE GROUP_10
  cookie insert browser-expire
  serverfarm ACE_TERM_SERVERS_CLEAR
sticky http-cookie SSL_INIT_COOKIE GROUP_20
  cookie insert browser-expire
  serverfarm ACE_INIT_SERVERS_SSL
sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
  cookie insert browser-expire
  serverfarm ACE_END_TO_END_SERVERS_SSL
sticky http-cookie NON_SSL_TESTING GROUP_40
  cookie insert browser-expire
  serverfarm NON-SSL-TEST
sticky ip-netmask 255.255.255.0 address both NEW_GROUP
  serverfarm NON-SSL-TEST

class-map match-all GENERIC
class-map type http loadbalance match-all LB_CLASS_HTTP
  2 match http url .*
class-map match-all NON-SSL_TEST
  description NON-SSL_TEST
class-map match-all SSL_END_TO_END_13
  description STICKY FOR SSL TESTING
  3 match source-address 192.168.130.0 255.255.255.0
class-map match-all TCP-NORM-TEST
  description TCP NORM TEST
  2 match virtual-address 192.168.130.177 tcp eq 22

policy-map type management first-match POLICY_MGMT

policy-map type loadbalance first-match GENERIC
  class LB_CLASS_HTTP
    serverfarm ONE-IIS-SERVER
  class class-default
    serverfarm ONE-IIS-SERVER
policy-map type loadbalance first-match NON_SSL_TESTING
  class LB_CLASS_HTTP
    serverfarm NON-SSL-TEST
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http I_AM header-value "NON_SSL"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match PLBSF_L4
  class class-default
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
  class LB_CLASS_HTTP
    serverfarm ACE_END_TO_END_SERVERS_SSL
    insert-http I_AM header-value "SSL_END_TO_END"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_INIT
  class LB_CLASS_HTTP
    serverfarm ACE_INIT_SERVERS_SSL
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http I_AM header-value "SSL_INIT"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_TERM
  class LB_CLASS_HTTP
    serverfarm ACE_TERM_SERVERS_CLEAR
    insert-http SRC_IP header-value "is"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM"
policy-map type loadbalance first-match TCP-NORM-TESTING
  class LB_CLASS_HTTP
    serverfarm TCP-NORM-FARM
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "TCP-NORM-TEST"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match TELNET-NORM_TESTING
  class class-default
    serverfarm TELNET-NORM-TEST

policy-map type inspect http all-match INSPECT_GOOD_HTTP

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_END_TO_END_13
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class TCP-NORM-TEST
    loadbalance vip inservice
    loadbalance policy TCP-NORM-TESTING
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
  class GENERIC
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM

interface vlan 130
  ip address 192.168.130.38 255.255.255.0
  alias 192.168.130.37 255.255.255.0
  peer ip address 192.168.130.39 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.82 192.168.130.82 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.17 255.255.255.0
  alias 172.28.1.16 255.255.255.0
  peer ip address 172.28.1.18 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.83 192.168.130.83 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown

ip route 10.1.0.0 255.255.255.0 192.168.130.254
ip route 192.168.120.0 255.255.255.0 192.168.130.254
username admin password 5 $1$3ZmJoRxI$eDb3O981o6KQa8Bps/hoG0  role Admin domain
default-domain
username netmon password 5 $1$jTVnTHDe$tKnwxwOpXa9DmihIlfmKk0  role Network-Monitor domain
default-domain




6K-1_ACE2-1/130-VRT-3# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:15 ---

+++ 09:57:15 6K-1_ACE2-1 ctxExec +++
changeto 130-VRT-3


6K-1_ACE2-1/130-VRT-3#


6K-1_ACE2-1/130-VRT-3# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:15 ---

+++ 09:57:15 6K-1_ACE2-1 ctxExec +++
changeto LARGE


6K-1_ACE2-1/LARGE#


6K-1_ACE2-1/LARGE# changeto Admin
```

```
6K-1_ACE2-1/Admin#
--- 09:57:15 ---

+++ 09:57:15 6K-1_ACE2-1 ctxExec +++
changeto LARGE


6K-1_ACE2-1/LARGE# show running-config

Generating configuration....


logging monitor 5









6K-1_ACE2-1/LARGE# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:16 ---

+++ 09:57:16 6K-1_ACE2-1 ctxExec +++
changeto LARGE


6K-1_ACE2-1/LARGE#


6K-1_ACE2-1/LARGE# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:16 ---

+++ 09:57:16 6K-1_ACE2-1 ctxExec +++
changeto SH-Gold


6K-1_ACE2-1/SH-Gold#


6K-1_ACE2-1/SH-Gold# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:16 ---

+++ 09:57:16 6K-1_ACE2-1 ctxExec +++
changeto SH-Gold


6K-1_ACE2-1/SH-Gold# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging history 5
logging buffered 5
logging queue 200
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514


tacacs-server key 7 "vwjjzamggu"
tacacs-server host 172.29.0.235 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.236 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.237 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-gold
  server 172.29.0.235
  server 172.29.0.236
  server 172.29.0.237

arp 192.168.120.80 00.00.11.11.22.22
arp 172.29.0.200 00.00.22.22.44.44
arp learned-interval 60
aaa authentication login default group sh-gold local
aaa accounting default group sh-gold
aaa authentication login error-enable

access-list ICMP-ONLY line 8 extended permit icmp any any
access-list NAT_ACCESS line 20 extended permit tcp any host 172.29.0.121
access-list NAT_ACCESS line 30 extended permit tcp any host 172.29.0.122
access-list NAT_ACCESS line 40 extended permit tcp any host 172.29.0.140
access-list anyone-ip line 10 extended permit ip any any
access-list anyone-tcp line 10 extended permit tcp any any

kalap udp
  ip address 10.1.0.214 encryption md5 safeharbor

script file 1 FTP_PROBE_SCRIPT
script file 2 TFTP_PROBE
script file 3 LDAP_PROBE


probe icmp FA-PURGE-ICMP
  ip address 172.29.0.253 routed
  interval 5
  passdetect interval 2
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
probe ftp FTP
  interval 10
  faildetect 2
  passdetect interval 10
  receive 5
  expect status 220 220
  open 3
probe icmp HA-ICMP
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:1755
  port 1755
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:554
  port 554
  interval 2
  faildetect 2
  passdetect interval 2
probe http HTTP
  interval 5
  passdetect interval 10
  receive 5
  expect status 200 200
  open 3
probe icmp ICMP
  interval 10
  faildetect 2
  passdetect interval 10
probe dns PRB-DNS1
  description "all good addresses"
  interval 5
  passdetect interval 2
  domain www1.safeharbor.com
  expect address 1.1.1.1
  expect address 1.1.1.2
  expect address 1.1.1.3
probe dns PRB-DNS2
  description "one good address"
  interval 5
  passdetect interval 2
  domain www2.safeharbor.com
  expect address 2.1.1.1
probe dns PRB-DNS3
  description "2 good addresses, bumpy case"
  interval 5
  passdetect interval 2
  domain WwW3.SaFeHaRbOr.CoM
  expect address 3.1.1.3
  expect address 3.1.1.2
probe dns PRB-DNS4
  description "one good and one bad address"
  interval 5
  passdetect interval 2
  domain www4.safeharbor.com
  expect address 192.168.1.4
  expect address 4.1.1.1
probe dns PRB-DNS5
  description "all bad addresses"
  interval 5
  passdetect interval 2
  domain www4.safeharbor.com
  expect address 192.168.1.5
  expect address 192.168.1.6
  expect address 1.1.1.1
probe dns PRB-DNS6:2222
  description "dns not running on this port, but addresses good"
  port 2222
  interval 5
  passdetect interval 2
  domain www1.safeharbor.com
  expect address 1.1.1.1
  expect address 1.1.1.2
  expect address 1.1.1.3
probe http PRB-HTTP:84
  port 84
  interval 5
  passdetect interval 4
  passdetect count 10
  expect status 200 200
  connection term forced
probe http PRB-HTTP:85
  description Server RST 1byte data
  port 85
  interval 5
  passdetect interval 2
  expect status 200 200
  connection term forced
probe http PRB-HTTP:86
  description Server RST 3200byte data
  port 86
  interval 5
  passdetect interval 2
  expect status 200 200
  connection term forced
probe http PRB-HTTP:87
  description Server FIN 1byte data
  port 87
  interval 5
  passdetect interval 2
  expect status 200 200
probe http PRB-HTTP:88
  description Server FIN 3200byte data
  port 88
  interval 5
  passdetect interval 2
  expect status 200 200
probe https PRB-SSL:443
  interval 5
  passdetect interval 10
  request method get url /index.txt
  expect status 200 200
  header Via header-value "PRB-SSL:443 Probe Header"
  hash
probe icmp PRED-PING
  ip address 172.29.0.243 routed
  interval 5
  faildetect 2
  passdetect interval 2
probe radius RADIUS
  interval 2
  faildetect 2
  passdetect interval 2
  credentials lab labtest1 secret ace
probe scripted SCRIPT_FTP:21
  interval 10
  passdetect interval 2
  passdetect count 2
  receive 5
  script FTP_PROBE_SCRIPT /home/lab/ftp-files/file01.log lab labtest1
probe scripted SCRIPT_LDAP
  interval 10
  passdetect interval 5
  passdetect count 2
  receive 5
  script LDAP_PROBE
```

```
probe scripted SCRIPT_TFTP
  interval 10
  passdetect interval 5
  passdetect count 2
  receive 5
  script TFTP_PROBE "large file name to test the tftp scripted probe.exe"
probe https SSL
  interval 5
  passdetect interval 10
  expect status 200 299
  header Via header-value "ACE_Gold_SSL"
  connection term forced
probe https SSL-445:FIN
  port 445
  interval 5
  passdetect interval 10
  expect status 200 200
probe https SSL-445:RST
  port 445
  interval 5
  passdetect interval 10
  expect status 200 200
  connection term forced
probe tcp TCP
  interval 5
  faildetect 2
  passdetect interval 10
  open 3
probe udp UDP
  interval 5
  passdetect interval 2
probe udp UDP:2222
  port 2222
  interval 5
  passdetect interval 2


parameter-map type connection 120SECOND-IDLE
  set timeout inactivity 120
  set tcp timeout half-closed 30
parameter-map type connection 1SECOND-IDLE
  set timeout inactivity 1
parameter-map type connection 2SECOND-IDLE
  set timeout inactivity 2
parameter-map type connection 60SECOND-IDLE
  set timeout inactivity 60
  set tcp timeout half-closed 30
parameter-map type http COOKIE-DELIM
  persistence-rebalance
  set secondary-cookie-delimiters @$
parameter-map type http COOKIE-INSERT-HDR-PARSE
  persistence-rebalance
  set header-maxparse-length 4000
parameter-map type ssl EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type connection HALF-CLOSE_4s
  set tcp timeout half-closed 4
parameter-map type connection HALF-CLOSE_4send
parameter-map type connection INFINITE-IDLE
  set timeout inactivity 0
  set tcp timeout half-closed 30
parameter-map type connection LLFlows
  set timeout inactivity 0
parameter-map type connection NORM_IP
  set ip tos 22
parameter-map type connection NORM_TCP
  tcp-options timestamp allow
  reserved-bits clear
  syn-data drop
  urgent-flag clear
parameter-map type http PARSE_LENGTH
  persistence-rebalance
parameter-map type http PERSIST-INSERT
  header modify per-request
parameter-map type http PERSIST-REBAL-4K
  persistence-rebalance
parameter-map type http PERSIST-REBALANCE
  persistence-rebalance
parameter-map type connection PRED-CONNS-UDP_CONN
  set timeout inactivity 300
parameter-map type ssl RC4_128_MD5_CIPHER
  cipher RSA_WITH_RC4_128_MD5
  version TLS1
parameter-map type http REUSE-REBAL
  server-conn reuse
  persistence-rebalance
parameter-map type ssl STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type connection T1
parameter-map type ssl TERM_SSL
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
parameter-map type ssl WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type connection idle-timeout
  set timeout inactivity 480
parameter-map type connection wan-opt
  set tcp wan-optimization rtt 0

rserver host BRG-11
  ip address 192.168.120.11
  inservice
rserver host BRG-12
  ip address 192.168.120.12
  inservice
rserver host BRG-13
  ip address 192.168.120.13
  inservice
rserver host BRG-14
  ip address 192.168.120.14
  inservice
```

```
rserver host BRG-15
  ip address 192.168.120.15
  inservice
rserver host LOCAL-239-FRAGRTR
  ip address 172.29.0.239
  inservice
rserver host LOCAL-240
  ip address 172.29.0.240
  inservice
rserver host LOCAL-241
  ip address 172.29.0.241
  inservice
rserver host LOCAL-242
  ip address 172.29.0.242
  inservice
rserver host LOCAL-243
  ip address 172.29.0.243
  inservice
rserver host LOCAL-244
  ip address 172.29.0.244
  inservice
rserver host LOCAL-245
  ip address 172.29.0.245
  inservice
rserver redirect REDIRECT-100K
  webhost-redirection http://192.168.120.132/redirect-100k.html 302
  inservice
rserver redirect REDIRECT-10K
  webhost-redirection http://192.168.120.133/redirect-10k.html 302
  inservice
rserver redirect REDIRECT-1K
  webhost-redirection http://192.168.120.134/redirect-1k.html 302
  inservice
rserver host RT-151
  ip address 172.28.0.151
  inservice
rserver host RT-152
  ip address 172.28.0.152
  inservice
rserver host RT-153
  ip address 172.28.0.153
  inservice
rserver host RT-154
  ip address 172.28.0.154
  inservice
rserver host WEIGHT-80
  ip address 10.1.0.235
  weight 80
  inservice

ssl-proxy service ACE_TERM
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options TERM_SSL

serverfarm host ADD-REM-SRV
  predictor leastconns
  probe TCP
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-153
    inservice
  rserver RT-154
    inservice
serverfarm host COOKIE
  probe ICMP
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host COOKIE-HASH
  predictor leastconns
  rserver LOCAL-240
    inservice
  rserver LOCAL-240 90
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-241 90
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-242 90
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-243 90
    inservice
serverfarm host COOKIE-INSERT
  rserver BRG-14
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-152
    inservice
serverfarm host COOKIE-INSERT2
  probe HTTP
  rserver BRG-14
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-152
    inservice
serverfarm host COOKIE1
  probe HTTP
  rserver BRG-12
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-154
    inservice
serverfarm host COOKIE2
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-152
    inservice
serverfarm host CS-COOKIES
```

```
  probe TCP
  rserver BRG-12
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-154
    inservice
serverfarm host CS-MOZILLA
  rserver LOCAL-240
    inservice
  rserver RT-151
    inservice
serverfarm host CS-MSIE
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
serverfarm host DEFAULT
  probe ICMP
  rserver BRG-15
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-153
    inservice
serverfarm host FA-PURGE
  failaction purge
  rserver BRG-11
    inservice
  rserver BRG-14
    probe FA-PURGE-ICMP
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-244
    probe FA-PURGE-ICMP
    inservice
  rserver RT-151
    inservice
  rserver RT-154
    probe FA-PURGE-ICMP
    inservice
serverfarm host FTP
  probe FTP
  rserver BRG-11 21
    inservice
  rserver BRG-12 21
    inservice
  rserver LOCAL-240 21
    inservice
  rserver LOCAL-241 21
    inservice
  rserver RT-151 21
    inservice
  rserver RT-152 21
    inservice
serverfarm host GEN-443
  probe SSL
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-80
  predictor leastconns
  probe TCP
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-FTP
  probe FTP
  rserver BRG-13
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-UDP
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host GEN2-80
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-153
    inservice
serverfarm host HDR-IXIA
  rserver BRG-14
    inservice
  rserver LOCAL-241
    inservice
serverfarm host HEADER
  probe HTTP
  rserver LOCAL-240
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
serverfarm host HEADER-INSERT
  rserver BRG-13 80
    inservice
  rserver RT-151
    inservice
  rserver RT-151 80
```

```
    inservice
  rserver RT-153
    inservice
  rserver RT-153 80
    inservice
serverfarm host HEADER-INSERT2
  rserver BRG-12 80
    inservice
  rserver BRG-14 80
    inservice
  rserver LOCAL-240 80
    inservice
  rserver LOCAL-241 80
    inservice
  rserver LOCAL-244 80
    inservice
  rserver LOCAL-245 80
    inservice
  rserver RT-153 80
    inservice
  rserver RT-154 80
    inservice
serverfarm host ICMP
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
serverfarm host ICMP2
  rserver BRG-11 7777
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242 7777
    inservice
  rserver RT-152
    inservice
  rserver RT-153 7777
    inservice
serverfarm host IDLE-TCP
  probe TCP
  rserver BRG-15
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-154
    inservice
serverfarm host IDLE-UDP
  probe UDP:2222
  rserver BRG-15
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-154
    inservice
serverfarm host L3
  rserver LOCAL-244
    inservice
```

```
serverfarm host LDAP
  probe SCRIPT_LDAP
  rserver BRG-15
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host LENGTHS
  rserver LOCAL-241
    inservice
  rserver LOCAL-244
    inservice
serverfarm host LENGTHS-2
  rserver LOCAL-240
    inservice
  rserver LOCAL-245
    inservice
serverfarm host MAX-CONN
  probe HTTP
  rserver RT-151
    conn-limit max 4 min 2
    inservice
serverfarm host MAX-CONN2
  probe HTTP
  rserver LOCAL-243
    conn-limit max 500 min 2
    inservice
  rserver RT-151
    conn-limit max 500 min 2
    inservice
  rserver RT-151 90
    conn-limit max 500 min 2
    inservice
  rserver RT-151 91
    conn-limit max 500 min 2
    inservice
  rserver RT-151 92
    conn-limit max 500 min 2
    inservice
  rserver RT-151 93
    conn-limit max 500 min 2
    inservice
  rserver RT-151 94
    conn-limit max 500 min 2
    inservice
  rserver RT-151 95
    conn-limit max 500 min 2
    inservice
  rserver RT-154
    inservice
serverfarm host NORM
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-153
    inservice
serverfarm host NORM2_L7
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  retcode 100 599 check count
  rserver BRG-11 80
    inservice
  rserver LOCAL-241 80
    inservice
  rserver RT-153 80
    inservice
serverfarm host PERSISTENT
  rserver LOCAL-240
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver RT-154
    inservice
serverfarm host PRED-CONNS
  predictor leastconns
  rserver BRG-11
    inservice
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver BRG-15
    inservice
  rserver LOCAL-240
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
  rserver RT-154
    inservice
serverfarm host PRED-CONNS-UDP
  failaction purge
  predictor leastconns
  rserver BRG-11 2222
    inservice
  rserver LOCAL-240 2222
    inservice
  rserver LOCAL-242 2222
    inservice
  rserver LOCAL-244 2222
    probe PRED-PING
    inservice
  rserver RT-151 2222
    inservice
  rserver RT-153 2222
    inservice
  rserver RT-154 2222
```

```
    inservice
serverfarm host PREDICTOR
  predictor leastconns
  probe TCP
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
serverfarm host PROBES
  predictor leastconns
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-154
    inservice
serverfarm host PROBES-2
  predictor leastconns
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
  rserver RT-154
    inservice
serverfarm host PROBES-MANY
  predictor leastconns
  probe SCRIPT_TFTP
  rserver BRG-11
    probe RADIUS
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    probe RADIUS
    inservice
  rserver RT-152
    inservice
  rserver RT-154
    probe RADIUS
    inservice
serverfarm host RADIUS
  probe RADIUS
  rserver BRG-11
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host RED-ALL-SVRS
  rserver LOCAL-240
```

```
    inservice
  rserver LOCAL-241
    inservice
serverfarm host REDIRECT
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
serverfarm redirect REDIRECT-100K
  rserver REDIRECT-100K
    inservice
serverfarm redirect REDIRECT-10K
  rserver REDIRECT-10K
    inservice
serverfarm redirect REDIRECT-1K
  rserver REDIRECT-1K
    inservice
serverfarm host RHI
  rserver BRG-11
    conn-limit max 4 min 2
    inservice
  rserver LOCAL-241
    conn-limit max 4 min 2
    inservice
  rserver RT-154
    conn-limit max 4 min 2
    inservice
serverfarm host SORRY
  rserver LOCAL-240
    inservice
serverfarm host SORRY-BACK
  rserver LOCAL-243
    inservice
  rserver RT-151
    inservice
serverfarm host STICKY-COOKIE
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver RT-154
    inservice
serverfarm host STICKY-HEADER
  probe HTTP
  rserver BRG-12
    inservice
  rserver LOCAL-243
    inservice
  rserver RT-151
    inservice
serverfarm host STICKY-HEADER2
  probe HTTP
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
```

```
    inservice
serverfarm host STICKY-NETMASK
  probe ICMP
  rserver BRG-12
    inservice
  rserver LOCAL-242
    inservice
  rserver RT-153
    inservice
serverfarm host TCP-REUSE
  rserver BRG-15
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-154
    inservice
serverfarm host UDP
  probe UDP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-151
    inservice
serverfarm host URL-MAP-128K
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
serverfarm host URL-MAP-16K
  rserver BRG-12
    inservice
  rserver LOCAL-242
    inservice
serverfarm host URL-MAP-32K
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
serverfarm host URL-MAP-512K
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
serverfarm host URL-MAP-64K
  rserver LOCAL-242 91
    inservice
  rserver LOCAL-244
    inservice
serverfarm host URL-MAPS
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-244
    inservice
serverfarm host WEIGHT
  probe HTTP
  rserver BRG-11
    weight 10
    inservice
  rserver LOCAL-240
    weight 20
    inservice
  rserver LOCAL-243
    weight 30
    inservice
  rserver RT-152
    weight 40
    inservice
serverfarm host fUDP

sticky ip-netmask 255.255.255.255 address source STKY-GRP-30
  timeout 40
  replicate sticky
  serverfarm GEN-80
sticky http-cookie cookie-gold-grp40 STKY-GRP-40
  cookie insert browser-expire
  timeout 1
  replicate sticky
  serverfarm GEN2-80
sticky ip-netmask 255.255.255.255 address both STKY-GRP-31
  timeout 40
  replicate sticky
  serverfarm GEN-UDP
sticky ip-netmask 255.255.255.255 address both STKY-GRP-32
  timeout 40
  replicate sticky
  serverfarm GEN-FTP
sticky http-cookie COOKIE_TEST COOKIE-GROUP
  cookie secondary URLCOOKIE
  timeout 40
  replicate sticky
  serverfarm STICKY-COOKIE
  8 static cookie-value "REDSOX0" rserver RT-154
  16 static cookie-value "PATRIOTS0" rserver RT-151
sticky http-header MSISDN HEADER-GROUP-42
  timeout 30
  replicate sticky
  serverfarm STICKY-HEADER
sticky http-header TestHeader HEADER-GROUP-41
  header offset 15 length 7
  timeout 30
  replicate sticky
  serverfarm STICKY-HEADER2
sticky http-cookie COOKIE_INSERT COOKIE-INSERT-GROUP-45
  cookie insert
  timeout 1
  replicate sticky
  serverfarm COOKIE-HASH
sticky http-cookie COOKIE_TEST COOKIE-MAP-GROUP
  replicate sticky
  serverfarm CS-COOKIES
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-29
  timeout 30
  replicate sticky
  serverfarm MAX-CONN
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-11
  timeout 30
  replicate sticky
  serverfarm URL-MAP-16K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-13
  timeout 30
  replicate sticky
  serverfarm URL-MAP-64K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-12
  timeout 30
  replicate sticky
```

```
    serverfarm URL-MAP-32K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-14
    timeout 30
    replicate sticky
    serverfarm URL-MAP-128K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-15
    timeout 30
    replicate sticky
    serverfarm URL-MAP-512K
  sticky http-cookie Safeharbor-Cookie1 COOKIE-GROUP-42
    cookie insert
    timeout 30
    replicate sticky
  sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes 
  COOKIE-INSERT-GROUP-46
    cookie insert
    timeout 1
    replicate sticky
    serverfarm COOKIE-INSERT
  sticky http-cookie COOKIE_TEST STKY-GRP-43
    cookie offset 1 length 999
    timeout 30
    replicate sticky
    serverfarm PREDICTOR
  sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes 
  COOKIE-INS2-GROUP-46
    cookie insert
    timeout 1
    serverfarm COOKIE-INSERT
  sticky ip-netmask 255.255.255.255 address source STKY-GRP-50
    timeout 20
    serverfarm SORRY backup SORRY-BACK
  sticky ip-netmask 255.255.255.255 address both STKY-GRP-33
    timeout 20
    replicate sticky
    serverfarm STICKY-NETMASK

  class-map type http loadbalance match-all 128K-FORWARDING
    2 match http url .*128k.*
  class-map type http loadbalance match-all 16K-FORWARDING
    2 match http url .*16k.*
  class-map type http loadbalance match-all 32K-FORWARDING
    2 match http url .*32k.*
  class-map type http loadbalance match-all 512K-FORWARDING
    2 match http url .*512k.*
  class-map type http loadbalance match-all 64K-FORWARDING
    2 match http url .*64k.*
  class-map match-all ADD-REM-SRV-VIP_110:80
    2 match virtual-address 192.168.120.110 tcp eq www
  class-map type http loadbalance match-all BROWSER_FIREFOX
    2 match http header User-Agent header-value ".*Firefox.*"
  class-map type http loadbalance match-all BROWSER_MOZILLA
    2 match http header User-Agent header-value ".*Mozilla.*"
  class-map type http loadbalance match-all BROWSER_MOZILLA40
    2 match http header User-Agent header-value ".*Mozilla/4.0.*"
  class-map type http loadbalance match-all BROWSER_MOZILLA50
    2 match http header User-Agent header-value ".*Mozilla/5.0.*"
  class-map type http loadbalance match-all BROWSER_MSIE
    2 match http header User-Agent header-value ".*MSIE.*"
  class-map match-all COOKIE-HASH-VIP_10.20.30.40:80
    2 match virtual-address 10.20.30.40 tcp eq www
  class-map match-all COOKIE-INS2-VIP_118:8888
    2 match virtual-address 192.168.120.118 tcp eq 8888
  class-map match-all COOKIE-INSERT-VIP_118:80
  2 match virtual-address 192.168.120.118 tcp eq www
class-map match-all COOKIE-MAP-VIP_124:80
  2 match virtual-address 192.168.120.124 tcp eq www
class-map type http loadbalance match-all COOKIE-MAP:80
  2 match http cookie COOKIE_TEST cookie-value "This is a test0"
class-map match-all FA-PURGE-VIP_113:ANY
  2 match virtual-address 192.168.120.113 any
class-map type ftp inspect match-any FTP-L7-MAX-DENY
  2 match request-method appe
  3 match request-method cdup
  4 match request-method get
  5 match request-method help
  6 match request-method mkd
  7 match request-method rmd
  8 match request-method rnfr
  9 match request-method rnto
  10 match request-method site
  11 match request-method stou
  12 match request-method cwd
class-map type ftp inspect match-any FTP-L7-MAX-DENY2
  2 match request-method syst
class-map type ftp inspect match-any FTP-L7-MIN-DENY
  2 match request-method mkd
  3 match request-method rmd
class-map match-all FTP-VIP-NAT_119
  2 match destination-address 192.168.120.119 255.255.255.255
class-map match-all FTP-VIP_119:1111
  2 match virtual-address 192.168.120.119 tcp eq 1111
class-map match-all FTP-VIP_119:3333-4444
  2 match virtual-address 192.168.120.119 tcp range 3333 4444
class-map match-all GEN-NAT_120
  2 match virtual-address 192.168.120.120 any
class-map match-all GEN-VIP_120:21
  2 match virtual-address 192.168.120.120 tcp eq ftp
class-map match-all GEN-VIP_120:443
  2 match virtual-address 192.168.120.120 tcp eq https
class-map match-all GEN-VIP_120:80
  2 match virtual-address 192.168.120.120 tcp eq www
class-map match-all GEN-VIP_120:UDP
  2 match virtual-address 192.168.120.120 udp any
class-map match-all HDR-IXIA-VIP_123:80
  2 match virtual-address 192.168.120.123 tcp eq www
class-map match-all HEADER-INSERT-VIP_121:443
  2 match virtual-address 192.168.120.121 tcp eq https
class-map match-all HEADER-INSERT-VIP_121:80
  3 match virtual-address 192.168.120.121 tcp eq www
class-map match-all HEADER-INSERT2-VIP_122:443
  2 match virtual-address 192.168.120.122 tcp eq https
class-map match-all HEADER-INSERT2-VIP_122:80
  2 match virtual-address 192.168.120.122 tcp eq www
class-map match-all HEADER-VIP_125:80
  2 match virtual-address 192.168.120.125 tcp eq www
class-map match-all ICMP
  2 match access-list ICMP-ONLY
class-map match-all ICMP-UDP-VIP_138
  3 match virtual-address 192.168.120.138 udp eq 2222
class-map match-all ICMP-URL-VIP_138:80
  2 match virtual-address 192.168.120.138 tcp eq www
class-map match-all IDLE-VIP_111:TCP
  2 match virtual-address 192.168.120.111 tcp any
class-map match-all IDLE-VIP_111:UDP
  2 match virtual-address 192.168.120.111 udp any
class-map type http loadbalance match-all INDEX.HTML
  2 match http url /index.html
```

```
class-map match-all L3_139
  2 match virtual-address 192.168.120.139 any
class-map match-all LENGTHS-VIP_136:80
  2 match virtual-address 192.168.120.136 tcp eq www
class-map match-all MAX-CONN-VIP_105
  2 match virtual-address 192.168.120.105 any
class-map match-all MAX-CONN-VIP_126:80
  2 match virtual-address 192.168.120.126 tcp eq www
class-map type management match-any MGT
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
  8 match protocol kalap-udp any
class-map type http loadbalance match-all MSISDN
  2 match http header MSISDN header-value ".*"
class-map match-any NAT_CLASS
  2 match access-list NAT_ACCESS
class-map match-any NORM-VIP_142:80
  2 match virtual-address 192.168.120.142 tcp eq www
class-map match-any NORM2_L7-VIP_142:888
  2 match virtual-address 192.168.120.142 tcp eq 888
class-map match-all NORM_ALL_TRAFFIC-VIP_ANY
  2 match any
class-map type http loadbalance match-any P-COOKIE-INS
  2 match http url /index.html* method GET
class-map type http loadbalance match-any P-COOKIE-INS2
  2 match http url .*
class-map type http loadbalance match-all P-HDR-INSERT
  2 match http url .*
class-map type http loadbalance match-all P-HDR-IXIA
  2 match http url .*
class-map type http loadbalance match-all P-HDR-SRCDST-IP
  2 match http url .*
class-map match-all PERSISTENT-VIP_131:80
  2 match virtual-address 192.168.120.131 tcp eq www
class-map match-all PRED-CONNS-UDP-VIP_128:2222
  2 match virtual-address 192.168.120.128 udp any
class-map match-all PRED-CONNS-VIP_128:80
  2 match virtual-address 192.168.120.128 tcp eq www
class-map match-all PREDICTOR_117:80
  2 match virtual-address 192.168.120.117 tcp eq www
class-map match-all RADIUS-NAT_134
  2 match virtual-address 192.168.120.134 udp eq radius-auth
class-map match-all RED-100K-VIP_132:80
  2 match virtual-address 192.168.120.132 tcp eq www
class-map match-all RED-10K-VIP_133:80
  2 match virtual-address 192.168.120.133 tcp eq www
class-map match-all RED-1K-VIP_134:80
  2 match virtual-address 192.168.120.134 tcp eq www
class-map type http loadbalance match-all REDIRECT-100K
  2 match http url .*redirect-100k.html
class-map type http loadbalance match-all REDIRECT-10K
  2 match http url .*redirect-10k.html
class-map type http loadbalance match-all REDIRECT-1K
  2 match http url .*redirect-1k.html
class-map match-all REDIRECT-VIP_135:80
  2 match virtual-address 192.168.120.135 tcp eq www
class-map match-all REDIRECT_135:80
  2 match virtual-address 192.168.120.135 tcp eq www
class-map match-all RHI_125.100:80
```

```
  2 match virtual-address 192.168.125.100 tcp eq www
class-map match-all RHI_125.101:80
  2 match virtual-address 192.168.125.101 tcp eq www
class-map match-all RHI_125.102:80
  2 match virtual-address 192.168.125.102 tcp eq www
class-map match-all RHI_125.103:80
  2 match virtual-address 192.168.125.103 tcp eq www
class-map match-all RHI_125.104:80
  2 match virtual-address 192.168.125.104 tcp eq www
class-map match-all RHI_125.105:80
  2 match virtual-address 192.168.125.105 tcp eq www
class-map match-all RHI_125.106:80
  2 match virtual-address 192.168.125.106 tcp eq www
class-map match-all RHI_125.107:80
  2 match virtual-address 192.168.125.107 tcp eq www
class-map match-all RHI_125.108:80
  2 match virtual-address 192.168.125.108 tcp eq www
class-map match-all RHI_125.109:80
  2 match virtual-address 192.168.125.109 tcp eq www
class-map match-all RHI_125.110:80
  2 match virtual-address 192.168.125.110 tcp eq www
class-map match-all RHI_125.111:80
  2 match virtual-address 192.168.125.111 tcp eq www
class-map match-all RHI_125.112:80
  2 match virtual-address 192.168.125.112 tcp eq www
class-map match-all RHI_125.113:80
  2 match virtual-address 192.168.125.113 tcp eq www
class-map match-all RHI_125.114:80
  2 match virtual-address 192.168.125.114 tcp eq www
class-map match-all RHI_125.115:80
  2 match virtual-address 192.168.125.115 tcp eq www
class-map match-all RHI_125.116:80
  2 match virtual-address 192.168.125.116 tcp eq www
class-map match-all RHI_125.117:80
  2 match virtual-address 192.168.125.117 tcp eq www
class-map match-all RHI_125.118:80
  2 match virtual-address 192.168.125.118 tcp eq www
class-map match-all RHI_125.119:80
  2 match virtual-address 192.168.125.119 tcp eq www
class-map match-all RHI_125.120-127
  2 match virtual-address 192.168.125.127 255.255.255.248 any
class-map match-all SORRY-VIP_137:80
  2 match virtual-address 192.168.120.137 tcp eq www
class-map match-all STICKY-COOKIE-VIP_127:80
  2 match virtual-address 192.168.120.127 tcp eq www
class-map match-all STICKY-HEADER_129:80
  2 match virtual-address 192.168.120.129 tcp eq www
class-map match-all STICKY-IP_115:ANY
  2 match virtual-address 192.168.120.115 any
class-map match-any TCP-REUSE-VIP_141:80
  10 match virtual-address 192.168.120.141 tcp eq www
class-map match-any TCP-REUSE-VIP_141:81
  10 match virtual-address 192.168.120.141 tcp eq 81
class-map type http loadbalance match-all TestHeader
  2 match http header TestHeader header-value ".*"
class-map match-all UDP-VIP_114:UDP
  2 match virtual-address 192.168.120.114 udp any
class-map match-all UDP-VIP_114:UDP-53
  2 match virtual-address 192.168.120.114 udp eq domain
class-map type http loadbalance match-all URL*_L7
  2 match http url .*
class-map match-all URL-MAPS-VIP_130:80
  2 match virtual-address 192.168.120.130 tcp eq www
class-map type http loadbalance match-all URLCOOKIE-MAP1
```

```
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE1"
class-map type http loadbalance match-all URLCOOKIE-MAP2
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE2"
class-map match-all WEIGHT_112:80
  2 match virtual-address 192.168.120.112 tcp eq www

policy-map type management first-match P-MGT
  class MGT
    permit

policy-map type loadbalance first-match FTP-LB-SF_FTP
  class class-default
    serverfarm FTP
policy-map type loadbalance first-match MAX-CONN-LB-SF_MAX-CONN2
  class INDEX.HTML
    serverfarm MAX-CONN
  class URL*_L7
    serverfarm MAX-CONN2
policy-map type loadbalance first-match PLBSF-FTP-test
  class class-default
policy-map type loadbalance first-match PLBSF_ADD-REM-SRV
  class class-default
    serverfarm ADD-REM-SRV
policy-map type loadbalance first-match PLBSF_COOKIE-HASH
  class class-default
    sticky-serverfarm COOKIE-INSERT-GROUP-45
policy-map type loadbalance first-match PLBSF_COOKIE-INS2
  class class-default
    sticky-serverfarm COOKIE-INS2-GROUP-46
policy-map type loadbalance first-match PLBSF_COOKIE-INSERT
  class P-COOKIE-INS
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Source-IP header-value "%is"
    insert-http Destination_IP header-value "%id"
  class P-COOKIE-INS2
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Source-IP header-value "%is"
    insert-http Destination_IP header-value "%id"
policy-map type loadbalance first-match PLBSF_COOKIE-MAP
  class URLCOOKIE-MAP2
    serverfarm COOKIE2
  class URLCOOKIE-MAP1
    serverfarm COOKIE1
  class COOKIE-MAP:80
    serverfarm COOKIE
  class class-default
    serverfarm GEN-80
policy-map type loadbalance first-match PLBSF_FA-PURGE
  class class-default
    serverfarm FA-PURGE
policy-map type loadbalance first-match PLBSF_GEN-443
  class class-default
    serverfarm GEN-443
policy-map type loadbalance first-match PLBSF_GEN-80
  class INDEX.HTML
    sticky-serverfarm STKY-GRP-40
  class class-default
    sticky-serverfarm STKY-GRP-30
policy-map type loadbalance first-match PLBSF_GEN-FTP
  class class-default
    sticky-serverfarm STKY-GRP-32
policy-map type loadbalance first-match PLBSF_GEN-UDP
  class class-default
    sticky-serverfarm STKY-GRP-31
```

```
policy-map type loadbalance first-match PLBSF_HDR-IXIA
  class P-HDR-IXIA
    serverfarm HDR-IXIA
policy-map type loadbalance first-match PLBSF_HEADER
  class BROWSER_FIREFOX
    serverfarm CS-MOZILLA
  class BROWSER_MOZILLA40
    serverfarm CS-MSIE
  class BROWSER_MOZILLA50
    serverfarm CS-MOZILLA
  class BROWSER_MSIE
    serverfarm CS-MSIE
  class BROWSER_MOZILLA
    serverfarm CS-MOZILLA
  class class-default
    serverfarm HEADER
policy-map type loadbalance first-match PLBSF_HEADER-INSERT
  class P-HDR-INSERT
    serverfarm HEADER-INSERT
    insert-http
Custom-header_name_size_100bytes_abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQR
S_100BYTES header-value "Size of inserted header value is 100 bytes
abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHI_100BYTES"
    insert-http Destination_iP header-value "%id"
    insert-http Pragma header-value "Pragma no Pragma that is the question"
    insert-http Accept header-value "anything"
    insert-http Source-IP header-value "%is"
policy-map type loadbalance first-match PLBSF_HEADER-INSERT2
  class P-HDR-SRCDST-IP
    serverfarm HEADER-INSERT2
    insert-http Destination_iP header-value "%id"
    insert-http Source-IP header-value "%is"
policy-map type loadbalance first-match PLBSF_ICMP
  class INDEX.HTML
    serverfarm ICMP
policy-map type loadbalance first-match PLBSF_ICMP2
  class class-default
    serverfarm ICMP2
policy-map type loadbalance first-match PLBSF_IDLE-TCP
  class class-default
    serverfarm IDLE-TCP
policy-map type loadbalance first-match PLBSF_IDLE-UDP
  class class-default
    serverfarm IDLE-UDP
policy-map type loadbalance first-match PLBSF_L3
  class class-default
    serverfarm L3
policy-map type loadbalance first-match PLBSF_LENGTHS
  class INDEX.HTML
    serverfarm LENGTHS
  class class-default
    serverfarm LENGTHS-2
policy-map type loadbalance first-match PLBSF_MAX-CONN
  class INDEX.HTML
    sticky-serverfarm STICKY-GROUP-29
  class class-default
    serverfarm MAX-CONN2
policy-map type loadbalance first-match PLBSF_NORM
  class class-default
    serverfarm NORM
policy-map type loadbalance first-match PLBSF_NORM2_L7
  class class-default
    serverfarm NORM2_L7
policy-map type loadbalance first-match PLBSF_PERSISTENT
```

```
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm PERSISTENT
policy-map type loadbalance first-match PLBSF_PRED-CONNS
  class class-default
    serverfarm PRED-CONNS
policy-map type loadbalance first-match PLBSF_PRED-CONNS-UDP
  class class-default
    serverfarm PRED-CONNS-UDP
policy-map type loadbalance first-match PLBSF_PREDICTOR
  class class-default
    serverfarm PREDICTOR
policy-map type loadbalance first-match PLBSF_RADIUS-1812
  class class-default
    serverfarm RADIUS
policy-map type loadbalance first-match PLBSF_RED-ALL-SVRS
  class class-default
    serverfarm RED-ALL-SVRS
policy-map type loadbalance first-match PLBSF_REDIRECT
  class REDIRECT-1K
    serverfarm REDIRECT-1K
  class REDIRECT-10K
    serverfarm REDIRECT-10K
  class REDIRECT-100K
    serverfarm REDIRECT-100K
  class class-default
    serverfarm REDIRECT
policy-map type loadbalance first-match PLBSF_RHI
  class URL*_L7
    serverfarm RHI
policy-map type loadbalance first-match PLBSF_SORRY
  class class-default
    sticky-serverfarm STKY-GRP-50
policy-map type loadbalance first-match PLBSF_STICKY-COOKIE
  class INDEX.HTML
    sticky-serverfarm COOKIE-GROUP
  class URL*_L7
    serverfarm GEN-80
policy-map type loadbalance first-match PLBSF_STICKY-HEADER
  class MSISDN
    sticky-serverfarm HEADER-GROUP-42
  class TestHeader
    sticky-serverfarm HEADER-GROUP-41
  class class-default
    serverfarm DEFAULT
policy-map type loadbalance first-match PLBSF_STICKY-NETMASK
  class class-default
    sticky-serverfarm STKY-GRP-33
policy-map type loadbalance first-match PLBSF_TCP-REUSE
  class URL*_L7
    serverfarm TCP-REUSE
policy-map type loadbalance first-match PLBSF_UDP
  class class-default
    serverfarm UDP
policy-map type loadbalance first-match PLBSF_URL-MAPS
```

```
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm URL-MAPS
policy-map type loadbalance first-match PLBSF_WEIGHT
  class class-default
    serverfarm WEIGHT

policy-map type inspect ftp first-match FTP-INSPSF_FTP
  class FTP-L7-MAX-DENY2
    mask-reply
  class FTP-L7-MAX-DENY
    deny

policy-map multi-match NAT_POLICY
  class NAT_CLASS
    nat dynamic 1 vlan 120
policy-map multi-match NORMALIZATION
  class NORM_ALL_TRAFFIC-VIP_ANY
    connection advanced-options NORM_IP
policy-map multi-match SH-Gold-VIPs
  class ADD-REM-SRV-VIP_110:80
    loadbalance vip inservice
    loadbalance policy PLBSF_ADD-REM-SRV
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
  class WEIGHT_112:80
    loadbalance vip inservice
    loadbalance policy PLBSF_WEIGHT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class FA-PURGE-VIP_113:ANY
    loadbalance vip inservice
    loadbalance policy PLBSF_FA-PURGE
    nat dynamic 1 vlan 120
    connection advanced-options INFINITE-IDLE
  class UDP-VIP_114:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_UDP
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    connection advanced-options 1SECOND-IDLE
  class ICMP-UDP-VIP_138
    loadbalance vip inservice
    loadbalance policy PLBSF_ICMP2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options 60SECOND-IDLE
  class COOKIE-HASH-VIP_10.20.30.40:80
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-HASH
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class URL-MAPS-VIP_130:80
    loadbalance vip inservice
    loadbalance policy PLBSF_URL-MAPS
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PERSISTENT-VIP_131:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PERSISTENT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class REDIRECT-VIP_135:80
    loadbalance vip inservice
    loadbalance policy PLBSF_REDIRECT
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class LENGTHS-VIP_136:80
    loadbalance vip inservice
    loadbalance policy PLBSF_LENGTHS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PARSE_LENGTH
  class NORM-VIP_142:80
    loadbalance vip inservice
    loadbalance policy PLBSF_NORM
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class FTP-VIP-NAT_119
    nat dynamic 1 vlan 120
    nat dynamic 1 vlan 29
  class FTP-VIP_119:1111
    loadbalance vip inservice
    loadbalance policy FTP-LB-SF_FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class MAX-CONN-VIP_105
    loadbalance vip inservice
    loadbalance policy MAX-CONN-LB-SF_MAX-CONN2
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class FTP-VIP_119:3333-4444
    loadbalance vip inservice
    loadbalance policy FTP-LB-SF_FTP
    loadbalance vip icmp-reply active
    inspect ftp strict
  class PREDICTOR_117:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PREDICTOR
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class GEN-VIP_120:80
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-80
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class GEN-VIP_120:443
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-443
    loadbalance vip icmp-reply active
  class GEN-VIP_120:21
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class GEN-VIP_120:UDP
```

```
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-UDP
    loadbalance vip icmp-reply active
    connection advanced-options 2SECOND-IDLE
  class IDLE-VIP_111:TCP
    loadbalance vip inservice
    loadbalance policy PLBSF_IDLE-TCP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options 60SECOND-IDLE
  class IDLE-VIP_111:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_IDLE-UDP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options 60SECOND-IDLE
  class STICKY-IP_115:ANY
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-NETMASK
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
  class STICKY-COOKIE-VIP_127:80
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-COOKIE
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-DELIM
  class STICKY-HEADER_129:80
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-HEADER
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class TCP-REUSE-VIP_141:81
    loadbalance vip inservice
    loadbalance policy PLBSF_TCP-REUSE
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options REUSE-REBAL
  class ICMP-URL-VIP_138:80
    loadbalance vip inservice
    loadbalance policy PLBSF_ICMP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class ICMP
    inspect icmp error
  class HEADER-INSERT-VIP_121:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class HDR-IXIA-VIP_123:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HDR-IXIA
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class HEADER-INSERT2-VIP_122:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class COOKIE-INSERT-VIP_118:80
```

```
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-INSERT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-INSERT-HDR-PARSE
  class COOKIE-MAP-VIP_124:80
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-MAP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-DELIM
  class HEADER-VIP_125:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER
    loadbalance vip icmp-reply
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PRED-CONNS-VIP_128:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PRED-CONNS
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PRED-CONNS-UDP-VIP_128:2222
    loadbalance vip inservice
    loadbalance policy PLBSF_PRED-CONNS-UDP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
    connection advanced-options PRED-CONNS-UDP_CONN
  class MAX-CONN-VIP_126:80
    loadbalance vip inservice
    loadbalance policy PLBSF_MAX-CONN
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options HALF-CLOSE_4s
  class RED-100K-VIP_132:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RED-10K-VIP_133:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RED-1K-VIP_134:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class SORRY-VIP_137:80
    loadbalance vip inservice
    loadbalance policy PLBSF_SORRY
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class TCP-REUSE-VIP_141:80
    loadbalance vip inservice
    loadbalance policy PLBSF_TCP-REUSE
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options REUSE-REBAL
  class NORM2_L7-VIP_142:888
    loadbalance vip inservice
    loadbalance policy PLBSF_NORM2_L7
```

```
      loadbalance vip icmp-reply active
      nat dynamic 1 vlan 120
    class COOKIE-INS2-VIP_118:8888
      loadbalance vip inservice
      loadbalance policy PLBSF_COOKIE-INS2
      loadbalance vip icmp-reply active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options COOKIE-INSERT-HDR-PARSE
    class RADIUS-NAT_134
      loadbalance vip inservice
      loadbalance policy PLBSF_RADIUS-1812
      loadbalance vip icmp-reply
      nat dynamic 1 vlan 120
      nat dynamic 1 vlan 29
    class UDP-VIP_114:UDP-53
      loadbalance vip inservice
      loadbalance policy PLBSF_UDP
      loadbalance vip icmp-reply
      nat dynamic 1 vlan 120
      inspect dns
      connection advanced-options 1SECOND-IDLE
    class L3_139
      loadbalance vip inservice
      loadbalance policy PLBSF_L3
      loadbalance vip icmp-reply active
      nat dynamic 1 vlan 120
    class GEN-NAT_120
      nat dynamic 1 vlan 120
    class HEADER-INSERT-VIP_121:443
      loadbalance vip inservice
      loadbalance policy PLBSF_HEADER-INSERT
      loadbalance vip icmp-reply active
      nat dynamic 1 vlan 120
      nat dynamic 1 vlan 29
      appl-parameter http advanced-options PERSIST-REBAL-4K
      ssl-proxy server ACE_TERM
    class HEADER-INSERT2-VIP_122:443
      loadbalance vip inservice
      loadbalance policy PLBSF_HEADER-INSERT2
      loadbalance vip icmp-reply active
      nat dynamic 1 vlan 120
      nat dynamic 1 vlan 29
      appl-parameter http advanced-options PERSIST-REBAL-4K
      ssl-proxy server ACE_TERM
  policy-map multi-match SH-Gold-VIPs3
    class RHI_125.100:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.101:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.102:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.103:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.104:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.105:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.106:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.107:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.108:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.109:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.110:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.111:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.112:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.113:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.114:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.115:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.116:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.117:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.118:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.119:80
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 120
      appl-parameter http advanced-options PERSIST-REBALANCE
    class RHI_125.120-127
      loadbalance vip inservice
      loadbalance policy PLBSF_RHI
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      loadbalance vip advertise metric 254
      nat dynamic 1 vlan 120
```

```
    appl-parameter http advanced-options PERSIST-REBALANCE

service-policy input P-MGT

interface vlan 28
interface vlan 29
  ip address 172.29.0.2 255.255.255.0
  alias 172.29.0.1 255.255.255.0
  peer ip address 172.29.0.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.71 192.168.120.71 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  no shutdown
interface vlan 99
  ip address 192.168.99.2 255.255.255.0
  peer ip address 192.168.99.3 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 105
  ip address 192.168.105.2 255.255.255.0
  peer ip address 192.168.105.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
interface vlan 120
  description Upstream VLAN_120 - Clients and VIPs
  ip address 192.168.120.2 255.255.255.0
  alias 192.168.120.1 255.255.255.0
  peer ip address 192.168.120.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.70 192.168.120.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  service-policy input NAT_POLICY
  no shutdown

ft track host GW_251-252
  track-host 192.168.16.251
  peer track-host 192.168.16.252
  probe HA-ICMP priority 10
  peer probe HA-ICMP priority 5
  priority 110
  peer priority 5
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 110
  peer priority 5
ft track interface Int_4/37_6/13_V99
  track-interface vlan 99
  peer track-interface vlan 99
  priority 110
  peer priority 5
ft track host RT-241
  track-host 172.29.0.241
  probe HA-TCP:554 priority 50
  probe HA-TCP:1755 priority 60

domain SH-Gold-Domain
```

```
  add-object all
domain KALAP_TAG
  add-object class-map GEN-VIP_120:21

role SHAdmin
  rule 1 permit create
  rule 2 permit monitor
  rule 3 permit modify
role SHUser
  rule 1 deny create
  rule 2 permit monitor
  rule 3 deny modify
  rule 4 permit debug
role TEST

ip route 10.1.0.0 255.255.255.0 192.168.120.254
ip route 172.28.0.0 255.255.255.0 172.29.0.253
ip route 192.168.16.251 255.255.255.255 192.168.105.251
ip route 192.168.16.252 255.255.255.255 192.168.105.252
ip route 10.3.0.0 255.255.255.0 192.168.120.254
username admin password 5 $1$hU1iScF8$WmpdK4IcQI2ofTMDm6l.N1  role Admin domain
default-domain
username localadmin password 5 $1$g5rd5HO2$C34zVe3a9f73Dce/WNvbM.  role Admin domain
SH-Gold-Domain default-domain
username localuser password 5 $1$I21oqX4Q$/OqAKTdBbe8xreKwZtWR3.  role Network-Monitor
domain SH-Gold-Domain
username vrtadmgold password 5 $1$.gNJPJS6$UtozYODAuirfw8XHR1FA8/  role Admin domain
SH-Gold-Domain
username vrtnetmongold password 5 $1$InjySHhu$oLsQV267Nu68q3fH6h7Z4.  role Network-Monitor
domain SH-Gold-Domain
username vrtjohndoe password 5 $1$klNcwN8c$3hTNwFSMtned/9k2a5RPg.  role Admin domain
SH-Gold-Domain

snmp-server community ACE-public group Network-Monitor
snmp-server community ACE-private group Network-Monitor

snmp-server host 10.1.0.242 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-1_ACE2-1/SH-Gold# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:16 ---

+++ 09:57:16 6K-1_ACE2-1 ctxExec +++
changeto SH-Gold


6K-1_ACE2-1/SH-Gold#


6K-1_ACE2-1/SH-Gold# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:17 ---
```

```
+++ 09:57:17 6K-1_ACE2-1 ctxExec +++
changeto SH-LOAD


6K-1_ACE2-1/SH-LOAD#


6K-1_ACE2-1/SH-LOAD# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:17 ---

+++ 09:57:17 6K-1_ACE2-1 ctxExec +++
changeto SH-LOAD


6K-1_ACE2-1/SH-LOAD# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging history 5
logging buffered 5
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514


aaa authentication login error-enable

access-list everyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 10
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_CLEAR"
probe https GEN_HTTPS
  interval 10
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_SSL"
probe icmp ICMP
  interval 5
  passdetect interval 10


parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
```

```
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA
    cipher RSA_WITH_AES_256_CBC_SHA
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA
    version SSL3
  parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA priority 2
    cipher RSA_WITH_AES_256_CBC_SHA priority 3
  parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA priority 2
    cipher RSA_WITH_DES_CBC_SHA priority 3
  parameter-map type ssl PARM_ACE_TERM
    cipher RSA_WITH_RC4_128_MD5
    version TLS1
  parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 6
  parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA priority 2
    cipher RSA_WITH_AES_256_CBC_SHA priority 3
  parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA priority 2
    cipher RSA_WITH_DES_CBC_SHA priority 3
  parameter-map type connection TCP_PARAM
    syn-data drop
    exceed-mss allow

  rserver host BRG-IIS-1
    ip address 172.28.1.26
    inservice
  rserver host BRG-IIS-2
    ip address 172.28.1.27
    inservice
  rserver host BRG-IIS-3
    ip address 172.28.1.28
    inservice
  rserver host BRG-IIS-4
    ip address 172.28.1.29
    inservice
  rserver host BRG-IIS-5
    ip address 172.28.1.30
    inservice
  rserver host BRG-LINUX-1
    ip address 172.28.1.21
    inservice
  rserver host BRG-LINUX-2
    ip address 172.28.1.22
    inservice
  rserver host BRG-LINUX-3
    ip address 172.28.1.23
    inservice
  rserver host BRG-LINUX-4
    ip address 172.28.1.24
    inservice
  rserver host BRG-LINUX-5
    ip address 172.28.1.25
    inservice

  ssl-proxy service ACE_AS_CLIENT
    ssl advanced-options PARM_ACE_AS_CLIENT
  ssl-proxy service ACE_END_TO_END
    key pkey.pem
    cert end-to-end.pem
    ssl advanced-options PARM_ACE_TERM
  ssl-proxy service ACE_TERM
    key pkey.pem
    cert term.pem
    ssl advanced-options PARM_ACE_TERM_STRONG_CIPHERS

  serverfarm host ACE_END_TO_END_SERVERS_SSL
    description SERVERS FOR END TO END SSL TESTING
    failaction purge
    probe GEN_HTTPS
    rserver BRG-IIS-1 443
      inservice
    rserver BRG-IIS-2 443
      inservice
    rserver BRG-IIS-3 443
      inservice
    rserver BRG-LINUX-1 443
      inservice
    rserver BRG-LINUX-2 443
      inservice
    rserver BRG-LINUX-3 443
      inservice
  serverfarm host ACE_INIT_SERVERS_SSL
    description SERVERS FOR SSL INIT TESTING
    failaction purge
    probe GEN_HTTPS
    rserver BRG-IIS-1 443
      inservice
    rserver BRG-IIS-2 443
      inservice
    rserver BRG-IIS-3 443
      inservice
    rserver BRG-LINUX-1 443
      inservice
    rserver BRG-LINUX-2 443
      inservice
    rserver BRG-LINUX-3 443
      inservice
  serverfarm host ACE_TERM_SERVERS_CLEAR
    description SERVERS FOR SSL TERM TESTING
    failaction purge
    probe GEN_HTTP
    rserver BRG-IIS-1 80
      inservice
    rserver BRG-IIS-2 80
      inservice
    rserver BRG-IIS-3 80
      inservice
    rserver BRG-LINUX-1 80
      inservice
    rserver BRG-LINUX-2 80
      inservice
    rserver BRG-LINUX-3 80
      inservice
  serverfarm host L3
    probe GEN_HTTP
    probe ICMP
    rserver BRG-IIS-4
      inservice
    rserver BRG-IIS-5
      inservice
    rserver BRG-LINUX-4
      inservice
    rserver BRG-LINUX-5
      inservice
  serverfarm host NON-SSL-TEST
    description SERVERS FOR NON SSL TESTING
    failaction purge
    probe GEN_HTTP
    rserver BRG-IIS-1 80
      inservice
    rserver BRG-IIS-2 80
      inservice
    rserver BRG-IIS-3 80
      inservice
    rserver BRG-LINUX-1 80
      inservice
    rserver BRG-LINUX-2 80
      inservice
    rserver BRG-LINUX-3 80
      inservice

  sticky http-cookie SSL_TERM_COOKIE GROUP_10
    cookie insert browser-expire
    serverfarm ACE_TERM_SERVERS_CLEAR
  sticky http-cookie SSL_INIT_COOKIE GROUP_20
    cookie insert browser-expire
    serverfarm ACE_INIT_SERVERS_SSL
  sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
    cookie insert browser-expire
    serverfarm ACE_END_TO_END_SERVERS_SSL
  sticky http-cookie NON_SSL_TESTING GROUP_40
    cookie insert browser-expire
    serverfarm NON-SSL-TEST

  class-map type http inspect match-any INSPECT_HTTP_GOOD
    2 match request-method rfc connect
    3 match request-method rfc delete
    4 match request-method rfc get
    5 match request-method rfc head
    6 match request-method rfc options
    7 match request-method rfc post
    8 match request-method rfc put
    9 match request-method rfc trace
    10 match url .*
    11 match request-method ext copy
    12 match request-method ext edit
    13 match request-method ext getattr
    14 match request-method ext getattrname
    15 match request-method ext getprops
    16 match request-method ext index
    17 match request-method ext lock
    18 match request-method ext mkdir
    19 match request-method ext move
    20 match request-method ext revadd
    21 match request-method ext revlabel
    22 match request-method ext revlog
    23 match request-method ext revnum
    24 match request-method ext save
    25 match request-method ext setattr
    26 match request-method ext startrev
    27 match request-method ext stoprev
    28 match request-method ext unedit
    29 match request-method ext unlock
  class-map match-all L3_114
    2 match virtual-address 192.168.130.114 any
  class-map type http loadbalance match-all LB_CLASS_HTTP
    2 match http url .*
    3 match source-address 10.1.0.0 255.255.0.0
  class-map match-all NON-SSL_TEST_114
    description NON-SSL_TEST
    2 match virtual-address 192.168.130.114 tcp eq www
  class-map type management match-any REMOTE
    2 match protocol telnet any
    3 match protocol ssh any
    4 match protocol icmp any
    5 match protocol http any
    6 match protocol https any
    7 match protocol snmp any
  class-map match-all SSL_END_TO_END_113
    description END to END SSL VIP
    2 match virtual-address 192.168.130.113 tcp eq https
  class-map match-all SSL_INIT_112
    description SSL INIT CLEAR VIP
    2 match virtual-address 192.168.130.112 tcp eq www
  class-map match-all SSL_TERM_111
    description TERM SSL VIP
    2 match virtual-address 192.168.130.111 tcp eq https
  class-map type http loadbalance match-all STICK_ME_TO_SERVER
    description STICKY FOR SSL TESTING
    2 match http url .*.jpg
    3 match source-address 10.1.0.0 255.255.0.0
  class-map match-all URL*_L7

  policy-map type management first-match POLICY_MGMT
    class REMOTE
      permit

  policy-map type loadbalance first-match NON_SSL_TESTING
    class STICK_ME_TO_SERVER
      sticky-serverfarm GROUP_40
      insert-http SRC_IP header-value "%is"
      insert-http SRC_Port header-value "%ps"
      insert-http I_AM header-value "NON_SSL"
      insert-http DEST_IP header-value "%id"
      insert-http DEST_Port header-value "%pd"
    class LB_CLASS_HTTP
      serverfarm NON-SSL-TEST
      insert-http SRC_IP header-value "%is"
      insert-http I_AM header-value "NON_SSL"
      insert-http DEST_Port header-value "%pd"
      insert-http DEST_IP header-value "%id"
      insert-http SRC_Port header-value "%ps"
  policy-map type loadbalance first-match PLBSF_L3
    class class-default
      serverfarm L3
```

```
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
  class STICK_ME_TO_SERVER
    sticky-serverfarm GROUP_30
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "SSL_END_TO_END"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    ssl-proxy client ACE_AS_CLIENT
  class LB_CLASS_HTTP
    serverfarm ACE_END_TO_END_SERVERS_SSL
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_END_TO_END"
policy-map type loadbalance first-match POLICY_SSL_INIT
  class STICK_ME_TO_SERVER
    sticky-serverfarm GROUP_20
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "SSL_INIT"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    ssl-proxy client ACE_AS_CLIENT
  class LB_CLASS_HTTP
    serverfarm ACE_INIT_SERVERS_SSL
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "SSL_INIT"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_TERM
  class STICK_ME_TO_SERVER
    sticky-serverfarm GROUP_10
    insert-http I_AM header-value "SSL_TERM"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "is"
  class LB_CLASS_HTTP
    serverfarm ACE_TERM_SERVERS_CLEAR
    insert-http I_AM header-value "SSL_TERM"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "is"

policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_TERM_111
    loadbalance vip inservice
    loadbalance policy POLICY_SSL_TERM
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server ACE_TERM
    connection advanced-options TCP_PARAM
  class SSL_INIT_112
    loadbalance vip inservice
    loadbalance policy POLICY_SSL_INIT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class SSL_END_TO_END_113
    loadbalance vip inservice
    loadbalance policy POLICY_SSL_END_TO_END
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server ACE_END_TO_END
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST_114
    loadbalance vip inservice
    loadbalance policy NON_SSL_TESTING
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
  class L3_114
    loadbalance vip inservice
    loadbalance policy PLBSF_L3
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281

service-policy input POLICY_MGMT

interface vlan 130
  ip address 192.168.130.8 255.255.255.0
  alias 192.168.130.7 255.255.255.0
  peer ip address 192.168.130.9 255.255.255.0
  fragment chain 256
  fragment min-mtu 68
  access-group input everyone
  nat-pool 1 192.168.130.70 192.168.130.70 netmask 255.255.255.0 pat
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.8 255.255.255.0
  alias 172.28.1.7 255.255.255.0
  peer ip address 172.28.1.9 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input everyone
  nat-pool 1 192.168.130.71 192.168.130.71 netmask 255.255.255.0 pat
  no shutdown

ip route 10.1.0.0 255.255.255.0 192.168.130.254
username admin password 5 $1$wGwStjD1$n2XN6XsZ5uB50mwxvwQHA.  role Admin domain
default-domain

snmp-server community ACE-private group Network-Monitor
snmp-server community ACE-public group Network-Monitor

snmp-server host 10.1.0.236 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
```

```
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-1_ACE2-1/SH-LOAD# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:17 ---

+++ 09:57:17 6K-1_ACE2-1 ctxExec +++
changeto SH-LOAD


6K-1_ACE2-1/SH-LOAD#


6K-1_ACE2-1/SH-LOAD# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:17 ---

+++ 09:57:17 6K-1_ACE2-1 ctxExec +++
changeto SH-Silver


6K-1_ACE2-1/SH-Silver#


6K-1_ACE2-1/SH-Silver# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:17 ---

+++ 09:57:17 6K-1_ACE2-1 ctxExec +++
changeto SH-Silver


6K-1_ACE2-1/SH-Silver# show running-config

Generating configuration....


logging enable
logging standby
logging timestamp
logging trap 5
logging history 7
logging buffered 5
logging monitor 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7


tacacs-server key 7 "vwjjzamggu"
tacacs-server host 10.86.83.215 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-silver

crypto chaingroup CHAIN1
```

```
  cert term.pem
  cert init.pem
  cert end-to-end.pem
aaa authentication login default group sh-silver local
aaa accounting default group sh-silver local
aaa authentication login error-enable

access-list acl1 line 8 extended permit ip host 172.28.1.6 host 172.28.1.23
access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 5
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_CLEAR"
probe https GEN_HTTPS
  interval 5
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_SSL"


parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 30
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
  urgent-flag clear
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version SSL3
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
```

```
  cipher RSA_WITH_RC4_128_MD5 priority 6
  version SSL3
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  session-cache timeout 180
  version TLS1
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow

action-list type modify http Header-Deletions
  header insert response Cache-Control header-value "max-age=901"
  header delete response Etag
  header delete response ETag
  header delete response Pragma
  header delete response Set-Cookie2
action-list type modify http HTTP-HTTPS-Rewrite
  header insert response Cache-Control header-value "max-age=901"
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header insert request Pragma header-value "no-cache, no-cache is the only pragma value I 
have ever seen"
  header delete both Content-Type
  header delete response ETag
  header delete response Set-Cookie
  header rewrite response Keep-Alive header-value "timeout=150" replace "timeout=300"
  header rewrite request User-Agent header-value ".*curl.*" replace "Mozilla/4.0 
(compatible; cURL; Runs on Linux)"
  ssl url rewrite location "192.168.130.11"
  ssl url rewrite location "www.ssl-term-ace.com"
action-list type modify http Header-Deletions-Cache=2592008
  header insert response Cache-Control header-value "max-age=2592008"
  header delete response Etag
  header delete response ETag
  header delete response Pragma
  header delete response Set-Cookie2
action-list type modify http HTTP-HTTPS-Rewrite-E2E
  header insert response Cache-Control header-value "max-age=901"
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header delete both Content-Type
  header delete response ETag
  header delete response Set-Cookie
  ssl url rewrite location "192.168.130.13" sslport 444
  ssl url rewrite location "www.ssl-end-to-end-ace.com" sslport 444

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
```

```
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice
rserver redirect http-to-https
  webhost-redirection https://192.168.130.11
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  key pkey.pem
  cert end-to-end.pem
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  key pkey.pem
  cert term.pem
  ssl advanced-options PARM_ACE_TERM_EXPORT_CIPHERS

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
```

```
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host L4
  probe GEN_HTTP
  rserver BRG-IIS-1
    inservice
  rserver BRG-LINUX-1
    inservice
serverfarm host L7
  rserver BRG-IIS-2
    inservice
  rserver BRG-LINUX-2
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host ONE-IIS-SERVER
  rserver BRG-IIS-1
    inservice
serverfarm redirect REDIRECT-GET-SLASH
  rserver http-to-https
    inservice
serverfarm host TCP-NORM-FARM
  predictor leastconns
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-IIS-3
```

```
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver BRG-LINUX-3
    inservice
serverfarm host TELNET-NORM-TEST
  rserver BRG-LINUX-1
    inservice

sticky http-cookie SSL_TERM_COOKIE GROUP_10
  cookie insert browser-expire
  serverfarm ACE_TERM_SERVERS_CLEAR
sticky http-cookie SSL_INIT_COOKIE GROUP_20
  cookie insert browser-expire
  serverfarm ACE_INIT_SERVERS_SSL
sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
  cookie insert browser-expire
  timeout 30
  serverfarm ACE_END_TO_END_SERVERS_SSL
sticky http-cookie NON_SSL_TESTING GROUP_40
  cookie insert browser-expire
  serverfarm NON-SSL-TEST
sticky ip-netmask 255.255.255.0 address both NEW_GROUP
  serverfarm NON-SSL-TEST

class-map match-all GENERIC
class-map match-all L4-VIP_20:80
class-map match-all NON-SSL_TEST
  description NON-SSL_TEST
class-map match-all SSL_END_TO_END_13
  description STICKY FOR SSL TESTING
class-map match-all TCP-NORM-TEST
  description TCP NORM TEST
  2 match virtual-address 192.168.130.17 tcp eq 22

policy-map type management first-match POLICY_MGMT

policy-map type loadbalance first-match GENERIC
policy-map type loadbalance first-match NON_SSL_TESTING
policy-map type loadbalance first-match PLBSF_L4
  class class-default
    serverfarm L4
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
policy-map type loadbalance first-match POLICY_SSL_INIT
policy-map type loadbalance first-match POLICY_SSL_TERM
policy-map type loadbalance first-match TCP-NORM-TESTING
policy-map type loadbalance first-match TELNET-NORM_TESTING
  class class-default
    serverfarm TELNET-NORM-TEST

policy-map type inspect http all-match INSPECT_GOOD_HTTP

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_END_TO_END_13
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server ACE_END_TO_END
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class TCP-NORM-TEST
    loadbalance vip inservice
    loadbalance policy TCP-NORM-TESTING
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 281
    connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
  class GENERIC
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
  class L4-VIP_20:80
    nat dynamic 1 vlan 281

interface vlan 130
  ip address 192.168.130.2 255.255.255.0
  alias 192.168.130.1 255.255.255.0
  peer ip address 192.168.130.3 255.255.255.0
  fragment min-mtu 80
  access-group input eveyone
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.5 255.255.255.0
  alias 172.28.1.4 255.255.255.0
  peer ip address 172.28.1.6 255.255.255.0
  fragment min-mtu 80
  access-group input eveyone
  nat-pool 1 192.168.130.201 192.168.130.201 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown

domain SH-Silver-Domain
  add-object all

ip route 10.1.0.0 255.255.255.0 192.168.130.254
ip route 192.168.120.0 255.255.255.0 192.168.130.254
ip route 10.3.0.0 255.255.255.0 192.168.130.254
username admin password 5 $1$5SD7.E4T$/xUH04GK/gFCx8PdOKZ.L/  role Admin domain
default-domain
username vrtnetmonsilver password 5 $1$SGEScCRK$qW1k3UCNiR/jIFdcYsIZx.  role
Network-Monitor domain SH-Silver-Domain
username vrtadmsilver password 5 $1$c64ET9q4$fYPmJa5OVQvntr1quXN0O.  role Admin domain
SH-Silver-Domain
username vrtjohndoe password 5 $1$Ej3RvPBx$aMHRl4SocCBTriiB9nOmf/  role Admin domain
SH-Silver-Domain
username netmon password 5 $1$hPvrGd2M$Usu6WSw9RdIFxP.qec0vp1  role Network-Monitor domain
default-domain




6K-1_ACE2-1/SH-Silver# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:18 ---

+++ 09:57:18 6K-1_ACE2-1 ctxExec +++
changeto SH-Silver


6K-1_ACE2-1/SH-Silver#
```

```
6K-1_ACE2-1/SH-Silver# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:18 ---

+++ 09:57:18 6K-1_ACE2-1 ctxExec +++
changeto SH-Bridge



NOTE: Configuration mode has been disabled on all sessions


6K-1_ACE2-1/SH-Bridge#


6K-1_ACE2-1/SH-Bridge# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:18 ---

+++ 09:57:18 6K-1_ACE2-1 ctxExec +++
changeto SH-Bridge



NOTE: Configuration mode has been disabled on all sessions


6K-1_ACE2-1/SH-Bridge# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging history 5
logging buffered 5
logging device-id context-name
logging host 10.1.0.242 udp/514
logging host 10.1.0.236 udp/514


tacacs-server key 7 "vwjjzamggu"
tacacs-server host 172.29.0.235 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.236 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.237 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-bridge
  server 172.29.0.236
  server 172.29.0.237

arp 192.168.120.80 00.00.11.11.22.22
arp 172.29.0.200 00.00.22.22.44.44
arp learned-interval 60
aaa authentication login default group sh-bridge local
aaa accounting default group sh-bridge
access-list BPDU-ALLOW ethertype permit bpdu
```

```
access-list ICMP-ONLY line 8 extended permit icmp any any
access-list NAT_ACCESS line 20 extended permit tcp any host 172.28.3.121
access-list NAT_ACCESS line 30 extended permit tcp any host 172.28.3.122
access-list NAT_ACCESS line 40 extended permit tcp any host 172.28.3.140
access-list anyone-ip line 10 extended permit ip any any
access-list anyone-tcp line 10 extended permit tcp any any

script file 1 FTP_PROBE_SCRIPT
script file 2 TFTP_PROBE
script file 3 LDAP_PROBE


probe icmp FA-PURGE-ICMP
  ip address 172.28.4.253 routed
  interval 5
  passdetect interval 2
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
probe ftp FTP
  interval 10
  faildetect 2
  passdetect interval 10
  receive 5
  expect status 220 220
  open 3
probe icmp HA-ICMP
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:1755
  port 1755
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:554
  port 554
  interval 2
  faildetect 2
  passdetect interval 2
probe http HTTP
  interval 5
  passdetect interval 10
  receive 5
  expect status 200 200
  open 3
probe icmp ICMP
  interval 10
  faildetect 2
  passdetect interval 10
probe dns PRB-DNS1
  description "all good addresses"
  interval 5
  passdetect interval 2
  domain www1.safeharbor.com
  expect address 1.1.1.1
  expect address 1.1.1.2
  expect address 1.1.1.3
probe dns PRB-DNS2
```

```
    description "one good address"
    interval 5
    passdetect interval 2
    domain www2.safeharbor.com
    expect address 2.1.1.1
  probe dns PRB-DNS3
    description "2 good addresses, bumpy case"
    interval 5
    passdetect interval 2
    domain WwW3.SaFeHaRbOr.CoM
    expect address 3.1.1.3
    expect address 3.1.1.2
  probe dns PRB-DNS4
    description "one good and one bad address"
    interval 5
    passdetect interval 2
    domain www4.safeharbor.com
    expect address 192.168.1.4
    expect address 4.1.1.1
  probe dns PRB-DNS5
    description "all bad addresses"
    interval 5
    passdetect interval 2
    domain www4.safeharbor.com
    expect address 192.168.1.5
    expect address 192.168.1.6
    expect address 1.1.1.1
  probe dns PRB-DNS6:2222
    description "dns not running on this port, but addresses good"
    port 2222
    interval 5
    passdetect interval 2
    domain www1.safeharbor.com
    expect address 1.1.1.1
    expect address 1.1.1.2
    expect address 1.1.1.3
  probe http PRB-HTTP:84
    port 84
    interval 5
    passdetect interval 4
    passdetect count 10
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:85
    description Server RST 1byte data
    port 85
    interval 5
    passdetect interval 2
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:86
    description Server RST 3200byte data
    port 86
    interval 5
    passdetect interval 2
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:87
    description Server FIN 1byte data
    port 87
    interval 5
    passdetect interval 2
    expect status 200 200
  probe http PRB-HTTP:88
    description Server FIN 3200byte data
    port 88
    interval 5
    passdetect interval 2
    expect status 200 200
  probe https PRB-SSL:443
    interval 5
    passdetect interval 10
    request method get url /index.txt
    expect status 200 200
    header Via header-value "PRB-SSL:443 Probe Header"
    hash
  probe icmp PRED-PING
    ip address 172.28.4.243 routed
    interval 5
    faildetect 2
    passdetect interval 2
  probe radius RADIUS
    interval 2
    faildetect 2
    passdetect interval 2
    credentials lab labtest1 secret ace
  probe scripted SCRIPT_FTP:21
    interval 10
    passdetect interval 2
    passdetect count 2
    receive 5
    script FTP_PROBE_SCRIPT /home/lab/ftp-files/file01.log lab labtest1
  probe scripted SCRIPT_LDAP
    interval 10
    passdetect interval 5
    passdetect count 2
    receive 5
    script LDAP_PROBE
  probe scripted SCRIPT_TFTP
    interval 10
    passdetect interval 5
    passdetect count 2
    receive 5
    script TFTP_PROBE "large file name to test the tftp scripted probe.exe"
  probe https SSL
    interval 5
    passdetect interval 10
    expect status 200 299
    header Via header-value "ACE_Gold_SSL"
    connection term forced
  probe https SSL-445:FIN
    port 445
    interval 5
    passdetect interval 10
    expect status 200 200
  probe https SSL-445:RST
    port 445
    interval 5
    passdetect interval 10
    expect status 200 200
    connection term forced
  probe tcp TCP
    interval 5
    faildetect 2
    passdetect interval 10
    open 3
  probe udp UDP
    interval 5
    passdetect interval 2
  probe udp UDP:2222
    port 2222
    interval 5
    passdetect interval 2


  parameter-map type connection 120SECOND-IDLE
    set timeout inactivity 120
    set tcp timeout half-closed 30
  parameter-map type connection 1SECOND-IDLE
    set timeout inactivity 1
  parameter-map type connection 2SECOND-IDLE
    set timeout inactivity 2
  parameter-map type connection 60SECOND-IDLE
    set timeout inactivity 60
    set tcp timeout half-closed 30
  parameter-map type http COOKIE-DELIM
    persistence-rebalance
    set secondary-cookie-delimiters @$
  parameter-map type http COOKIE-INSERT-HDR-PARSE
    persistence-rebalance
    set header-maxparse-length 4000
  parameter-map type ssl EXPORT_CIPHERS
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  parameter-map type connection HALF-CLOSE_4s
    set tcp timeout half-closed 4
  parameter-map type connection INFINITE-IDLE
    set timeout inactivity 0
    set tcp timeout half-closed 30
  parameter-map type connection LLFlows
    set timeout inactivity 0
  parameter-map type connection NORM
    reserved-bits clear
  parameter-map type connection NORM_TCP
    tcp-options timestamp allow
    reserved-bits drop
    syn-data drop
    urgent-flag clear
  parameter-map type http PARSE_LENGTH
    persistence-rebalance
  parameter-map type http PERSIST-REBAL-4K
    persistence-rebalance
  parameter-map type http PERSIST-REBALANCE
    persistence-rebalance
  parameter-map type connection PRED-CONNS-UDP_CONN
    set timeout inactivity 300
  parameter-map type ssl RC4_128_MD5_CIPHER
    cipher RSA_WITH_RC4_128_MD5
    version TLS1
  parameter-map type http REUSE-REBAL
    server-conn reuse
    persistence-rebalance
  parameter-map type ssl STRONG_CIPHERS
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA priority 2
    cipher RSA_WITH_AES_256_CBC_SHA priority 3
  parameter-map type ssl WEAK_CIPHERS
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA priority 2
    cipher RSA_WITH_DES_CBC_SHA priority 3
  parameter-map type connection wan-opt
    set tcp wan-optimization rtt 0

  rserver host BRG-11
    ip address 172.28.3.11
    inservice
  rserver host BRG-12
    ip address 172.28.3.12
    inservice
  rserver host BRG-13
    ip address 172.28.3.13
    inservice
  rserver host BRG-14
    ip address 172.28.3.14
    inservice
  rserver host BRG-15
    ip address 172.28.3.15
    inservice
  rserver host LOCAL-151
    ip address 172.28.3.151
    inservice
  rserver host LOCAL-152
    ip address 172.28.3.152
    inservice
  rserver host LOCAL-153
    ip address 172.28.3.153
    inservice
  rserver host LOCAL-154
    ip address 172.28.3.154
    inservice
  rserver redirect REDIRECT-100K
    webhost-redirection http://172.28.3.132/redirect-100k.html 302
    inservice
  rserver redirect REDIRECT-10K
    webhost-redirection http://172.28.3.133/redirect-10k.html 302
    inservice
  rserver redirect REDIRECT-1K
    webhost-redirection http://172.28.3.134/redirect-1k.html 302
    inservice
  rserver host RT-239-FRAGRTR
    ip address 172.28.4.239
    inservice
  rserver host RT-240
    ip address 172.28.4.240
    inservice
  rserver host RT-241
    ip address 172.28.4.241
    inservice
  rserver host RT-242
    ip address 172.28.4.242
    inservice
  rserver host RT-243
    ip address 172.28.4.243
    inservice
  rserver host RT-244
    ip address 172.28.4.244
    inservice
  rserver host RT-245
    ip address 172.28.4.245
    inservice
  rserver host WEIGHT-80
    ip address 10.1.0.235
    weight 80
```

```
    inservice

  serverfarm host ADD-REM-SRV
    predictor leastconns
    probe TCP
    rserver BRG-13
      inservice
    rserver BRG-14
      inservice
    rserver LOCAL-153
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-243
      inservice
    rserver RT-244
      inservice
  serverfarm host COOKIE
    probe ICMP
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver RT-244
      inservice
  serverfarm host COOKIE-HASH
    predictor leastconns
    rserver RT-240
      inservice
    rserver RT-240 90
      inservice
    rserver RT-241
      inservice
    rserver RT-241 90
      inservice
    rserver RT-242
      inservice
    rserver RT-242 90
      inservice
    rserver RT-243
      inservice
    rserver RT-243 90
      inservice
  serverfarm host COOKIE-INSERT
    rserver BRG-14
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-245
      inservice
  serverfarm host COOKIE-INSERT2
    probe HTTP
    rserver BRG-14
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-245
      inservice
  serverfarm host COOKIE1
    probe HTTP
    rserver BRG-12
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-240
      inservice
  serverfarm host COOKIE2
    probe TCP
    rserver BRG-11
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-241
      inservice
  serverfarm host CS-COOKIES
    probe TCP
    rserver BRG-12
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-240
      inservice
  serverfarm host CS-MOZILLA
    rserver LOCAL-151
      inservice
    rserver RT-240
      inservice
  serverfarm host CS-MSIE
    rserver RT-242
      inservice
    rserver RT-243
      inservice
  serverfarm host DEFAULT
    probe ICMP
    rserver BRG-15
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-245
      inservice
  serverfarm host FA-PURGE
    failaction purge
    rserver BRG-11
      inservice
    rserver BRG-14
      probe FA-PURGE-ICMP
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-154
      probe FA-PURGE-ICMP
      inservice
    rserver RT-242
      inservice
    rserver RT-244
      probe FA-PURGE-ICMP
      inservice
  serverfarm host FTP
    probe FTP
    rserver BRG-11 21
      inservice
    rserver BRG-12 21
      inservice
    rserver LOCAL-151 21
      inservice
    rserver LOCAL-152 21
      inservice
    rserver RT-240 21
```

```
      inservice
    rserver RT-241 21
      inservice
  serverfarm host GEN-443
    probe SSL
    rserver BRG-12
      inservice
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm host GEN-80
    probe TCP
    rserver BRG-12
      inservice
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm host GEN-FTP
    probe FTP
    rserver BRG-13
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-240
      inservice
  serverfarm host GEN-UDP
    probe ICMP
    rserver BRG-11
      inservice
    rserver LOCAL-151
      inservice
    rserver RT-244
      inservice
  serverfarm host GEN2-80
    probe TCP
    rserver BRG-11
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-241
      inservice
  serverfarm host HDR-IXIA
    rserver BRG-14
      inservice
    rserver RT-241
      inservice
  serverfarm host HEADER
    probe HTTP
    rserver LOCAL-152
      inservice
    rserver RT-240
      inservice
    rserver RT-244
      inservice
  serverfarm host HEADER-INSERT
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-153
      inservice
  serverfarm host HEADER-INSERT2
    rserver BRG-12
      inservice
    rserver BRG-14
      inservice
    rserver LOCAL-153
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-240
      inservice
    rserver RT-241
      inservice
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm host ICMP
    probe ICMP
    rserver BRG-11
      inservice
    rserver LOCAL-152
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-241
      inservice
    rserver RT-242
      inservice
  serverfarm host ICMP2
    rserver BRG-11 7777
      inservice
    rserver LOCAL-152
      inservice
    rserver LOCAL-153 7777
      inservice
    rserver RT-241
      inservice
    rserver RT-242 7777
      inservice
  serverfarm host IDLE-TCP
    probe TCP
    rserver BRG-15
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-244
      inservice
  serverfarm host IDLE-UDP
    probe UDP:2222
    rserver BRG-15
      inservice
    rserver LOCAL-154
```

```
    inservice
  rserver RT-244
    inservice
serverfarm host LDAP
  rserver BRG-15
    inservice
  rserver LOCAL-151
    inservice
  rserver RT-244
    inservice
serverfarm host LENGTHS
  rserver RT-241
    inservice
  rserver RT-244
    inservice
serverfarm host LENGTHS-2
  rserver RT-240
    inservice
  rserver RT-245
    inservice
serverfarm host MAX-CONN
  probe HTTP
  rserver LOCAL-151
    conn-limit max 4 min 2
    inservice
serverfarm host MAX-CONN2
  probe HTTP
  rserver LOCAL-151
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 90
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 91
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 92
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 93
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 94
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 95
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-243
    conn-limit max 500 min 2
    inservice
serverfarm host NORM
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-153
    inservice
  rserver RT-241
    inservice
serverfarm host NORM2_L7
```

```
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  retcode 100 599 check count
  rserver BRG-11 80
    inservice
  rserver LOCAL-153 80
    inservice
  rserver RT-241 80
    inservice
serverfarm host PERSISTENT
  rserver LOCAL-154
    inservice
  rserver RT-240
    inservice
  rserver RT-242
    inservice
  rserver RT-243
    inservice
serverfarm host PRED-CONNS
  predictor leastconns
  rserver BRG-11
    inservice
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver BRG-15
    inservice
  rserver LOCAL-151
    inservice
  rserver LOCAL-152
    inservice
  rserver LOCAL-153
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-240
    inservice
  rserver RT-241
    inservice
  rserver RT-242
    inservice
  rserver RT-243
    inservice
  rserver RT-244
    inservice
serverfarm host PRED-CONNS-UDP
  failaction purge
  predictor leastconns
  rserver BRG-11 2222
    inservice
  rserver LOCAL-151 2222
    inservice
  rserver LOCAL-153 2222
    inservice
  rserver LOCAL-154 2222
    inservice
  rserver RT-240 2222
    inservice
  rserver RT-242 2222
    inservice
  rserver RT-244 2222
    probe PRED-PING
    inservice
serverfarm host PREDICTOR
  predictor leastconns
  probe TCP
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver LOCAL-152
    inservice
  rserver LOCAL-153
    inservice
  rserver RT-243
    inservice
  rserver RT-244
    inservice
serverfarm host PROBES
  predictor leastconns
  rserver BRG-13
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-244
    inservice
serverfarm host PROBES-2
  predictor leastconns
  rserver BRG-11
    inservice
  rserver LOCAL-152
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-241
    inservice
  rserver RT-244
    inservice
serverfarm host PROBES-MANY
  predictor leastconns
  probe SCRIPT_TFTP
  rserver BRG-11
    probe RADIUS
    inservice
  rserver LOCAL-152
    inservice
  rserver LOCAL-154
    probe RADIUS
    inservice
  rserver RT-241
    inservice
  rserver RT-243
    inservice
  rserver RT-244
    probe RADIUS
    inservice
serverfarm host RADIUS
  rserver BRG-11
    inservice
  rserver LOCAL-151
    inservice
  rserver RT-244
    inservice
serverfarm host RED-ALL-SVRS
```

```
    rserver RT-240
      inservice
    rserver RT-241
      inservice
  serverfarm host REDIRECT
    rserver RT-242
      inservice
    rserver RT-243
      inservice
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm redirect REDIRECT-100K
    rserver REDIRECT-100K
      inservice
  serverfarm redirect REDIRECT-10K
    rserver REDIRECT-10K
      inservice
  serverfarm redirect REDIRECT-1K
    rserver REDIRECT-1K
      inservice
  serverfarm host RHI
    rserver BRG-11
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-241
      inservice
  serverfarm host SORRY
    rserver RT-240
      inservice
  serverfarm host SORRY-BACK
    rserver LOCAL-151
      inservice
    rserver RT-243
      inservice
  serverfarm host STICKY-COOKIE
    probe ICMP
    rserver BRG-11
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-241
      inservice
    rserver RT-242
      inservice
  serverfarm host STICKY-HEADER
    probe HTTP
    rserver BRG-12
      inservice
    rserver LOCAL-151
      inservice
    rserver RT-243
      inservice
  serverfarm host STICKY-HEADER2
    probe HTTP
    rserver BRG-13
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-244
      inservice
  serverfarm host STICKY-NETMASK
```

```
    probe ICMP
    rserver BRG-12
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-242
      inservice
  serverfarm host TCP-REUSE
    rserver BRG-15
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-245
      inservice
  serverfarm host UDP
    probe UDP
    rserver BRG-11
      inservice
    rserver LOCAL-151
      inservice
    rserver RT-241
      inservice
  serverfarm host URL-MAP-128K
    rserver RT-241
      inservice
    rserver RT-242
      inservice
  serverfarm host URL-MAP-16K
    rserver BRG-12
      inservice
    rserver RT-242
      inservice
  serverfarm host URL-MAP-32K
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm host URL-MAP-512K
    rserver RT-242
      inservice
    rserver RT-243
      inservice
  serverfarm host URL-MAP-64K
    rserver RT-242 91
      inservice
    rserver RT-244
      inservice
  serverfarm host URL-MAPS
    rserver RT-241
      inservice
    rserver RT-242
      inservice
    rserver RT-244
      inservice
  serverfarm host WEIGHT
    probe HTTP
    rserver BRG-11
      weight 10
      inservice
    rserver LOCAL-152
      weight 40
      inservice
    rserver RT-240
      weight 20
      inservice
    rserver RT-243
      weight 30
      inservice

  sticky ip-netmask 255.255.255.255 address source STKY-GRP-30
    timeout 40
    replicate sticky
    serverfarm GEN-80
  sticky http-cookie cookie-gold-grp40 STKY-GRP-40
    cookie insert browser-expire
    timeout 1
    replicate sticky
    serverfarm GEN2-80
  sticky ip-netmask 255.255.255.255 address both STKY-GRP-31
    timeout 40
    replicate sticky
    serverfarm GEN-UDP
  sticky ip-netmask 255.255.255.255 address both STKY-GRP-32
    timeout 40
    replicate sticky
    serverfarm GEN-FTP
  sticky http-cookie COOKIE_TEST COOKIE-GROUP
    cookie secondary URLCOOKIE
    timeout 40
    replicate sticky
    serverfarm STICKY-COOKIE
    16 static cookie-value "PATRIOTS0" rserver LOCAL-151
  sticky http-header MSISDN HEADER-GROUP-42
    timeout 30
    replicate sticky
    serverfarm STICKY-HEADER
  sticky http-header TestHeader HEADER-GROUP-41
    header offset 15 length 7
    timeout 30
    replicate sticky
    serverfarm STICKY-HEADER2
  sticky http-cookie COOKIE_INSERT COOKIE-INSERT-GROUP-45
    cookie insert
    timeout 1
    replicate sticky
    serverfarm COOKIE-HASH
  sticky http-cookie COOKIE_TEST COOKIE-MAP-GROUP
    replicate sticky
    serverfarm CS-COOKIES
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-29
    timeout 30
    replicate sticky
    serverfarm MAX-CONN
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-11
    timeout 30
    replicate sticky
    serverfarm URL-MAP-16K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-13
    timeout 30
    replicate sticky
    serverfarm URL-MAP-64K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-12
    timeout 30
    replicate sticky
    serverfarm URL-MAP-32K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-14
    timeout 30
    replicate sticky
    serverfarm URL-MAP-128K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-15
    timeout 30
    replicate sticky
    serverfarm URL-MAP-512K
  sticky http-cookie Safeharbor-Cookie1 COOKIE-GROUP-42
    cookie insert
    timeout 30
    replicate sticky
  sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes
  COOKIE-INSERT-GROUP-46
    cookie insert
    timeout 1
    replicate sticky
    serverfarm COOKIE-INSERT
  sticky http-cookie COOKIE_TEST STKY-GRP-43
    replicate sticky
    serverfarm PREDICTOR
  sticky ip-netmask 255.255.255.255 address both STKY-GRP-33
    timeout 20
    replicate sticky
    serverfarm STICKY-NETMASK

  class-map type http loadbalance match-all 128K-FORWARDING
    2 match http url .*128k.*
  class-map type http loadbalance match-all 16K-FORWARDING
    2 match http url .*16k.*
  class-map type http loadbalance match-all 32K-FORWARDING
    2 match http url .*32k.*
  class-map type http loadbalance match-all 512K-FORWARDING
    2 match http url .*512k.*
  class-map type http loadbalance match-all 64K-FORWARDING
    2 match http url .*64k.*
  class-map match-all ADD-REM-SRV-VIP_110:80
  class-map type http loadbalance match-all BROWSER_MOZILLA
    2 match http header User-Agent header-value ".*Mozilla.*"
  class-map type http loadbalance match-all BROWSER_MOZILLA40
    2 match http header User-Agent header-value ".*Mozilla/4.0.*"
  class-map type http loadbalance match-all BROWSER_MOZILLA50
    2 match http header User-Agent header-value ".*Mozilla/5.0.*"
  class-map type http loadbalance match-all BROWSER_MSIE
    2 match http header User-Agent header-value ".*MSIE.*"
  class-map match-all COOKIE-HASH-VIP_10.20.30.40:80
  class-map match-all COOKIE-MAP:80
  class-map match-all FA-PURGE-VIP_113:ANY
  class-map type ftp inspect match-any FTP-L7-MAX-DENY2
    2 match request-method syst
  class-map type ftp inspect match-any FTP-L7-MIN-DENY
    2 match request-method mkd
    3 match request-method rmd
  class-map match-all FTP-VIP-NAT_119
    2 match destination-address 172.28.3.119 255.255.255.255
  class-map match-all FTP-VIP_119:1111
    2 match access-list ICMP-ONLY
  class-map match-all ICMP-UDP-VIP_138
  class-map match-all LENGTHS-VIP_136:80
    description remote-access-traffic-match
  class-map type http loadbalance match-all MSISDN
    2 match http header MSISDN header-value ".*"
  class-map match-any NAT_CLASS
    2 match access-list NAT_ACCESS
  class-map match-any NORM-VIP_142:80
    2 match any
  class-map type http loadbalance match-any P-COOKIE-INS
```

```
  2 match http url /index.html* method GET
class-map type http loadbalance match-any P-COOKIE-INS2
  2 match http url .*
class-map type http loadbalance match-all P-HDR-INSERT
  2 match http url .*
class-map type http loadbalance match-all P-HDR-IXIA
  2 match http url .*
class-map type http loadbalance match-all P-HDR-SRCDST-IP
  2 match http url .*
class-map match-all PERSISTENT-VIP_131:80
class-map type http loadbalance match-all REDIRECT-10K
  2 match http url .*redirect-10k.html
class-map type http loadbalance match-all REDIRECT-1K
  2 match http url .*redirect-1k.html
class-map match-all REDIRECT-VIP_135:80
class-map match-all UDP-VIP_114:UDP
class-map match-all URL*_L7
class-map match-all URL-MAPS-VIP_130:80
class-map type http loadbalance match-all URLCOOKIE-MAP2
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE2"
class-map match-all WEIGHT_112:80
  2 match virtual-address 172.28.3.112 tcp eq www

policy-map type management first-match P-MGT

policy-map type loadbalance first-match FTP-LB-SF_FTP
  class class-default
    serverfarm FTP
policy-map type loadbalance first-match MAX-CONN-LB-SF_MAX-CONN2
policy-map type loadbalance first-match PLBSF-FTP-test
  class class-default
policy-map type loadbalance first-match PLBSF_ADD-REM-SRV
  class class-default
    serverfarm ADD-REM-SRV
policy-map type loadbalance first-match PLBSF_COOKIE-HASH
  class class-default
    sticky-serverfarm COOKIE-INSERT-GROUP-45
policy-map type loadbalance first-match PLBSF_COOKIE-INSERT
  class P-COOKIE-INS
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Destination_IP header-value "%id"
    insert-http Source-IP header-value "%is"
  class P-COOKIE-INS2
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Destination_IP header-value "%id"
    insert-http Source-IP header-value "%is"
policy-map type loadbalance first-match PLBSF_COOKIE-MAP
  class URLCOOKIE-MAP2
    serverfarm COOKIE2
policy-map type loadbalance first-match PLBSF_FA-PURGE
  class class-default
    serverfarm FA-PURGE
policy-map type loadbalance first-match PLBSF_GEN-443
  class class-default
    serverfarm GEN-443
policy-map type loadbalance first-match PLBSF_GEN-80
  class class-default
    sticky-serverfarm STKY-GRP-30
policy-map type loadbalance first-match PLBSF_GEN-FTP
  class class-default
    sticky-serverfarm STKY-GRP-32
policy-map type loadbalance first-match PLBSF_GEN-UDP
  class class-default
    sticky-serverfarm STKY-GRP-31
```

```
policy-map type loadbalance first-match PLBSF_HDR-IXIA
  class P-HDR-IXIA
    serverfarm HDR-IXIA
policy-map type loadbalance first-match PLBSF_HEADER
policy-map type loadbalance first-match PLBSF_HEADER-INSERT
  class P-HDR-INSERT
    serverfarm HEADER-INSERT
    insert-http Source-IP header-value "%is"
    insert-http Accept header-value "anything"
    insert-http Pragma header-value "Pragma no Pragma that is the question"
    insert-http Destination_iP header-value "%id"
policy-map type loadbalance first-match PLBSF_HEADER-INSERT2
  class P-HDR-SRCDST-IP
    serverfarm HEADER-INSERT2
    insert-http Source-IP header-value "%is"
    insert-http Destination_iP header-value "%id"
policy-map type loadbalance first-match PLBSF_ICMP
policy-map type loadbalance first-match PLBSF_ICMP2
  class class-default
    serverfarm ICMP2
policy-map type loadbalance first-match PLBSF_IDLE-TCP
  class class-default
    serverfarm IDLE-TCP
policy-map type loadbalance first-match PLBSF_IDLE-UDP
  class class-default
    serverfarm IDLE-UDP
policy-map type loadbalance first-match PLBSF_LENGTHS
policy-map type loadbalance first-match PLBSF_MAX-CONN
  class class-default
    serverfarm MAX-CONN2
policy-map type loadbalance first-match PLBSF_NORM
  class class-default
    serverfarm NORM
policy-map type loadbalance first-match PLBSF_NORM2_L7
  class class-default
    serverfarm NORM2_L7
policy-map type loadbalance first-match PLBSF_PERSISTENT
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm PERSISTENT
policy-map type loadbalance first-match PLBSF_PRED-CONNS
  class class-default
    serverfarm PRED-CONNS
policy-map type loadbalance first-match PLBSF_PRED-CONNS-UDP
  class class-default
    serverfarm PRED-CONNS-UDP
policy-map type loadbalance first-match PLBSF_PREDICTOR
  class class-default
    serverfarm PREDICTOR
policy-map type loadbalance first-match PLBSF_RED-ALL-SVRS
  class class-default
    serverfarm RED-ALL-SVRS
policy-map type loadbalance first-match PLBSF_REDIRECT
  class REDIRECT-1K
    serverfarm REDIRECT-1K
  class REDIRECT-10K
    serverfarm REDIRECT-10K
policy-map type loadbalance first-match PLBSF_RHI
policy-map type loadbalance first-match PLBSF_SORRY
  class class-default
    serverfarm SORRY backup SORRY-BACK
policy-map type loadbalance first-match PLBSF_STICKY-COOKIE
policy-map type loadbalance first-match PLBSF_STICKY-HEADER
  class MSISDN
    sticky-serverfarm HEADER-GROUP-42
  class class-default
    serverfarm DEFAULT
policy-map type loadbalance first-match PLBSF_STICKY-NETMASK
  class class-default
    sticky-serverfarm STKY-GRP-33
policy-map type loadbalance first-match PLBSF_TCP-REUSE
policy-map type loadbalance first-match PLBSF_UDP
  class class-default
    serverfarm UDP
policy-map type loadbalance first-match PLBSF_URL-MAPS
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm URL-MAPS
policy-map type loadbalance first-match PLBSF_WEIGHT
  class class-default
    serverfarm WEIGHT

policy-map type inspect ftp first-match FTP-INSPSF_FTP
  class FTP-L7-MAX-DENY2
    mask-reply

policy-map multi-match NAT_POLICY
  class NAT_CLASS
    nat dynamic 1 vlan 2830
policy-map multi-match NORMALIZATION
policy-map multi-match SH-Gold-VIPs
  class FTP-VIP-NAT_119
    nat dynamic 1 vlan 2830
    nat dynamic 1 vlan 283
  class FTP-VIP_119:1111
  class ADD-REM-SRV-VIP_110:80
    nat dynamic 1 vlan 2830
  class WEIGHT_112:80
    loadbalance vip inservice
    loadbalance policy PLBSF_WEIGHT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 2830
    appl-parameter http advanced-options PERSIST-REBALANCE
  class FA-PURGE-VIP_113:ANY
    nat dynamic 1 vlan 2830
    connection advanced-options INFINITE-IDLE
  class UDP-VIP_114:UDP
    nat dynamic 1 vlan 2830
    connection advanced-options 1SECOND-IDLE
  class ICMP-UDP-VIP_138
    nat dynamic 1 vlan 2830
    connection advanced-options 60SECOND-IDLE
  class COOKIE-HASH-VIP_10.20.30.40:80
    appl-parameter http advanced-options PERSIST-REBALANCE
  class URL-MAPS-VIP_130:80
    nat dynamic 1 vlan 2830
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PERSISTENT-VIP_131:80
    nat dynamic 1 vlan 2830
    appl-parameter http advanced-options PERSIST-REBALANCE
  class REDIRECT-VIP_135:80
    appl-parameter http advanced-options PERSIST-REBALANCE
  class LENGTHS-VIP_136:80
    appl-parameter http advanced-options PARSE_LENGTH
  class NORM-VIP_142:80
    nat dynamic 1 vlan 2830
policy-map multi-match SH-Gold-VIPs3

service-policy input P-MGT

interface vlan 99
  ip address 192.168.99.5 255.255.255.0
  peer ip address 192.168.99.6 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 106
  ip address 192.168.106.2 255.255.255.0
  peer ip address 192.168.106.3 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
interface vlan 283
  description Downstream Bridge VLAN_283-2830
  bridge-group 283
  fragment chain 20
  fragment min-mtu 68
  access-group input BPDU-ALLOW
  access-group input anyone-ip
  nat-pool 1 172.28.3.71 172.28.3.71 netmask 255.255.255.0 pat
  no shutdown
interface vlan 2830
  description Upstream Bridge VLAN_2830-283
  bridge-group 283
  fragment chain 20
  fragment min-mtu 68
  access-group input BPDU-ALLOW
  access-group input anyone-ip
  nat-pool 1 172.28.3.70 172.28.3.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  service-policy input NAT_POLICY
  no shutdown

interface bvi 283
  ip address 172.28.3.2 255.255.255.0
  alias 172.28.3.1 255.255.255.0
  peer ip address 172.28.3.3 255.255.255.0
  no shutdown

ft track host GW_251-252
  track-host 192.168.16.251
  peer track-host 192.168.16.252
  probe HA-ICMP priority 10
  peer probe HA-ICMP priority 5
  priority 110
  peer priority 5
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 110
  peer priority 5
ft track interface Int_4/37_6/13_V99
  track-interface vlan 99
  peer track-interface vlan 99
  priority 110
  peer priority 5
ft track host LOCAL-241
  track-host 172.28.4.241
  probe HA-TCP:554 priority 50
  probe HA-TCP:1755 priority 60

domain SH-Gold-Domain
  add-object all

role SHAdmin
  rule 1 permit create
  rule 2 permit monitor
  rule 3 permit modify
role SHUser
  rule 1 deny create
  rule 2 permit monitor
  rule 3 deny modify
  rule 4 permit debug
role TEST

ip route 10.1.0.0 255.255.255.0 172.28.3.254
ip route 172.28.4.0 255.255.255.0 172.28.3.253
ip route 172.28.0.0 255.255.254.0 172.28.3.253
ip route 192.168.16.251 255.255.255.255 192.168.106.251
ip route 192.168.16.252 255.255.255.255 192.168.106.252
username admin password 5 $1$hU1iScF8$WmpdK4IcQI2ofTMDm6l.N1  role Admin domain
default-domain
username localadmin password 5 $1$g5rd5HO2$C34zVe3a9f73Dce/WNvbM.  role Admin domain
SH-Gold-Domain default-domain
username localuser password 5 $1$I21oqX4Q$/OqAKTdBbe8xreKwZtWR3.  role Network-Monitor
domain SH-Gold-Domain
username vrtadmgold password 5 $1$.gNJPJS6$UtozYODAuirfw8XHR1FA8/  role Admin domain
SH-Gold-Domain
username vrtnetmongold password 5 $1$InjySHhu$oLsQV267Nu68q3fH6h7Z4.  role Network-Monitor
domain SH-Gold-Domain
username vrtjohndoe password 5 $1$klNcwN8c$3hTNwFSMtned/9k2a5RPg.  role Admin domain
SH-Gold-Domain

snmp-server community ACE-public group Network-Monitor
snmp-server community ACE-private group Network-Monitor

snmp-server host 10.1.0.236 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-1_ACE2-1/SH-Bridge# changeto Admin
```

```
6K-1_ACE2-1/Admin#
--- 09:57:19 ---

+++ 09:57:19 6K-1_ACE2-1 ctxExec +++
changeto SH-Bridge



NOTE: Configuration mode has been disabled on all sessions


6K-1_ACE2-1/SH-Bridge#


6K-1_ACE2-1/SH-Bridge# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:19 ---

+++ 09:57:19 6K-1_ACE2-1 ctxExec +++
changeto A2


6K-1_ACE2-1/A2#


6K-1_ACE2-1/A2# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:20 ---

+++ 09:57:20 6K-1_ACE2-1 ctxExec +++
changeto A2


6K-1_ACE2-1/A2# show running-config

Generating configuration....


logging enable
logging standby
logging console 6
logging timestamp
logging trap 5
logging history 5
logging buffered 6
logging monitor 5
logging queue 200
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514
logging message 302022 level 7
logging message 302023 level 7
logging message 302024 level 7
logging message 302025 level 7
logging message 302026 level 7
logging message 302027 level 7
```

```
tacacs-server key 7 "vwjjzamggu"
tacacs-server host 172.29.0.235 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.236 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.237 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-gold
  server 172.29.0.236
  server 172.29.0.237

arp 192.168.120.81 00.00.11.11.22.33
arp 172.29.0.201 00.00.22.22.44.55
arp learned-interval 60


crypto authgroup CAUTH_GRP
  cert init.pem
  cert bigcert.pem
  cert SH-ACE-CA.cer
crypto authgroup CAUTH:444_GRP
  cert init.pem
  cert bigcert.pem
  cert SH-ACE-CA.cer
  cert CA-57.cer
crypto authgroup CAUTH_CRP
  cert CA-57.cer
crypto crl SH_ACE_CRL http://sh-ace-ca.cisco.com/CertEnroll/SH-ACE-CA.crl
crypto crl SH_ACE_CRL4 http://sh-ace-ca.cisco.com/CertEnroll/NO-FILE.crl
crypto crl SH_ACE_CRL2 http://10.86.83.127/CertEnroll/SH-ACE-CA-expired.crl
crypto crl SH_ACE_CRL3 http://sh-ace-ca/CertEnroll/CA-57.crl
aaa authentication login error-enable

object-group service OBJ_SERV_1-100
  tcp
  icmp
  igmp
  3
  ip-in-ip
  5
  7
  8
  9
  10
  11
  12
  13
  14
  15
  16
  18
  19
  20
  21
  22
  23
  24
  25
  26
  27
  28
  29
  30
  31
  32
  33
  34
```

```
35
36
37
38
39
40
41
42
43
44
45
46
gre
48
49
esp
ah
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
eigrp
ospf
90
91
92
93
94
95
96
97
98
```

```
    99
    100
    udp
  object-group network OBJ_NET_10
    host 10.1.0.234
    host 10.10.0.2
    host 10.10.0.3
    host 10.10.0.4
    host 10.10.0.5
    host 10.10.0.6
    host 10.10.0.7
    host 10.10.0.8
    host 10.10.0.9
    host 10.10.0.10
  object-group network OBJ_NET_50
    host 10.10.0.51
    host 10.10.0.52
    host 10.10.0.53
    host 10.10.0.54
    host 10.10.0.55
    host 10.10.0.56
    host 10.10.0.57
    host 10.10.0.58
    host 10.10.0.59
    host 10.10.0.60
    host 10.10.0.61
    host 10.10.0.62
    host 10.10.0.63
    host 10.10.0.64
    host 10.10.0.65
    host 10.10.0.66
    host 10.10.0.67
    host 10.10.0.68
    host 10.10.0.69
    host 10.10.0.70
    host 10.10.0.71
    host 10.10.0.72
    host 10.10.0.73
    host 10.10.0.74
    host 10.10.0.75
    host 10.10.0.76
    host 10.10.0.77
    host 10.10.0.78
    host 10.10.0.79
    host 10.10.0.80
    host 10.10.0.81
    host 10.10.0.82
    host 10.10.0.83
    host 10.10.0.84
    host 10.10.0.85
    host 10.10.0.86
    host 10.10.0.87
    host 10.10.0.88
    host 10.10.0.89
    host 10.10.0.90
    host 10.10.0.91
    host 10.10.0.92
    host 10.10.0.93
    host 10.10.0.94
    host 10.10.0.95
    host 10.10.0.96
    host 10.10.0.97
    host 10.10.0.98
    host 10.10.0.99
    host 192.168.140.107

  access-list NAT_ACCESS line 8 extended permit ip any any
  access-list anyone-ip line 10 extended permit ip any any

  script file 1 FTP_PROBE_SCRIPT
  script file 2 TFTP_PROBE
  script file 3 LDAP_PROBE


  probe http FORCED-FAIL
    interval 10
    faildetect 2
    passdetect interval 5
    receive 5
    request method get url /notthere.html
    expect status 200 299
    open 3
  probe ftp FTP
    interval 10
    faildetect 2
    passdetect interval 10
    receive 5
    expect status 220 220
    open 3
  probe http GEN_HTTP
    interval 10
    passdetect interval 2
    passdetect count 2
    expect status 200 200
    header Via header-value "A2_ctx_ACE_CLEAR"
  probe https GEN_HTTPS
    interval 10
    passdetect interval 2
    passdetect count 2
    expect status 200 200
    header Via header-value "A2_ctx_ACE_SSL"
  probe icmp HA-ICMP
    interval 2
    faildetect 2
    passdetect interval 2
  probe tcp HA-TCP:1755
    port 1755
    interval 2
    faildetect 2
    passdetect interval 2
  probe tcp HA-TCP:554
    port 554
    interval 2
    faildetect 2
    passdetect interval 2
  probe http HTTP
    interval 10
    passdetect interval 10
    receive 5
    expect status 200 200
    open 3
  probe icmp ICMP
    interval 10
    faildetect 2
    passdetect interval 10
  probe https SSL
    interval 10
    passdetect interval 7
    passdetect count 2
    expect status 200 200
    header Via header-value "A2_ctx_SSL_Prb"
  probe tcp TCP
    interval 5
    faildetect 2
    passdetect interval 10
    open 3
  probe tcp TCP:443
    port 443
    interval 5
    passdetect interval 10
    connection term forced
    open 3
  probe udp UDP:2222
    port 2222
    interval 5
    passdetect interval 2
  probe udp UDP:53
    interval 5
    passdetect interval 2
  probe icmp WAE_ICMP
    interval 2
    faildetect 1
    passdetect interval 2
    passdetect count 1
  probe snmp linuxCpu
    interval 5
    passdetect interval 2
    passdetect count 2
    community ace-public
    oid 1.3.6.1.4.1.2021.10.1.3.1
      weight 8000
    oid 1.3.6.1.4.1.2021.10.1.3.2
      weight 8000
  probe snmp windowsCpu
    interval 5
    passdetect interval 2
    passdetect count 2
    community ace-public
    oid .1.3.6.1.2.1.25.3.3.1.2.1

  ip domain-lookup
  ip domain-list cisco.com
  ip name-server 172.28.0.152
  ip name-server 161.44.124.122
  ip name-server 10.86.83.121

  parameter-map type ssl CLIENT_SSL
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA
    cipher RSA_WITH_DES_CBC_SHA
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA
    cipher RSA_WITH_AES_256_CBC_SHA
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA
    session-cache timeout 0
    close-protocol disabled
  parameter-map type http HTTP_PARAM
    case-insensitive
    persistence-rebalance
  parameter-map type http HTTP_REBAL_REUSE
    server-conn reuse
    case-insensitive
    persistence-rebalance
  parameter-map type ssl INIT_SSL
    cipher RSA_WITH_RC4_128_MD5 priority 5
    cipher RSA_WITH_RC4_128_SHA
    cipher RSA_WITH_DES_CBC_SHA
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA
    cipher RSA_WITH_AES_256_CBC_SHA priority 10
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA
    session-cache timeout 3660
  parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
    nagle
    slowstart
    set timeout inactivity 30
    tcp-options timestamp allow
    syn-data drop
    exceed-mss allow
    urgent-flag clear
  parameter-map type ssl PARM_ACE_AS_CLIENT
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA
    cipher RSA_WITH_DES_CBC_SHA
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA
    cipher RSA_WITH_AES_256_CBC_SHA
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA
    version TLS1
  parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA priority 2
    cipher RSA_WITH_AES_256_CBC_SHA priority 3
  parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA priority 2
    cipher RSA_WITH_DES_CBC_SHA priority 3
  parameter-map type ssl PARM_ACE_TERM
    cipher RSA_WITH_RC4_128_MD5 priority 6
    version TLS1
  parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA priority 2
    cipher RSA_WITH_AES_256_CBC_SHA priority 3
  parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA priority 2
    cipher RSA_WITH_DES_CBC_SHA priority 3
    version TLS1
    close-protocol disabled
  parameter-map type connection RL_BW
    rate-limit bandwidth 800000
  parameter-map type ssl SSL-INIT
    cipher RSA_WITH_RC4_128_MD5 priority 5
    cipher RSA_WITH_RC4_128_SHA
    cipher RSA_WITH_DES_CBC_SHA
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA
    cipher RSA_WITH_AES_256_CBC_SHA priority 10
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA
    session-cache timeout 3660
  parameter-map type generic SSL-PARSE
    set max-parse-length 60
  parameter-map type ssl SSL_CAUTH
    cipher RSA_WITH_RC4_128_MD5 priority 5
    cipher RSA_WITH_RC4_128_SHA
    cipher RSA_WITH_DES_CBC_SHA
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA
    cipher RSA_WITH_AES_256_CBC_SHA priority 10
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA
    session-cache timeout 3660
  parameter-map type connection TCP_PARAM
    syn-data drop
    exceed-mss allow
  parameter-map type http TEMP_HTTP_REBAL_REUSE
    server-conn reuse
    case-insensitive
    persistence-rebalance
  parameter-map type connection TEMP_TCP_PARAM
    syn-data drop
    exceed-mss allow
  parameter-map type ssl TERM_SSL
    cipher RSA_WITH_RC4_128_MD5 priority 5
    cipher RSA_WITH_RC4_128_SHA
    cipher RSA_WITH_DES_CBC_SHA
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA
    cipher RSA_WITH_AES_256_CBC_SHA priority 10
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA
    session-cache timeout 3660

  action-list type modify http HTTP-HTTPS-Rewrite-Term
    header insert response Cache-Control header-value "max-age=901"
    header insert response SourceIP header-value "%is"
    header insert response DestIP header-value "%id"
    header insert request Pragma header-value "no-cache, no-cache is the only pragma value I
  have ever seen"
    header delete response ETag
    header delete response Set-Cookie
    header rewrite response Keep-Alive header-value "timeout=150" replace "timeout=300"
    header rewrite request User-Agent header-value ".*petescape.*" replace "Mozilla/4.0
  (compatible; Tarantula Client; Runs on Linux)"
    ssl url rewrite location "www.ssl-term-ace.com"
    ssl url rewrite location "192.168.140.103"
  action-list type modify http HTTP-HTTPS-Rewrite-E2E
    header insert response SourceIP header-value "%is"
    header insert response DestIP header-value "%id"
    header insert response Cache-Control header-value "max-age=2592008"
    header insert request Pragma header-value "cache, no-cache is the only pragma value I
  have ever seen but on this one we will use cache"
    header delete response ETag
    header delete response Set-Cookie
    header rewrite request User-Agent header-value ".*petescape.*" replace "Mozilla/5.0
  (compatible; Tarantula Client; Does not run on a MAC)"
    ssl url rewrite location "www.ssl-end-to-end-ace.com" sslport 444
    ssl url rewrite location "192.168.140.105" sslport 444
  action-list type modify http HTTP-HTTPS-Rewrite-Init
    header insert response Cache-Control header-value "max-age=2008"
    header insert response SourceIP header-value "%is"
    header insert response DestIP header-value "%id"
    header delete response ETag
    header delete response Set-Cookie
    header rewrite response Keep-Alive header-value "timeout=150" replace "timeout=300"
    header rewrite request User-Agent header-value ".*petescape.*" replace "Mozilla/4.0
  (compatible; Tarantula Client; Runs on Linux)"

  rserver host BRG-IIS-1
    ip address 172.28.1.26
    inservice
  rserver host BRG-IIS-2
    ip address 172.28.1.27
    inservice
  rserver host BRG-IIS-3
    ip address 172.28.1.28
    inservice
  rserver host BRG-IIS-4
    ip address 172.28.1.29
    inservice
  rserver host BRG-IIS-5
    ip address 172.28.1.30
    inservice
  rserver host BRG-LINUX-1
    ip address 172.28.1.21
    inservice
  rserver host BRG-LINUX-11
    ip address 192.168.120.11
    inservice
  rserver host BRG-LINUX-12
    ip address 192.168.120.12
    inservice
  rserver host BRG-LINUX-13
    ip address 192.168.120.13
    inservice
  rserver host BRG-LINUX-14
    ip address 192.168.120.14
    inservice
  rserver host BRG-LINUX-15
```

```
  ip address 192.168.120.15
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice
rserver host LOCAL-239-FRAGRTR
  ip address 172.29.0.239
  inservice
rserver host LOCAL-IIS-241
  ip address 172.29.0.241
  inservice
rserver host LOCAL-IIS-243
  ip address 172.29.0.243
  inservice
rserver host LOCAL-IIS-245
  ip address 172.29.0.245
  inservice
rserver host LOCAL-LINUX-240
  ip address 172.29.0.240
  inservice
rserver host LOCAL-LINUX-242
  ip address 172.29.0.242
  inservice
rserver host LOCAL-LINUX-244
  ip address 172.29.0.244
  inservice
rserver host RT-IIS-152
  ip address 172.28.0.152
  inservice
rserver host RT-LINUX-151
  ip address 172.28.0.151
  inservice
rserver host RT-LINUX-153
  ip address 172.28.0.153
  inservice
rserver host RT-LINUX-154
  ip address 172.28.0.154
  inservice
rserver host WAE_1
  description WAE 1
  ip address 172.16.0.30
rserver host WAE_2
  description WAE 2
  ip address 172.16.0.31
  inservice

ssl-proxy service ACE_TERM
  ssl advanced-options PARM_ACE_TERM_EXPORT_CIPHERS
ssl-proxy service CAAUTH
ssl-proxy service CAUTH
  key term-wc.key
  cert term-wc.cer
  authgroup CAUTH_GRP
  crl SH_ACE_CRL4
  ssl advanced-options SSL_CAUTH
```

```
ssl-proxy service CAUTH:444
  key term-wc.key
  cert term-wc.cer
  authgroup CAUTH:444_GRP
  crl best-effort
  ssl advanced-options SSL_CAUTH
ssl-proxy service CAUTHL4
  authgroup CAUTH_CRP
  crl SH_ACE_CRL4
ssl-proxy service E2E_SSL
  key pkey.pem
  cert end-to-end.pem
  ssl advanced-options TERM_SSL
ssl-proxy service INIT_E2E_SSL
  ssl advanced-options CLIENT_SSL
ssl-proxy service INIT_SSL
  ssl advanced-options INIT_SSL
ssl-proxy service RL_E2E_SSL
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options TERM_SSL
ssl-proxy service RL_INIT_E2E_SSL
  ssl advanced-options CLIENT_SSL
ssl-proxy service SSL-INIT
  ssl advanced-options SSL-INIT
ssl-proxy service TERM_SSL
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options PARM_ACE_TERM

serverfarm host E2E
  failaction purge
  predictor leastconns
  probe SSL
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-IIS-241 443
    inservice
  rserver LOCAL-IIS-245 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-IIS-152 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host E2E_CHUNK
  failaction purge
  predictor leastconns
  probe TCP:443
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host E2E_POST
  failaction purge
```

```
    predictor leastconns
    probe TCP:443
    retcode 100 599 check count
    rserver BRG-LINUX-1 443
      inservice
    rserver BRG-LINUX-11 443
      inservice
    rserver LOCAL-LINUX-240 443
      inservice
    rserver RT-LINUX-151 443
      inservice
  serverfarm host FTP
    failaction purge
    predictor leastconns
    probe FTP
    rserver BRG-LINUX-11
      inservice
    rserver LOCAL-IIS-241
      inservice
    rserver LOCAL-LINUX-240
      inservice
    rserver RT-IIS-152
      inservice
    rserver RT-LINUX-151
      inservice
  serverfarm host INIT
    failaction purge
    predictor leastconns
    rserver BRG-IIS-1 443
      inservice
    rserver BRG-LINUX-1 443
      inservice
    rserver BRG-LINUX-11 443
      inservice
    rserver LOCAL-IIS-241 443
      inservice
    rserver LOCAL-IIS-245 443
      inservice
    rserver LOCAL-LINUX-240 443
      inservice
    rserver RT-IIS-152 443
      inservice
    rserver RT-LINUX-151 443
      inservice
  serverfarm host INIT2
    failaction purge
    predictor leastconns
    probe TCP
    rserver BRG-IIS-1 443
      inservice
    rserver BRG-LINUX-1 443
      inservice
    rserver BRG-LINUX-11 443
      inservice
    rserver LOCAL-IIS-241 443
      inservice
    rserver LOCAL-IIS-245 443
      inservice
    rserver LOCAL-LINUX-240 443
      inservice
    rserver RT-IIS-152 443
      inservice
    rserver RT-LINUX-151 443
      inservice
```

```
serverfarm host L4
  failaction purge
  probe HTTP
  probe SSL
  rserver BRG-LINUX-11
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
serverfarm host L7
  failaction purge
  predictor leastconns
  probe TCP
  retcode 100 599 check count
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host LEASTLOAD
  predictor least-loaded probe linuxCpu
    weight connection
    autoadjust average
  rserver LOCAL-LINUX-240
    inservice
  rserver LOCAL-LINUX-242
    inservice
serverfarm host LEASTLOAD-2
  predictor least-loaded probe windowsCpu
    weight connection
    autoadjust average
  rserver LOCAL-IIS-243
    inservice
  rserver LOCAL-IIS-245
    inservice
serverfarm host ORIGIN-SF-WAAS
  description SERVER SF FOR WAAS RETURN TRAFFIC
  predictor leastconns
  probe TCP
  rserver BRG-IIS-1
    conn-limit max 6500 min 5000
    inservice
  rserver BRG-LINUX-1
    conn-limit max 6500 min 5000
    inservice
serverfarm host PREDICTOR
  failaction purge
  predictor least-bandwidth samples 2 assess-time 1
  probe HTTP
  rserver BRG-IIS-1
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-11
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-IIS-245
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
serverfarm host RL_SSL
  failaction purge
  probe SSL
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
serverfarm host RL_WWW
  failaction purge
  probe HTTP
  retcode 100 599 check count
  rserver BRG-LINUX-11
    inservice
  rserver LOCAL-IIS-241
    inservice
serverfarm host SSL-SESSION
  failaction purge
  predictor leastconns
  probe TCP:443
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-11
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-IIS-245
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
serverfarm host SSL_CAUTH
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host SSL_CAUTH2
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host TERM
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 81
    inservice
  rserver BRG-LINUX-11 81
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 81
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 81
    inservice
serverfarm host TERM2
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host WAAS
  description WAAS SF TRANSPARENT MODE
  transparent
  predictor leastconns
  probe WAE_ICMP
  rserver WAE_1
    conn-limit max 6500 min 5000
    inservice
  rserver WAE_2
    conn-limit max 6500 min 5000
    inservice

sticky http-cookie TERM_SSL STKY-CKY_TERM
  cookie insert
  timeout 30
  replicate sticky
  serverfarm TERM
sticky http-cookie INIT_SSL STKY-CKY_INIT
  cookie insert
  timeout 30
  replicate sticky
  serverfarm INIT
sticky http-cookie E2E_SSL STKY-CKY_E2E_ANY
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E
sticky http-cookie E2E_SSL STKY-CKY_E2E_POST
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E_POST
sticky http-cookie E2E_SSL STKY-CKY_E2E_CHUNK
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E_CHUNK
sticky layer4-payload SSL-SESSION_STKY
  timeout 600
  replicate sticky
  serverfarm SSL-SESSION
  response sticky
  layer4-payload offset 43 length 32 begin-pattern "\x20"
sticky http-cookie SSL_CAUTH STKY-CKY_CAUTH
  cookie insert
  timeout 30
  replicate sticky
  serverfarm SSL_CAUTH
sticky ip-netmask 255.255.255.255 address source STKY-SRCIP-GRP30
  timeout 30
  replicate sticky
  serverfarm RL_SSL
sticky http-cookie RL_E2E_SSL RL-COOKIE
  cookie insert
  timeout 30
  replicate sticky
  serverfarm RL_SSL

class-map type http inspect match-all CHUNK-HDR_INSPECT
  2 match header Transfer-Encoding header-value "chunked"
class-map type http inspect match-all CHUNK-XFR_INSPECT
  2 match transfer-encoding chunked
class-map type http loadbalance match-all CHUNK.PL_URL
  2 match http url .*.chunk.pl
class-map type http loadbalance match-all CHUNK_HDR
  2 match http header Transfer-Encoding header-value "chunked"
class-map type http loadbalance match-any CHUNK_POST.PL_URL
  2 match http url .*.chunk.pl
  3 match http url .*cgipostform.pl
class-map match-all E2E-VIP_105:443
  2 match virtual-address 192.168.140.105 tcp eq https
class-map match-all E2E-VIP_105:444
  2 match virtual-address 192.168.140.105 tcp eq 444
class-map match-all FTP-VIP_107
  2 match virtual-address 192.168.140.107 tcp eq ftp
class-map match-all HEADER-INSERT-VIP_121:80
  2 match virtual-address 192.168.120.121 tcp eq www
class-map match-all INIT-VIP_104:80
  2 match virtual-address 192.168.140.104 tcp eq www
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  10 match url .*
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
  31 match request-method rfc get
class-map match-all L4-VIP_107:NAT
  2 match destination-address 192.168.140.107 255.255.255.255
class-map match-all L4-VIP_107:TCP_80-443
  2 match virtual-address 192.168.140.107 tcp range 80 443
class-map match-all L4-VIP_107:UDP_ANY
  2 match virtual-address 192.168.140.107 udp any
class-map match-all L7-VIP_110:80
  2 match virtual-address 192.168.140.110 tcp eq www
class-map match-all LL_VIP_111:80
  2 match virtual-address 192.168.140.111 tcp eq www
class-map match-all LL_VIP_112:80
  2 match virtual-address 192.168.140.112 tcp eq www
```

```
class-map type management match-any MGT
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
class-map type http loadbalance match-any POST.PL_URL
  2 match http url .*cgipostform.pl
class-map type http inspect match-all POST_INSPECT
  2 match request-method rfc post
class-map match-all PRED-VIP_108:80
  2 match virtual-address 192.168.140.108 tcp eq www
class-map type management match-any REMOTE
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol https any
class-map match-all RL_E2E-VIP_109:443
  2 match virtual-address 192.168.140.109 tcp eq https
class-map match-all RL_WWW-VIP_109
  2 match virtual-address 192.168.140.109 tcp eq www
class-map match-any SSL-SESSION-VIP_106:443
  2 match virtual-address 192.168.140.106 tcp eq https
class-map type generic match-any SSLID-32_REGEX
  2 match layer4-payload regex "\x16\x03[\x00\x01]..[\x01\x02].*"
  3 match layer4-payload regex "\x80\x4c.*"
class-map match-all SSL_CAUTH-VIP_102:443
  2 match virtual-address 192.168.140.102 tcp eq https
class-map match-all SSL_CAUTH-VIP_102:444
  2 match virtual-address 192.168.140.102 tcp eq 444
class-map match-all TERM-VIP_103:443
  2 match virtual-address 192.168.140.103 tcp eq https
class-map type http loadbalance match-all URL*_L7
  2 match http url .*
class-map type http loadbalance match-all URL_*.GIF
  2 match http url .*.gif
class-map type http loadbalance match-all URL_*.JPG
  2 match http url .*.jpg
class-map match-any WAAS-TO-ACE-VIP_100:ANY
  2 match virtual-address 192.168.140.100 tcp any
class-map match-any WAAS-VIP_100:80
  2 match virtual-address 192.168.140.100 tcp eq www

policy-map type management first-match P-MGT
  class MGT
    permit

policy-map type loadbalance first-match L3
  class class-default
    serverfarm SSL-SESSION
policy-map type loadbalance first-match PLBSF_FTP
  class class-default
    serverfarm FTP
policy-map type loadbalance first-match PLBSF_INIT
  class URL_*.GIF
    serverfarm INIT2
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_INIT.GIF"
    ssl-proxy client INIT_SSL
  class URL_*.JPG
```

```
    serverfarm INIT2
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_INIT.JPG"
    ssl-proxy client INIT_SSL
  class URL*_L7
    sticky-serverfarm STKY-CKY_INIT
    action HTTP-HTTPS-Rewrite-Init
    insert-http I_AM header-value "SSL_INIT_COOKIE_INS"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client INIT_SSL
policy-map type loadbalance first-match PLBSF_L4
  class class-default
    serverfarm L4
policy-map type loadbalance first-match PLBSF_L7
  class URL*_L7
    serverfarm L7
    insert-http I_AM header-value "SF_L7"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match PLBSF_LL
  class class-default
    serverfarm LEASTLOAD
policy-map type loadbalance first-match PLBSF_LL-2
  class class-default
    serverfarm LEASTLOAD-2
policy-map type loadbalance first-match PLBSF_LL_E2E
  class POST.PL_URL
    sticky-serverfarm STKY-CKY_E2E_POST
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "E2E_SSL_POST"
    ssl-proxy client INIT_E2E_SSL
  class CHUNK.PL_URL
    sticky-serverfarm STKY-CKY_E2E_CHUNK
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "E2E_SSL_CHUNK_XFER"
    ssl-proxy client INIT_E2E_SSL
  class class-default
    sticky-serverfarm STKY-CKY_E2E_ANY
    action HTTP-HTTPS-Rewrite-E2E
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "E2E_SSL_ANY"
    ssl-proxy client INIT_E2E_SSL
policy-map type loadbalance first-match PLBSF_ORIGIN-SF-WAAS
  class class-default
    serverfarm ORIGIN-SF-WAAS
policy-map type loadbalance first-match PLBSF_PREDICTOR
  class URL*_L7
    serverfarm PREDICTOR
    insert-http I_AM header-value "SF_PREDICTOR"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match PLBSF_RL_SSL
  class URL*_L7
    sticky-serverfarm RL-COOKIE
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "Rate_Limit_E2E_SSL"
    ssl-proxy client RL_INIT_E2E_SSL
policy-map type loadbalance first-match PLBSF_RL_WWW
```

```
  class class-default
    serverfarm RL_WWW
policy-map type loadbalance first-match PLBSF_SSL_CAUTH
  class URL_*.GIF
    serverfarm SSL_CAUTH2
    insert-http I_AM header-value "SSL_CAUTH.GIF"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
  class URL_*.JPG
    serverfarm SSL_CAUTH2
    insert-http I_AM header-value "SSL_CAUTH.JPG"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
  class URL*_L7
    sticky-serverfarm STKY-CKY_CAUTH
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_CAUTH_COOKIE_INS"
policy-map type loadbalance first-match PLBSF_TERM
  class URL_*.GIF
    serverfarm TERM2
    action HTTP-HTTPS-Rewrite-Term
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM.GIF"
  class URL_*.JPG
    serverfarm TERM2
    action HTTP-HTTPS-Rewrite-Term
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM.JPG"
  class URL*_L7
    sticky-serverfarm STKY-CKY_TERM
    action HTTP-HTTPS-Rewrite-Term
    insert-http I_AM header-value "SSL_TERM_COOKIE_INS"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match PLBSF_WAE
  class class-default
    serverfarm WAAS backup ORIGIN-SF-WAAS

policy-map type loadbalance generic first-match PLBSF_SSL-SESSION
  class SSLID-32_REGEX
    sticky-serverfarm SSL-SESSION_STKY

policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit

policy-map multi-match A2-VIPS
  class E2E-VIP_105:443
    loadbalance vip inservice
    loadbalance policy PLBSF_LL_E2E
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_REBAL_REUSE
    ssl-proxy server E2E_SSL
    connection advanced-options TCP_PARAM
  class SSL_CAUTH-VIP_102:443
    loadbalance vip inservice
    loadbalance policy PLBSF_SSL_CAUTH
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server CAUTH
    connection advanced-options TCP_PARAM
  class L4-VIP_107:NAT
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
  class L4-VIP_107:TCP_80-443
    loadbalance vip inservice
    loadbalance policy PLBSF_L4
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class PRED-VIP_108:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PREDICTOR
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options HTTP_REBAL_REUSE
  class RL_E2E-VIP_109:443
    loadbalance vip inservice
    loadbalance policy PLBSF_RL_SSL
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_REBAL_REUSE
    ssl-proxy server RL_E2E_SSL
    connection advanced-options TCP_PARAM
  class TERM-VIP_103:443
    loadbalance vip inservice
    loadbalance policy PLBSF_TERM
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server TERM_SSL
    connection advanced-options TCP_PARAM
  class INIT-VIP_104:80
    loadbalance vip inservice
    loadbalance policy PLBSF_INIT
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options HTTP_PARAM
  class L7-VIP_110:80
    loadbalance vip inservice
    loadbalance policy PLBSF_L7
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
  class SSL-SESSION-VIP_106:443
    loadbalance vip inservice
    loadbalance policy PLBSF_SSL-SESSION
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    appl-parameter generic advanced-options SSL-PARSE
  class SSL_CAUTH-VIP_102:444
    loadbalance vip inservice
    loadbalance policy PLBSF_SSL_CAUTH
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server CAUTH:444
    connection advanced-options TCP_PARAM
  class FTP-VIP_107
    loadbalance vip inservice
    loadbalance policy PLBSF_FTP
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    inspect ftp
  class E2E-VIP_105:444
    loadbalance vip inservice
    loadbalance policy PLBSF_LL_E2E
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_REBAL_REUSE
    ssl-proxy server E2E_SSL
    connection advanced-options TCP_PARAM
  class L4-VIP_107:UDP_ANY
    loadbalance vip inservice
    loadbalance policy PLBSF_L4
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
  class LL_VIP_111:80
    loadbalance vip inservice
    loadbalance policy PLBSF_LL
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
  class LL_VIP_112:80
    loadbalance vip inservice
    loadbalance policy PLBSF_LL-2
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
  class RL_WWW-VIP_109
    loadbalance vip inservice
    loadbalance policy PLBSF_RL_WWW
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
policy-map multi-match CLIENTS_TO_WAAS
  class WAAS-VIP_100:80
    loadbalance vip inservice
    loadbalance policy PLBSF_WAE
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
policy-map multi-match WAAS-TO-ACE
  class WAAS-TO-ACE-VIP_100:ANY
    loadbalance vip inservice
    loadbalance policy PLBSF_ORIGIN-SF-WAAS
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 29

service-policy input P-MGT

interface vlan 29
  description SERVER VLAN-29
  ip address 172.29.0.5 255.255.255.0
  alias 172.29.0.4 255.255.255.0
  peer ip address 172.29.0.6 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.106.75 192.168.106.75 netmask 255.255.255.0 pat
  no shutdown
interface vlan 83
  ip address 10.86.83.209 255.255.255.0
  peer ip address 10.86.83.212 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 99
  ip address 192.168.99.9 255.255.255.0
  peer ip address 192.168.99.8 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 106
  description CLIENT VLAN-106
  ip address 192.168.106.5 255.255.255.0
  alias 192.168.106.4 255.255.255.0
  peer ip address 192.168.106.6 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  no icmp-guard
  access-group input anyone-ip
  nat-pool 1 192.168.106.70 192.168.106.70 netmask 255.255.255.0 pat
  service-policy input CLIENTS_TO_WAAS
  service-policy input A2-VIPS
  no shutdown
interface vlan 120
  description CLIENT VLAN-120
  ip address 192.168.120.5 255.255.255.0
  alias 192.168.120.4 255.255.255.0
  peer ip address 192.168.120.6 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.80 192.168.120.80 netmask 255.255.255.0 pat
  no shutdown
interface vlan 160
  description WAE VLAN-160
  ip address 172.16.0.2 255.255.255.0
  alias 172.16.0.1 255.255.255.0
  peer ip address 172.16.0.3 255.255.255.0
  no normalization
  fragment chain 20
  fragment min-mtu 68
  no icmp-guard
  access-group input anyone-ip
  nat-pool 1 192.168.106.85 192.168.106.85 netmask 255.255.255.0 pat
  service-policy input WAAS-TO-ACE
  no shutdown

ft track host GW_251-252
  track-host 192.168.16.251
  peer track-host 192.168.16.252
  probe HA-ICMP priority 10
  peer probe HA-ICMP priority 5
  priority 110
  peer priority 5
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 110
  peer priority 5
ft track interface Int_4/37_6/13_V99
  track-interface vlan 99
  peer track-interface vlan 99
  priority 110
ft track host RT-241
  track-host 172.29.0.241
  probe HA-TCP:1755 priority 60
  probe HA-TCP:554 priority 50

domain A2
  add-object all

role SHAdmin
  rule 1 permit create
  rule 2 permit monitor
  rule 3 permit modify
role SHUser
  rule 1 deny create
  rule 2 permit monitor
  rule 3 deny modify
  rule 4 permit debug

ip route 172.28.0.0 255.254.0.0 172.29.0.253
ip route 192.168.16.251 255.255.255.255 192.168.106.251
ip route 192.168.16.252 255.255.255.255 192.168.106.252
ip route 10.1.0.0 255.255.255.0 192.168.106.254
ip route 10.3.0.0 255.255.255.0 192.168.106.254
ip route 0.0.0.0 0.0.0.0 192.168.106.254
username admin password 5 $1$hU1iScF8$WmpdK4IcQI2ofTMDm6l.N1  role Admin domain
default-domain
username localadmin password 5 $1$g5rd5HO2$C34zVe3a9f73Dce/WNvbM.  role Admin domain
default-domain
username localuser password 5 $1$I21oqX4Q$/OqAKTdBbe8xreKwZtWR3.  role Network-Monitor
domain default-domain
```

```
username vrtadma2 password 5 $1$.gNJPJS6$UtozYODAuirfw8XHR1FA8/  role Admin domain
default-domain
username vrtnetmongold password 5 $1$InjySHhu$oLsQV267Nu68q3fH6h7Z4.  role Network-Monitor
domain default-domain
username vrtjohndoe password 5 $1$klNcwN8c$3hTNwFSMtned/9k2a5RPg.  role Admin domain
default-domain
username jw password 5 $1$yemV0ad1$4oqyG73BYLG3Q4yGk6Udq.  role Admin domain
default-domain

snmp-server community ACE-public group Network-Monitor
snmp-server community ACE-private group Network-Monitor

snmp-server host 10.1.0.236 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-1_ACE2-1/A2# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:20 ---

+++ 09:57:20 6K-1_ACE2-1 ctxExec +++
changeto A2


6K-1_ACE2-1/A2#


6K-1_ACE2-1/A2# changeto Admin


6K-1_ACE2-1/Admin#
--- 09:57:20 ---

+++ 09:57:26 6K-1_ACE2-1 ctxExec +++
term width 512


6K-1_ACE2-1/Admin#
--- 09:57:26 ---

+++ 09:57:26 6K-1_ACE2-1 ctxExec +++
term length 0


6K-1_ACE2-1/Admin#
--- 09:57:26 ---

+++ 09:57:26 6K-1_ACE2-1 ctxConfig +++
config terminal

Enter configuration commands, one per line.  End with CNTL/Z.

6K-1_ACE2-1/Admin(config)# logging console 5


6K-1_ACE2-1/Admin(config)# end
```

```
6K-1_ACE2-1/Admin#
--- 09:57:27 ---

+++ 09:57:27 6K-1_ACE2-1 destroy +++
```

Return to 6K-1 ACE1 All Context, page 298

6K-2 ACE1 All Context

Go to Basic Topology: Test Device Configurations, page 212

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11

```
--- 09:57:12 ---

+++ 09:57:12 6K-2_ACE2-1 logging +++

--- 09:57:12 ---

+++ 09:57:12 6K-2_ACE2-1 ctxExec +++
show context


Number of Contexts = 10

Name: Admin , Id: 0
Config count: 302
Description:
Resource-class: default
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: 130-VRT-1 , Id: 1
Config count: 672
Description:
Resource-class: POINT_FIVE
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: 130-VRT-2 , Id: 2
Config count: 482
Description:
Resource-class: POINT_FIVE
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: 130-VRT-3 , Id: 3
Config count: 676
Description:
```

```
Resource-class: POINT_FIVE
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: A2 , Id: 4
Config count: 3036
Description:
Resource-class: 15_PERCENT
Vlans:  Vlan29, Vlan83, Vlan99, Vlan106, Vlan120, Vlan160
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: LARGE , Id: 5
Config count: 1
Description:
Resource-class: default
Vlans:  Vlan83, Vlan130, Vlan281, Vlan320, Vlan329
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: SH-Bridge , Id: 6
Config count: 2694
Description:
Resource-class: 10_PERCENT
Vlans:  Vlan99, Vlan106, Vlan283, Vlan2830
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: enabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: enabled


Name: SH-Gold , Id: 7
Config count: 4287
Description:
Resource-class: 15_PERCENT
Vlans:  Vlan29, Vlan99, Vlan105, Vlan120
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: SH-LOAD , Id: 8
Config count: 847
Description:
Resource-class: 5_PERCENT
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


Name: SH-Silver , Id: 9
```

```
Config count: 677
Description:
Resource-class: 10_PERCENT
Vlans:  Vlan130, Vlan281
FT Auto-sync running-cfg configured state: enabled
FT Auto-sync running-cfg actual state: disabled
FT Auto-sync startup-cfg configured state: enabled
FT Auto-sync startup-cfg actual state: disabled


6K-2_ACE2-1/Admin#
--- 09:57:12 ---

+++ 09:57:20 6K-2_ACE2-1 ctxExec +++



6K-2_ACE2-1/Admin#
--- 09:57:20 ---

+++ 09:57:20 6K-2_ACE2-1 ctxExec +++
show running-config

Generating configuration....


logging enable
logging standby
logging timestamp
logging trap 5
logging buffered 5
logging monitor 5
logging queue 150
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514

ldap-server host 172.29.0.244
aaa group server ldap SH_LDAP
  baseDN "dc=bxb-safeharbor,dc=com"

peer hostname 6K-1_ACE2-1

login timeout 0
hostname 6K-2_ACE2-1
boot system image:c6ace-t1k9-mz.A2_1_2.bin
shared-vlan-hostid 15
peer shared-vlan-hostid 16

resource-class 10_PERCENT
  limit-resource all minimum 10.00 maximum unlimited
  limit-resource buffer syslog minimum 10.00 maximum equal-to-min
  limit-resource sticky minimum 10.00 maximum equal-to-min
resource-class 15_PERCENT
  limit-resource all minimum 15.00 maximum unlimited
  limit-resource buffer syslog minimum 15.00 maximum equal-to-min
  limit-resource sticky minimum 15.00 maximum equal-to-min
resource-class 1PERCENT
  limit-resource all minimum 1.00 maximum equal-to-min
  limit-resource sticky minimum 1.00 maximum equal-to-min
resource-class 5_PERCENT
  limit-resource all minimum 5.00 maximum unlimited
  limit-resource buffer syslog minimum 5.00 maximum equal-to-min
  limit-resource sticky minimum 5.00 maximum equal-to-min
```

```
resource-class JUST_STICKY
  limit-resource all minimum 0.00 maximum unlimited
  limit-resource sticky minimum 1.00 maximum equal-to-min
resource-class MIN-ALL
  limit-resource all minimum 0.01 maximum equal-to-min
  limit-resource mgmt-connections minimum 1.00 maximum equal-to-min
  limit-resource rate syslog minimum 1.00 maximum equal-to-min
  limit-resource sticky minimum 0.01 maximum equal-to-min
resource-class OVER-LIMIT
  limit-resource all minimum 90.00 maximum equal-to-min
resource-class POINT_FIVE
  limit-resource all minimum 0.50 maximum equal-to-min
tacacs-server key 7 "vwjjzamggu"
tacacs-server host 10.86.83.215 key 7 "vwjjzamggu"
aaa group server tacacs+ SafeHarbor-Tacacs
  server 10.86.83.215
aaa group server tacacs+ sh-admin
  server 10.86.83.215

clock timezone standard EST
clock summer-time standard EDT
aaa authentication login default group sh-admin local
aaa accounting default group sh-admin
aaa authentication login error-enable

access-list anyone line 10 extended permit ip any any



class-map type management match-any REMOTE-ACCESS_ALL
  description "Remote Management for ALL"
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
class-map type management match-any REMOTE-ACCESS_LOCAL
  description "Remote Management for BXB only"
  2 match protocol telnet source-address 10.80.0.0 255.248.0.0
  4 match protocol ssh source-address 10.80.0.0 255.248.0.0
  5 match protocol icmp source-address 10.80.0.0 255.248.0.0
  6 match protocol http source-address 10.80.0.0 255.248.0.0
  7 match protocol snmp source-address 10.80.0.0 255.248.0.0
  8 match protocol https source-address 10.80.0.0 255.248.0.0
  9 match protocol ssh source-address 161.44.64.0 255.255.252.0
  10 match protocol icmp source-address 161.44.64.0 255.255.252.0
  11 match protocol http source-address 161.44.64.0 255.255.252.0
  12 match protocol snmp source-address 161.44.64.0 255.255.252.0
  13 match protocol https source-address 161.44.64.0 255.255.252.0
  14 match protocol telnet source-address 161.44.64.0 255.255.252.0

policy-map type management first-match REMOTE-MGMT
  class REMOTE-ACCESS_ALL
    permit

interface vlan 83
  ip address 10.86.83.161 255.255.255.0
  peer ip address 10.86.83.160 255.255.255.0
  service-policy input REMOTE-MGMT
  no shutdown

ft interface vlan 900
  ip address 192.168.1.2 255.255.255.0
  peer ip address 192.168.1.1 255.255.255.0
  no shutdown

ft peer 1
  heartbeat interval 300
  heartbeat count 10
  ft-interface vlan 900
ft group 1
  peer 1
  peer priority 200
  associate-context Admin
  inservice

domain SH-Admin-Domain
  add-object all
domain art
  add-object all

role CiscoRole
  rule 1 permit modify
role SH-Role
  rule 1 permit create
  rule 2 permit modify

ip route 0.0.0.0 0.0.0.0 10.86.83.1

context 130-VRT-1
  allocate-interface vlan 130
  allocate-interface vlan 281
  member POINT_FIVE
context 130-VRT-2
  allocate-interface vlan 130
  allocate-interface vlan 281
  member POINT_FIVE
context 130-VRT-3
  allocate-interface vlan 130
  allocate-interface vlan 281
  member POINT_FIVE
context A2
  allocate-interface vlan 29
  allocate-interface vlan 83
  allocate-interface vlan 99
  allocate-interface vlan 106
  allocate-interface vlan 120
  allocate-interface vlan 160
  member 15_PERCENT
context LARGE
  allocate-interface vlan 83
  allocate-interface vlan 130
  allocate-interface vlan 281
  allocate-interface vlan 320
  allocate-interface vlan 329
context SH-Bridge
  allocate-interface vlan 99
  allocate-interface vlan 106
  allocate-interface vlan 283
  allocate-interface vlan 2830
  member 10_PERCENT
context SH-Gold
  allocate-interface vlan 29
  allocate-interface vlan 99
  allocate-interface vlan 105
  allocate-interface vlan 120
  member 15_PERCENT
```

```
context SH-LOAD
  allocate-interface vlan 130
  allocate-interface vlan 281
  member 5_PERCENT
context SH-Silver
  allocate-interface vlan 130
  allocate-interface vlan 281
  member 10_PERCENT

snmp-server community ACE-private group Network-Monitor
snmp-server community ACE-public group Network-Monitor

snmp-server host 10.86.83.236 traps version 2c ACE-public

snmp-server enable traps snmp coldstart
snmp-server enable traps virtual-context
snmp-server enable traps license
snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown

ft group 2
  peer 1
  peer priority 200
  associate-context SH-Gold
  inservice
ft group 3
  peer 1
  peer priority 200
  associate-context SH-Silver
  inservice
ft group 4
  peer 1
  peer priority 200
  associate-context SH-LOAD
  inservice
ft group 5
  peer 1
  peer priority 200
  associate-context LARGE
  inservice
ft group 6
  peer 1
  peer priority 200
  associate-context 130-VRT-1
  inservice
ft group 7
  peer 1
  peer priority 200
  associate-context 130-VRT-2
  inservice
ft group 8
  peer 1
  peer priority 200
  associate-context 130-VRT-3
  inservice
ft group 10
  peer 1
  peer priority 200
  associate-context SH-Bridge
  inservice
```

```
ft group 11
  peer 1
  peer priority 200
  associate-context A2
  inservice
username admin password 5 $1$YBxN8VJN$KqLLEtXDnrXm6M9vozWI40  role Admin domain
default-domain
username www password 5 $1$UZIiwUk7$QMVYN1JASaycabrHkhGcS/  role Admin domain
default-domain
username netmon password 5 $1$rYHZjgGm$iWMZNUv/9oHEUsPxZM.tq.  role Network-Monitor domain
default-domain
username sh password 5 $1$NZk8T1Xf$uhYawy7j7m9Q2/EUBx5Ti/  role Admin domain
default-domain
username art password 5 $1$pzEPn3Gk$tuSFSxtxQuuB3O6NzIGLr0  role Admin domain
default-domain
username bxb-safeharbor password 5 $1$IvCJDwOx$eanTW5mtGQYWB47SPhqZx.  role
Network-Monitor domain default-domain
username root password 5 $1$mbLJQdlS$mQAerSjxe0E4qxlx8VKO4.  role Admin domain
default-domain
ssh key rsa 4096 force
ssh key rsa1 4096 force


6K-2_ACE2-1/Admin#
--- 09:57:20 ---

+++ 09:57:20 6K-2_ACE2-1 ctxExec +++



6K-2_ACE2-1/Admin#
--- 09:57:21 ---

+++ 09:57:21 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-1



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-1#


6K-2_ACE2-1/130-VRT-1# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:21 ---

+++ 09:57:21 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-1



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-1# show running-config
```

```
Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging buffered 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7



crypto chaingroup CHAIN1
aaa authentication login error-enable

access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-1_probe_GEN-HTTP"
  connection term forced
probe https GEN_HTTPS
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-1_probe_GEN-HTTPS"
  connection term forced


parameter-map type http HTTP_PARAM
  server-conn reuse
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 20
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version TLS1
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  nagle
  syn-data drop
  exceed-mss allow
  urgent-flag clear

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
```

```
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  ssl advanced-options PARM_ACE_TERM

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
```

```
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host ONE-IIS-SERVER
  rserver BRG-IIS-1
    inservice
serverfarm host TCP-NORM-FARM
  predictor leastconns
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-IIS-3
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver BRG-LINUX-3
    inservice
serverfarm host TELNET-NORM-TEST
  rserver BRG-LINUX-1
    inservice

sticky http-cookie SSL_TERM_COOKIE GROUP_10
  cookie insert browser-expire
  serverfarm ACE_TERM_SERVERS_CLEAR
sticky http-cookie SSL_INIT_COOKIE GROUP_20
  cookie insert browser-expire
  serverfarm ACE_INIT_SERVERS_SSL
sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
  cookie insert browser-expire
  serverfarm ACE_END_TO_END_SERVERS_SSL
sticky http-cookie NON_SSL_TESTING GROUP_40
  cookie insert browser-expire
  serverfarm NON-SSL-TEST
sticky ip-netmask 255.255.255.0 address both NEW_GROUP
  serverfarm NON-SSL-TEST

class-map match-all GENERIC
class-map type http loadbalance match-all LB_CLASS_HTTP
  2 match http url .*
class-map match-all NON-SSL_TEST
  description NON-SSL_TEST
class-map match-all SSL_END_TO_END_13
  description STICKY FOR SSL TESTING
  3 match source-address 192.168.130.0 255.255.255.0
class-map match-all TCP-NORM-TEST
  description TCP NORM TEST
  2 match virtual-address 192.168.130.137 tcp eq 22

policy-map type management first-match POLICY_MGMT

policy-map type loadbalance first-match GENERIC
  class LB_CLASS_HTTP
    serverfarm ONE-IIS-SERVER
  class class-default
    serverfarm ONE-IIS-SERVER
policy-map type loadbalance first-match NON_SSL_TESTING
  class LB_CLASS_HTTP
    serverfarm NON-SSL-TEST
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http I_AM header-value "NON_SSL"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
  class LB_CLASS_HTTP
    serverfarm ACE_END_TO_END_SERVERS_SSL
    insert-http I_AM header-value "SSL_END_TO_END"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_INIT
  class LB_CLASS_HTTP
    serverfarm ACE_INIT_SERVERS_SSL
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http I_AM header-value "SSL_INIT"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_TERM
  class LB_CLASS_HTTP
    serverfarm ACE_TERM_SERVERS_CLEAR
    insert-http SRC_IP header-value "is"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM"
policy-map type loadbalance first-match TCP-NORM-TESTING
  class LB_CLASS_HTTP
    serverfarm TCP-NORM-FARM
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "TCP-NORM-TEST"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match TELNET-NORM_TESTING
  class class-default
    serverfarm TELNET-NORM-TEST

policy-map type inspect http all-match INSPECT_GOOD_HTTP

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_END_TO_END_13
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class TCP-NORM-TEST
    loadbalance vip inservice
    loadbalance policy TCP-NORM-TESTING
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
  class GENERIC
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM

interface vlan 130
  ip address 192.168.130.33 255.255.255.0
  alias 192.168.130.31 255.255.255.0
  peer ip address 192.168.130.32 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.74 192.168.130.74 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.15 255.255.255.0
  alias 172.28.1.13 255.255.255.0
  peer ip address 172.28.1.14 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.75 192.168.130.75 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown

ip route 10.1.0.0 255.255.255.0 192.168.130.254
ip route 192.168.120.0 255.255.255.0 192.168.130.254
username admin password 5 $1$Dyc2pgwi$j.ib5ZiBo.7Xm7J7kEmrW/  role Admin domain
default-domain
username netmon password 5 $1$G6bSxh54$7Jop/aBTWNrdN2iS18JX2.  role Admin domain
default-domain




6K-2_ACE2-1/130-VRT-1# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:21 ---

+++ 09:57:21 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-1



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-1#


6K-2_ACE2-1/130-VRT-1# changeto Admin



NOTE: Configuration mode has been disabled on all sessions
```

```
6K-2_ACE2-1/Admin#
--- 09:57:21 ---

+++ 09:57:21 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-2



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-2#


6K-2_ACE2-1/130-VRT-2# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:21 ---

+++ 09:57:21 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-2



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-2# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging buffered 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7



crypto chaingroup CHAIN1
aaa authentication login error-enable

access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-2_probe_GEN-HTTP"
  connection term forced
probe https GEN_HTTPS
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 407
  header Via header-value "130-VRT-2_probe_GEN-HTTPS"
  connection term forced


parameter-map type http HTTP_PARAM
  server-conn reuse
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 20
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version TLS1
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  urgent-flag clear
parameter-map type connection TEST

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  ssl advanced-options PARM_ACE_TERM

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
```

```
    failaction purge
    probe GEN_HTTPS
    rserver BRG-IIS-1 443
      inservice
    rserver BRG-IIS-2 443
      inservice
    rserver BRG-IIS-3 443
      inservice
    rserver BRG-LINUX-1 443
      inservice
    rserver BRG-LINUX-2 443
      inservice
    rserver BRG-LINUX-3 443
      inservice
  serverfarm host ACE_TERM_SERVERS_CLEAR
    description SERVERS FOR SSL TERM TESTING
    probe GEN_HTTP
    rserver BRG-IIS-1 80
      inservice
    rserver BRG-IIS-2 80
      inservice
    rserver BRG-IIS-3 80
      inservice
    rserver BRG-LINUX-1 80
      inservice
    rserver BRG-LINUX-2 80
      inservice
    rserver BRG-LINUX-3 80
      inservice
  serverfarm host NON-SSL-TEST
    description SERVERS FOR NON SSL TESTING
    failaction purge
    probe GEN_HTTP
    rserver BRG-IIS-1 80
      inservice
    rserver BRG-IIS-2 80
      inservice
    rserver BRG-IIS-3 80
      inservice
    rserver BRG-LINUX-1 80
      inservice
    rserver BRG-LINUX-2 80
      inservice
    rserver BRG-LINUX-3 80
      inservice
  serverfarm host ONE-IIS-SERVER
    rserver BRG-IIS-1
      inservice
  serverfarm host TCP-NORM-FARM
    predictor leastconns
    rserver BRG-IIS-1
      inservice
    rserver BRG-IIS-2
      inservice
    rserver BRG-IIS-3
      inservice
    rserver BRG-LINUX-1
      inservice
    rserver BRG-LINUX-2
      inservice
    rserver BRG-LINUX-3
      inservice
  serverfarm host TELNET-NORM-TEST
    rserver BRG-LINUX-1
      inservice

  class-map match-all GENERIC
  class-map type http loadbalance match-all LB_CLASS_HTTP
    2 match http url .*
    3 match source-address 192.168.130.0 255.255.255.0
  class-map match-all NON-SSL_TEST
    description NON-SSL_TEST
  class-map match-all SSL_END_TO_END_13
    description STICKY FOR SSL TESTING
    3 match source-address 192.168.130.0 255.255.255.0
  class-map match-all TCP-NORM-TEST
    description TCP NORM TEST
    2 match virtual-address 192.168.130.171 tcp eq 22

  policy-map type management first-match POLICY_MGMT

  policy-map type inspect http all-match INSPECT_GOOD_HTTP

  policy-map multi-match SSL_TEST_SUITE_VIPS
    class SSL_END_TO_END_13
      appl-parameter http advanced-options HTTP_PARAM
      connection advanced-options TCP_PARAM
    class NON-SSL_TEST
      appl-parameter http advanced-options HTTP_PARAM
      connection advanced-options TCP_PARAM
    class TCP-NORM-TEST
      loadbalance vip inservice
      loadbalance vip icmp-reply
      connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
    class GENERIC
      appl-parameter http advanced-options HTTP_PARAM

  interface vlan 130
    ip address 192.168.130.36 255.255.255.0
    alias 192.168.130.34 255.255.255.0
    peer ip address 192.168.130.35 255.255.255.0
    access-group input eveyone
    service-policy input POLICY_MGMT
    no shutdown
  interface vlan 281
    ip address 172.28.1.12 255.255.255.0
    alias 172.28.1.10 255.255.255.0
    peer ip address 172.28.1.11 255.255.255.0
    access-group input eveyone
    service-policy input POLICY_MGMT
    no shutdown

  ip route 10.1.0.0 255.255.255.0 192.168.130.254
  username admin password 5 $1$Ot5XpO4e$KlAT/S/YguBVfXRjCgGZZ1  role Admin domain
  default-domain
  username netmon password 5 $1$RaQYCihb$n9azptTRpeFHrmOBsFDaB1  role Network-Monitor domain
  default-domain



  6K-2_ACE2-1/130-VRT-2# changeto Admin



  NOTE: Configuration mode has been disabled on all sessions
```

```
6K-2_ACE2-1/Admin#
--- 09:57:21 ---

+++ 09:57:21 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-2



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-2#


6K-2_ACE2-1/130-VRT-2# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:21 ---

+++ 09:57:21 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-3



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-3#


6K-2_ACE2-1/130-VRT-3# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:22 ---

+++ 09:57:22 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-3



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-3# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging buffered 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7



crypto chaingroup CHAIN1
aaa authentication login error-enable

access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-3_probe_GEN-HTTP"
  connection term forced
probe https GEN_HTTPS
  interval 20
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "130-VRT-3_probe_GEN-HTTPS"
  connection term forced


parameter-map type http HTTP_PARAM
  server-conn reuse
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 20
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version TLS1
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  nagle
  syn-data drop
  exceed-mss allow
  urgent-flag clear

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
```

```
  ssl advanced-options PARM_ACE_TERM

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
```

```
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host ONE-IIS-SERVER
  rserver BRG-IIS-1
    inservice
serverfarm host TCP-NORM-FARM
  predictor leastconns
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-IIS-3
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver BRG-LINUX-3
    inservice
serverfarm host TELNET-NORM-TEST
  rserver BRG-LINUX-1
    inservice

sticky http-cookie SSL_TERM_COOKIE GROUP_10
  cookie insert browser-expire
  serverfarm ACE_TERM_SERVERS_CLEAR
sticky http-cookie SSL_INIT_COOKIE GROUP_20
  cookie insert browser-expire
  serverfarm ACE_INIT_SERVERS_SSL
sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
  cookie insert browser-expire
  serverfarm ACE_END_TO_END_SERVERS_SSL
sticky http-cookie NON_SSL_TESTING GROUP_40
  cookie insert browser-expire
  serverfarm NON-SSL-TEST
sticky ip-netmask 255.255.255.0 address both NEW_GROUP
  serverfarm NON-SSL-TEST

class-map match-all GENERIC
class-map type http loadbalance match-all LB_CLASS_HTTP
  2 match http url .*
class-map match-all NON-SSL_TEST
  description NON-SSL_TEST
class-map match-all SSL_END_TO_END_13
  description STICKY FOR SSL TESTING
  3 match source-address 192.168.130.0 255.255.255.0
class-map match-all TCP-NORM-TEST
  description TCP NORM TEST
  2 match virtual-address 192.168.130.177 tcp eq 22

policy-map type management first-match POLICY_MGMT

policy-map type loadbalance first-match GENERIC
  class LB_CLASS_HTTP
    serverfarm ONE-IIS-SERVER
  class class-default
    serverfarm ONE-IIS-SERVER
policy-map type loadbalance first-match NON_SSL_TESTING
  class LB_CLASS_HTTP
    serverfarm NON-SSL-TEST
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "NON_SSL"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
policy-map type loadbalance first-match PLBSF_L4
  class class-default
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
  class LB_CLASS_HTTP
    serverfarm ACE_END_TO_END_SERVERS_SSL
    insert-http SRC_IP header-value "%is"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_END_TO_END"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_INIT
  class LB_CLASS_HTTP
    serverfarm ACE_INIT_SERVERS_SSL
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "SSL_INIT"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_TERM
  class LB_CLASS_HTTP
    serverfarm ACE_TERM_SERVERS_CLEAR
    insert-http I_AM header-value "SSL_TERM"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http SRC_IP header-value "is"
policy-map type loadbalance first-match TCP-NORM-TESTING
  class LB_CLASS_HTTP
    serverfarm TCP-NORM-FARM
    insert-http SRC_IP header-value "%is"
    insert-http I_AM header-value "TCP-NORM-TEST"
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
policy-map type loadbalance first-match TELNET-NORM_TESTING
  class class-default
    serverfarm TELNET-NORM-TEST

policy-map type inspect http all-match INSPECT_GOOD_HTTP

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_END_TO_END_13
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class TCP-NORM-TEST
    loadbalance vip inservice
    loadbalance policy TCP-NORM-TESTING
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
  class GENERIC
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM

interface vlan 130
  ip address 192.168.130.39 255.255.255.0
  alias 192.168.130.37 255.255.255.0
  peer ip address 192.168.130.38 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.82 192.168.130.82 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.18 255.255.255.0
  alias 172.28.1.16 255.255.255.0
  peer ip address 172.28.1.17 255.255.255.0
  access-group input eveyone
  nat-pool 1 192.168.130.83 192.168.130.83 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown

ip route 10.1.0.0 255.255.255.0 192.168.130.254
ip route 192.168.120.0 255.255.255.0 192.168.130.254
username admin password 5 $1$3ZmJoRxI$eDb3O981o6KQa8Bps/hoG0  role Admin domain
default-domain
username netmon password 5 $1$jTVnTHDe$tKnwxwOpXa9DmihIlfmKk0  role Network-Monitor domain
default-domain




6K-2_ACE2-1/130-VRT-3# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:22 ---

+++ 09:57:22 6K-2_ACE2-1 ctxExec +++
changeto 130-VRT-3



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/130-VRT-3#


6K-2_ACE2-1/130-VRT-3# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:22 ---

+++ 09:57:22 6K-2_ACE2-1 ctxExec +++
changeto LARGE



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/LARGE#


6K-2_ACE2-1/LARGE# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:22 ---

+++ 09:57:22 6K-2_ACE2-1 ctxExec +++
changeto LARGE



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/LARGE# show running-config

Generating configuration....


logging monitor 5








6K-2_ACE2-1/LARGE# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:22 ---

+++ 09:57:22 6K-2_ACE2-1 ctxExec +++
changeto LARGE



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/LARGE#


6K-2_ACE2-1/LARGE# changeto Admin
```

```
NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:22 ---

+++ 09:57:22 6K-2_ACE2-1 ctxExec +++
changeto SH-Gold



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-Gold#


6K-2_ACE2-1/SH-Gold# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:23 ---

+++ 09:57:23 6K-2_ACE2-1 ctxExec +++
changeto SH-Gold



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-Gold# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging history 5
logging buffered 5
logging queue 200
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514


tacacs-server key 7 "vwjjzamggu"
tacacs-server host 172.29.0.235 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.236 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.237 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-gold
  server 172.29.0.235
  server 172.29.0.236
  server 172.29.0.237
```

```
arp 192.168.120.80 00.00.11.11.22.22
arp 172.29.0.200 00.00.22.22.44.44
arp learned-interval 60
aaa authentication login default group sh-gold local
aaa accounting default group sh-gold
aaa authentication login error-enable

access-list ICMP-ONLY line 8 extended permit icmp any any
access-list NAT_ACCESS line 20 extended permit tcp any host 172.29.0.121
access-list NAT_ACCESS line 30 extended permit tcp any host 172.29.0.122
access-list NAT_ACCESS line 40 extended permit tcp any host 172.29.0.140
access-list anyone-ip line 10 extended permit ip any any
access-list anyone-tcp line 10 extended permit tcp any any

kalap udp
  ip address 10.1.0.214 encryption md5 safeharbor

script file 1 FTP_PROBE_SCRIPT
script file 2 TFTP_PROBE
script file 3 LDAP_PROBE


probe icmp FA-PURGE-ICMP
  ip address 172.29.0.253 routed
  interval 5
  passdetect interval 2
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
probe ftp FTP
  interval 10
  faildetect 2
  passdetect interval 10
  receive 5
  expect status 220 220
  open 3
probe icmp HA-ICMP
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:1755
  port 1755
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:554
  port 554
  interval 2
  faildetect 2
  passdetect interval 2
probe http HTTP
  interval 5
  passdetect interval 10
  receive 5
  expect status 200 200
  open 3
probe icmp ICMP
  interval 10
```

```
    faildetect 2
    passdetect interval 10
  probe dns PRB-DNS1
    description "all good addresses"
    interval 5
    passdetect interval 2
    domain www1.safeharbor.com
    expect address 1.1.1.1
    expect address 1.1.1.2
    expect address 1.1.1.3
  probe dns PRB-DNS2
    description "one good address"
    interval 5
    passdetect interval 2
    domain www2.safeharbor.com
    expect address 2.1.1.1
  probe dns PRB-DNS3
    description "2 good addresses, bumpy case"
    interval 5
    passdetect interval 2
    domain WwW3.SaFeHaRbOr.CoM
    expect address 3.1.1.3
    expect address 3.1.1.2
  probe dns PRB-DNS4
    description "one good and one bad address"
    interval 5
    passdetect interval 2
    domain www4.safeharbor.com
    expect address 192.168.1.4
    expect address 4.1.1.1
  probe dns PRB-DNS5
    description "all bad addresses"
    interval 5
    passdetect interval 2
    domain www4.safeharbor.com
    expect address 192.168.1.5
    expect address 192.168.1.6
    expect address 1.1.1.1
  probe dns PRB-DNS6:2222
    description "dns not running on this port, but addresses good"
    port 2222
    interval 5
    passdetect interval 2
    domain www1.safeharbor.com
    expect address 1.1.1.1
    expect address 1.1.1.2
    expect address 1.1.1.3
  probe http PRB-HTTP:84
    port 84
    interval 5
    passdetect interval 4
    passdetect count 10
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:85
    description Server RST 1byte data
    port 85
    interval 5
    passdetect interval 2
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:86
    description Server RST 3200byte data
    port 86
    interval 5
    passdetect interval 2
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:87
    description Server FIN 1byte data
    port 87
    interval 5
    passdetect interval 2
    expect status 200 200
  probe http PRB-HTTP:88
    description Server FIN 3200byte data
    port 88
    interval 5
    passdetect interval 2
    expect status 200 200
  probe https PRB-SSL:443
    interval 5
    passdetect interval 10
    request method get url /index.txt
    expect status 200 200
    header Via header-value "PRB-SSL:443 Probe Header"
    hash
  probe icmp PRED-PING
    ip address 172.29.0.243 routed
    interval 5
    faildetect 2
    passdetect interval 2
  probe radius RADIUS
    interval 2
    faildetect 2
    passdetect interval 2
    credentials lab labtest1 secret ace
  probe scripted SCRIPT_FTP:21
    interval 10
    passdetect interval 2
    passdetect count 2
    receive 5
    script FTP_PROBE_SCRIPT /home/lab/ftp-files/file01.log lab labtest1
  probe scripted SCRIPT_LDAP
    interval 10
    passdetect interval 5
    passdetect count 2
    receive 5
    script LDAP_PROBE
  probe scripted SCRIPT_TFTP
    interval 10
    passdetect interval 5
    passdetect count 2
    receive 5
    script TFTP_PROBE "large file name to test the tftp scripted probe.exe"
  probe https SSL
    interval 5
    passdetect interval 10
    expect status 200 299
    header Via header-value "ACE_Gold_SSL"
    connection term forced
  probe https SSL-445:FIN
    port 445
    interval 5
    passdetect interval 10
    expect status 200 200
  probe https SSL-445:RST
    port 445
    interval 5
    passdetect interval 10
    expect status 200 200
    connection term forced
  probe tcp TCP
    interval 5
    faildetect 2
    passdetect interval 10
    open 3
  probe udp UDP
    interval 5
    passdetect interval 2
  probe udp UDP:2222
    port 2222
    interval 5
    passdetect interval 2


  parameter-map type connection 120SECOND-IDLE
    set timeout inactivity 120
    set tcp timeout half-closed 30
  parameter-map type connection 1SECOND-IDLE
    set timeout inactivity 1
  parameter-map type connection 2SECOND-IDLE
    set timeout inactivity 2
  parameter-map type connection 60SECOND-IDLE
    set timeout inactivity 60
    set tcp timeout half-closed 30
  parameter-map type http COOKIE-DELIM
    persistence-rebalance
    set secondary-cookie-delimiters @$
  parameter-map type http COOKIE-INSERT-HDR-PARSE
    persistence-rebalance
    set header-maxparse-length 4000
  parameter-map type ssl EXPORT_CIPHERS
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  parameter-map type connection HALF-CLOSE_4s
    set tcp timeout half-closed 4
  parameter-map type connection HALF-CLOSE_4send
  parameter-map type connection INFINITE-IDLE
    set timeout inactivity 0
    set tcp timeout half-closed 30
  parameter-map type connection LLFlows
    set timeout inactivity 0
  parameter-map type connection NORM_IP
    set ip tos 22
  parameter-map type connection NORM_TCP
    tcp-options timestamp allow
    reserved-bits clear
    syn-data drop
    urgent-flag clear
  parameter-map type http PARSE_LENGTH
    persistence-rebalance
  parameter-map type http PERSIST-INSERT
    header modify per-request
  parameter-map type http PERSIST-REBAL-4K
    persistence-rebalance
  parameter-map type http PERSIST-REBALANCE
    persistence-rebalance
  parameter-map type connection PRED-CONNS-UDP_CONN
```

```
  set timeout inactivity 300
parameter-map type ssl RC4_128_MD5_CIPHER
  cipher RSA_WITH_RC4_128_MD5
  version TLS1
parameter-map type http REUSE-REBAL
  server-conn reuse
  persistence-rebalance
parameter-map type ssl STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type connection T1
parameter-map type ssl TERM_SSL
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
parameter-map type ssl WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type connection idle-timeout
  set timeout inactivity 480
parameter-map type connection wan-opt
  set tcp wan-optimization rtt 0

rserver host BRG-11
  ip address 192.168.120.11
  inservice
rserver host BRG-12
  ip address 192.168.120.12
  inservice
rserver host BRG-13
  ip address 192.168.120.13
  inservice
rserver host BRG-14
  ip address 192.168.120.14
  inservice
rserver host BRG-15
  ip address 192.168.120.15
  inservice
rserver host LOCAL-239-FRAGRTR
  ip address 172.29.0.239
  inservice
rserver host LOCAL-240
  ip address 172.29.0.240
  inservice
rserver host LOCAL-241
  ip address 172.29.0.241
  inservice
rserver host LOCAL-242
  ip address 172.29.0.242
  inservice
rserver host LOCAL-243
  ip address 172.29.0.243
  inservice
rserver host LOCAL-244
```

```
  ip address 172.29.0.244
  inservice
rserver host LOCAL-245
  ip address 172.29.0.245
  inservice
rserver redirect REDIRECT-100K
  webhost-redirection http://192.168.120.132/redirect-100k.html 302
  inservice
rserver redirect REDIRECT-10K
  webhost-redirection http://192.168.120.133/redirect-10k.html 302
  inservice
rserver redirect REDIRECT-1K
  webhost-redirection http://192.168.120.134/redirect-1k.html 302
  inservice
rserver host RT-151
  ip address 172.28.0.151
  inservice
rserver host RT-152
  ip address 172.28.0.152
  inservice
rserver host RT-153
  ip address 172.28.0.153
  inservice
rserver host RT-154
  ip address 172.28.0.154
  inservice
rserver host WEIGHT-80
  ip address 10.1.0.235
  weight 80
  inservice

ssl-proxy service ACE_TERM
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options TERM_SSL

serverfarm host ADD-REM-SRV
  predictor leastconns
  probe TCP
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-153
    inservice
  rserver RT-154
    inservice
serverfarm host COOKIE
  probe ICMP
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host COOKIE-HASH
  predictor leastconns
  rserver LOCAL-240
    inservice
  rserver LOCAL-240 90
```

```
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-241 90
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-242 90
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-243 90
    inservice
serverfarm host COOKIE-INSERT
  rserver BRG-14
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-152
    inservice
serverfarm host COOKIE-INSERT2
  probe HTTP
  rserver BRG-14
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-152
    inservice
serverfarm host COOKIE1
  probe HTTP
  rserver BRG-12
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-154
    inservice
serverfarm host COOKIE2
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-152
    inservice
serverfarm host CS-COOKIES
  probe TCP
  rserver BRG-12
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-154
    inservice
serverfarm host CS-MOZILLA
  rserver LOCAL-240
    inservice
  rserver RT-151
    inservice
serverfarm host CS-MSIE
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
serverfarm host DEFAULT
  probe ICMP
  rserver BRG-15
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-153
    inservice
serverfarm host FA-PURGE
  failaction purge
  rserver BRG-11
    inservice
  rserver BRG-14
    probe FA-PURGE-ICMP
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-244
    probe FA-PURGE-ICMP
    inservice
  rserver RT-151
    inservice
  rserver RT-154
    probe FA-PURGE-ICMP
    inservice
serverfarm host FTP
  probe FTP
  rserver BRG-11 21
    inservice
  rserver BRG-12 21
    inservice
  rserver LOCAL-240 21
    inservice
  rserver LOCAL-241 21
    inservice
  rserver RT-151 21
    inservice
  rserver RT-152 21
    inservice
serverfarm host GEN-443
  probe SSL
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-80
  predictor leastconns
  probe TCP
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-FTP
  probe FTP
  rserver BRG-13
    inservice
  rserver LOCAL-240
    inservice
  rserver RT-152
    inservice
serverfarm host GEN-UDP
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host GEN2-80
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-153
    inservice
serverfarm host HDR-IXIA
  rserver BRG-14
    inservice
  rserver LOCAL-241
    inservice
serverfarm host HEADER
  probe HTTP
  rserver LOCAL-240
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
serverfarm host HEADER-INSERT
  rserver BRG-13 80
    inservice
  rserver RT-151
    inservice
  rserver RT-151 80
    inservice
  rserver RT-153
    inservice
  rserver RT-153 80
    inservice
serverfarm host HEADER-INSERT2
  rserver BRG-12 80
    inservice
  rserver BRG-14 80
    inservice
  rserver LOCAL-240 80
    inservice
  rserver LOCAL-241 80
    inservice
  rserver LOCAL-244 80
    inservice
  rserver LOCAL-245 80
    inservice
  rserver RT-153 80
```

```
    inservice
  rserver RT-154 80
    inservice
serverfarm host ICMP
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
serverfarm host ICMP2
  rserver BRG-11 7777
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242 7777
    inservice
  rserver RT-152
    inservice
  rserver RT-153 7777
    inservice
serverfarm host IDLE-TCP
  probe TCP
  rserver BRG-15
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-154
    inservice
serverfarm host IDLE-UDP
  probe UDP:2222
  rserver BRG-15
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-154
    inservice
serverfarm host L3
  rserver LOCAL-244
    inservice
serverfarm host LDAP
  probe SCRIPT_LDAP
  rserver BRG-15
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host LENGTHS
  rserver LOCAL-241
    inservice
  rserver LOCAL-244
    inservice
serverfarm host LENGTHS-2
  rserver LOCAL-240
    inservice
  rserver LOCAL-245
    inservice
serverfarm host MAX-CONN
```

```
  probe HTTP
  rserver RT-151
    conn-limit max 4 min 2
    inservice
serverfarm host MAX-CONN2
  probe HTTP
  rserver LOCAL-243
    conn-limit max 500 min 2
    inservice
  rserver RT-151
    conn-limit max 500 min 2
    inservice
  rserver RT-151 90
    conn-limit max 500 min 2
    inservice
  rserver RT-151 91
    conn-limit max 500 min 2
    inservice
  rserver RT-151 92
    conn-limit max 500 min 2
    inservice
  rserver RT-151 93
    conn-limit max 500 min 2
    inservice
  rserver RT-151 94
    conn-limit max 500 min 2
    inservice
  rserver RT-151 95
    conn-limit max 500 min 2
    inservice
  rserver RT-154
    inservice
serverfarm host NORM
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver RT-153
    inservice
serverfarm host NORM2_L7
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  retcode 100 599 check count
  rserver BRG-11 80
    inservice
  rserver LOCAL-241 80
    inservice
  rserver RT-153 80
    inservice
serverfarm host PERSISTENT
  rserver LOCAL-240
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver RT-154
    inservice
serverfarm host PRED-CONNS
  predictor leastconns
  rserver BRG-11
    inservice
  rserver BRG-12
    inservice
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver BRG-15
    inservice
  rserver LOCAL-240
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
  rserver RT-154
    inservice
serverfarm host PRED-CONNS-UDP
  failaction purge
  predictor leastconns
  rserver BRG-11 2222
    inservice
  rserver LOCAL-240 2222
    inservice
  rserver LOCAL-242 2222
    inservice
  rserver LOCAL-244 2222
    probe PRED-PING
    inservice
  rserver RT-151 2222
    inservice
  rserver RT-153 2222
    inservice
  rserver RT-154 2222
    inservice
serverfarm host PREDICTOR
  predictor leastconns
  probe TCP
  rserver BRG-13
    inservice
  rserver BRG-14
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
  rserver RT-153
    inservice
serverfarm host PROBES
  predictor leastconns
  rserver BRG-13
```

```
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-154
    inservice
serverfarm host PROBES-2
  predictor leastconns
  rserver BRG-11
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-152
    inservice
  rserver RT-154
    inservice
serverfarm host PROBES-MANY
  predictor leastconns
  probe SCRIPT_TFTP
  rserver BRG-11
    probe RADIUS
    inservice
  rserver LOCAL-241
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    probe RADIUS
    inservice
  rserver RT-152
    inservice
  rserver RT-154
    probe RADIUS
    inservice
serverfarm host RADIUS
  probe RADIUS
  rserver BRG-11
    inservice
  rserver LOCAL-244
    inservice
  rserver RT-151
    inservice
serverfarm host RED-ALL-SVRS
  rserver LOCAL-240
    inservice
  rserver LOCAL-241
    inservice
serverfarm host REDIRECT
  rserver LOCAL-242
    inservice
  rserver LOCAL-243
    inservice
  rserver LOCAL-244
    inservice
  rserver LOCAL-245
    inservice
serverfarm redirect REDIRECT-100K
  rserver REDIRECT-100K
    inservice
serverfarm redirect REDIRECT-10K
  rserver REDIRECT-10K
    inservice
serverfarm redirect REDIRECT-1K
```

```
    rserver REDIRECT-1K
      inservice
  serverfarm host RHI
    rserver BRG-11
      conn-limit max 4 min 2
      inservice
    rserver LOCAL-241
      conn-limit max 4 min 2
      inservice
    rserver RT-154
      conn-limit max 4 min 2
      inservice
  serverfarm host SORRY
    rserver LOCAL-240
      inservice
  serverfarm host SORRY-BACK
    rserver LOCAL-243
      inservice
    rserver RT-151
      inservice
  serverfarm host STICKY-COOKIE
    probe ICMP
    rserver BRG-11
      inservice
    rserver LOCAL-241
      inservice
    rserver LOCAL-242
      inservice
    rserver RT-154
      inservice
  serverfarm host STICKY-HEADER
    probe HTTP
    rserver BRG-12
      inservice
    rserver LOCAL-243
      inservice
    rserver RT-151
      inservice
  serverfarm host STICKY-HEADER2
    probe HTTP
    rserver BRG-13
      inservice
    rserver LOCAL-244
      inservice
    rserver RT-152
      inservice
  serverfarm host STICKY-NETMASK
    probe ICMP
    rserver BRG-12
      inservice
    rserver LOCAL-242
      inservice
    rserver RT-153
      inservice
  serverfarm host TCP-REUSE
    rserver BRG-15
      inservice
    rserver LOCAL-245
      inservice
    rserver RT-154
      inservice
  serverfarm host UDP
    probe UDP
    rserver BRG-11
```

```
      inservice
    rserver LOCAL-241
      inservice
    rserver RT-151
      inservice
  serverfarm host URL-MAP-128K
    rserver LOCAL-241
      inservice
    rserver LOCAL-242
      inservice
  serverfarm host URL-MAP-16K
    rserver BRG-12
      inservice
    rserver LOCAL-242
      inservice
  serverfarm host URL-MAP-32K
    rserver LOCAL-244
      inservice
    rserver LOCAL-245
      inservice
  serverfarm host URL-MAP-512K
    rserver LOCAL-242
      inservice
    rserver LOCAL-243
      inservice
  serverfarm host URL-MAP-64K
    rserver LOCAL-242 91
      inservice
    rserver LOCAL-244
      inservice
  serverfarm host URL-MAPS
    rserver LOCAL-241
      inservice
    rserver LOCAL-242
      inservice
    rserver LOCAL-244
      inservice
  serverfarm host WEIGHT
    probe HTTP
    rserver BRG-11
      weight 10
      inservice
    rserver LOCAL-240
      weight 20
      inservice
    rserver LOCAL-243
      weight 30
      inservice
    rserver RT-152
      weight 40
      inservice
  serverfarm host fUDP

  sticky ip-netmask 255.255.255.255 address source STKY-GRP-30
    timeout 40
    replicate sticky
    serverfarm GEN-80
  sticky http-cookie cookie-gold-grp40 STKY-GRP-40
    cookie insert browser-expire
    timeout 1
    replicate sticky
    serverfarm GEN2-80
  sticky ip-netmask 255.255.255.255 address both STKY-GRP-31
    timeout 40
    replicate sticky
    serverfarm GEN-UDP
  sticky ip-netmask 255.255.255.255 address both STKY-GRP-32
    timeout 40
    replicate sticky
    serverfarm GEN-FTP
  sticky http-cookie COOKIE_TEST COOKIE-GROUP
    cookie secondary URLCOOKIE
    timeout 40
    replicate sticky
    serverfarm STICKY-COOKIE
    8 static cookie-value "REDSOX0" rserver RT-154
    16 static cookie-value "PATRIOTS0" rserver RT-151
  sticky http-header MSISDN HEADER-GROUP-42
    timeout 30
    replicate sticky
    serverfarm STICKY-HEADER
  sticky http-header TestHeader HEADER-GROUP-41
    header offset 15 length 7
    timeout 30
    replicate sticky
    serverfarm STICKY-HEADER2
  sticky http-cookie COOKIE_INSERT COOKIE-INSERT-GROUP-45
    cookie insert
    timeout 1
    replicate sticky
    serverfarm COOKIE-HASH
  sticky http-cookie COOKIE_TEST COOKIE-MAP-GROUP
    replicate sticky
    serverfarm CS-COOKIES
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-29
    timeout 30
    replicate sticky
    serverfarm MAX-CONN
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-11
    timeout 30
    replicate sticky
    serverfarm URL-MAP-16K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-13
    timeout 30
    replicate sticky
    serverfarm URL-MAP-64K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-12
    timeout 30
    replicate sticky
    serverfarm URL-MAP-32K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-14
    timeout 30
    replicate sticky
    serverfarm URL-MAP-128K
  sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-15
    timeout 30
    replicate sticky
    serverfarm URL-MAP-512K
  sticky http-cookie Safeharbor-Cookie1 COOKIE-GROUP-42
    cookie insert
    timeout 30
    replicate sticky
  sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes
  COOKIE-INSERT-GROUP-46
    cookie insert
    timeout 1
    replicate sticky
    serverfarm COOKIE-INSERT
```

```
sticky http-cookie COOKIE_TEST STKY-GRP-43
  cookie offset 1 length 999
  timeout 30
  replicate sticky
  serverfarm PREDICTOR
sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes
COOKIE-INS2-GROUP-46
  cookie insert
  timeout 1
  serverfarm COOKIE-INSERT
sticky ip-netmask 255.255.255.255 address source STKY-GRP-50
  timeout 20
  serverfarm SORRY backup SORRY-BACK
sticky ip-netmask 255.255.255.255 address both STKY-GRP-33
  timeout 20
  replicate sticky
  serverfarm STICKY-NETMASK

class-map type http loadbalance match-all 128K-FORWARDING
  2 match http url .*128k.*
class-map type http loadbalance match-all 16K-FORWARDING
  2 match http url .*16k.*
class-map type http loadbalance match-all 32K-FORWARDING
  2 match http url .*32k.*
class-map type http loadbalance match-all 512K-FORWARDING
  2 match http url .*512k.*
class-map type http loadbalance match-all 64K-FORWARDING
  2 match http url .*64k.*
class-map match-all ADD-REM-SRV-VIP_110:80
  2 match virtual-address 192.168.120.110 tcp eq www
class-map type http loadbalance match-all BROWSER_FIREFOX
  2 match http header User-Agent header-value ".*Firefox.*"
class-map type http loadbalance match-all BROWSER_MOZILLA
  2 match http header User-Agent header-value ".*Mozilla.*"
class-map type http loadbalance match-all BROWSER_MOZILLA40
  2 match http header User-Agent header-value ".*Mozilla/4.0.*"
class-map type http loadbalance match-all BROWSER_MOZILLA50
  2 match http header User-Agent header-value ".*Mozilla/5.0.*"
class-map type http loadbalance match-all BROWSER_MSIE
  2 match http header User-Agent header-value ".*MSIE.*"
class-map match-all COOKIE-HASH-VIP_10.20.30.40:80
  2 match virtual-address 10.20.30.40 tcp eq www
class-map match-all COOKIE-INS2-VIP_118:8888
  2 match virtual-address 192.168.120.118 tcp eq 8888
class-map match-all COOKIE-INSERT-VIP_118:80
  2 match virtual-address 192.168.120.118 tcp eq www
class-map match-all COOKIE-MAP-VIP_124:80
  2 match virtual-address 192.168.120.124 tcp eq www
class-map type http loadbalance match-all COOKIE-MAP:80
  2 match http cookie COOKIE_TEST cookie-value "This is a test0"
class-map match-all FA-PURGE-VIP_113:ANY
  2 match virtual-address 192.168.120.113 any
class-map type ftp inspect match-any FTP-L7-MAX-DENY
  2 match request-method appe
  3 match request-method cdup
  4 match request-method get
  5 match request-method help
  6 match request-method mkd
  7 match request-method rmd
  8 match request-method rnfr
  9 match request-method rnto
  10 match request-method site
  11 match request-method stou
  12 match request-method cwd
```

```
class-map type ftp inspect match-any FTP-L7-MAX-DENY2
  2 match request-method syst
class-map type ftp inspect match-any FTP-L7-MIN-DENY
  2 match request-method mkd
  3 match request-method rmd
class-map match-all FTP-VIP-NAT_119
  2 match destination-address 192.168.120.119 255.255.255.255
class-map match-all FTP-VIP_119:1111
  2 match virtual-address 192.168.120.119 tcp eq 1111
class-map match-all FTP-VIP_119:3333-4444
  2 match virtual-address 192.168.120.119 tcp range 3333 4444
class-map match-all GEN-NAT_120
  2 match virtual-address 192.168.120.120 any
class-map match-all GEN-VIP_120:21
  2 match virtual-address 192.168.120.120 tcp eq ftp
class-map match-all GEN-VIP_120:443
  2 match virtual-address 192.168.120.120 tcp eq https
class-map match-all GEN-VIP_120:80
  2 match virtual-address 192.168.120.120 tcp eq www
class-map match-all GEN-VIP_120:UDP
  2 match virtual-address 192.168.120.120 udp any
class-map match-all HDR-IXIA-VIP_123:80
  2 match virtual-address 192.168.120.123 tcp eq www
class-map match-all HEADER-INSERT-VIP_121:443
  2 match virtual-address 192.168.120.121 tcp eq https
class-map match-all HEADER-INSERT-VIP_121:80
  3 match virtual-address 192.168.120.121 tcp eq www
class-map match-all HEADER-INSERT2-VIP_122:443
  2 match virtual-address 192.168.120.122 tcp eq https
class-map match-all HEADER-INSERT2-VIP_122:80
  2 match virtual-address 192.168.120.122 tcp eq www
class-map match-all HEADER-VIP_125:80
  2 match virtual-address 192.168.120.125 tcp eq www
class-map match-all ICMP
  2 match access-list ICMP-ONLY
class-map match-all ICMP-UDP-VIP_138
  3 match virtual-address 192.168.120.138 udp eq 2222
class-map match-all ICMP-URL-VIP_138:80
  2 match virtual-address 192.168.120.138 tcp eq www
class-map match-all IDLE-VIP_111:TCP
  2 match virtual-address 192.168.120.111 tcp any
class-map match-all IDLE-VIP_111:UDP
  2 match virtual-address 192.168.120.111 udp any
class-map type http loadbalance match-all INDEX.HTML
  2 match http url /index.html
class-map match-all L3_139
  2 match virtual-address 192.168.120.139 any
class-map match-all LENGTHS-VIP_136:80
  2 match virtual-address 192.168.120.136 tcp eq www
class-map match-all MAX-CONN-VIP_105
  2 match virtual-address 192.168.120.105 any
class-map match-all MAX-CONN-VIP_126:80
  2 match virtual-address 192.168.120.126 tcp eq www
class-map type management match-any MGT
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
  8 match protocol kalap-udp any
class-map type http loadbalance match-all MSISDN
  2 match http header MSISDN header-value ".*"
```

```
class-map match-any NAT_CLASS
  2 match access-list NAT_ACCESS
class-map match-any NORM-VIP_142:80
  2 match virtual-address 192.168.120.142 tcp eq www
class-map match-any NORM2_L7-VIP_142:888
  2 match virtual-address 192.168.120.142 tcp eq 888
class-map match-all NORM_ALL_TRAFFIC-VIP_ANY
  2 match any
class-map type http loadbalance match-any P-COOKIE-INS
  2 match http url /index.html* method GET
class-map type http loadbalance match-any P-COOKIE-INS2
  2 match http url .*
class-map type http loadbalance match-all P-HDR-INSERT
  2 match http url .*
class-map type http loadbalance match-all P-HDR-IXIA
  2 match http url .*
class-map type http loadbalance match-all P-HDR-SRCDST-IP
  2 match http url .*
class-map match-all PERSISTENT-VIP_131:80
  2 match virtual-address 192.168.120.131 tcp eq www
class-map match-all PRED-CONNS-UDP-VIP_128:2222
  2 match virtual-address 192.168.120.128 udp any
class-map match-all PRED-CONNS-VIP_128:80
  2 match virtual-address 192.168.120.128 tcp eq www
class-map match-all PREDICTOR_117:80
  2 match virtual-address 192.168.120.117 tcp eq www
class-map match-all RADIUS-NAT_134
  2 match virtual-address 192.168.120.134 udp eq radius-auth
class-map match-all RED-100K-VIP_132:80
  2 match virtual-address 192.168.120.132 tcp eq www
class-map match-all RED-10K-VIP_133:80
  2 match virtual-address 192.168.120.133 tcp eq www
class-map match-all RED-1K-VIP_134:80
  2 match virtual-address 192.168.120.134 tcp eq www
class-map type http loadbalance match-all REDIRECT-100K
  2 match http url .*redirect-100k.html
class-map type http loadbalance match-all REDIRECT-10K
  2 match http url .*redirect-10k.html
class-map type http loadbalance match-all REDIRECT-1K
  2 match http url .*redirect-1k.html
class-map match-all REDIRECT-VIP_135:80
  2 match virtual-address 192.168.120.135 tcp eq www
class-map match-all REDIRECT_135:80
  2 match virtual-address 192.168.120.135 tcp eq www
class-map match-all RHI_125.100:80
  2 match virtual-address 192.168.125.100 tcp eq www
class-map match-all RHI_125.101:80
  2 match virtual-address 192.168.125.101 tcp eq www
class-map match-all RHI_125.102:80
  2 match virtual-address 192.168.125.102 tcp eq www
class-map match-all RHI_125.103:80
  2 match virtual-address 192.168.125.103 tcp eq www
class-map match-all RHI_125.104:80
  2 match virtual-address 192.168.125.104 tcp eq www
class-map match-all RHI_125.105:80
  2 match virtual-address 192.168.125.105 tcp eq www
class-map match-all RHI_125.106:80
  2 match virtual-address 192.168.125.106 tcp eq www
class-map match-all RHI_125.107:80
  2 match virtual-address 192.168.125.107 tcp eq www
class-map match-all RHI_125.108:80
  2 match virtual-address 192.168.125.108 tcp eq www
class-map match-all RHI_125.109:80
  2 match virtual-address 192.168.125.109 tcp eq www
```

```
class-map match-all RHI_125.110:80
  2 match virtual-address 192.168.125.110 tcp eq www
class-map match-all RHI_125.111:80
  2 match virtual-address 192.168.125.111 tcp eq www
class-map match-all RHI_125.112:80
  2 match virtual-address 192.168.125.112 tcp eq www
class-map match-all RHI_125.113:80
  2 match virtual-address 192.168.125.113 tcp eq www
class-map match-all RHI_125.114:80
  2 match virtual-address 192.168.125.114 tcp eq www
class-map match-all RHI_125.115:80
  2 match virtual-address 192.168.125.115 tcp eq www
class-map match-all RHI_125.116:80
  2 match virtual-address 192.168.125.116 tcp eq www
class-map match-all RHI_125.117:80
  2 match virtual-address 192.168.125.117 tcp eq www
class-map match-all RHI_125.118:80
  2 match virtual-address 192.168.125.118 tcp eq www
class-map match-all RHI_125.119:80
  2 match virtual-address 192.168.125.119 tcp eq www
class-map match-all RHI_125.120-127
  2 match virtual-address 192.168.125.127 255.255.255.248 any
class-map match-all SORRY-VIP_137:80
  2 match virtual-address 192.168.120.137 tcp eq www
class-map match-all STICKY-COOKIE-VIP_127:80
  2 match virtual-address 192.168.120.127 tcp eq www
class-map match-all STICKY-HEADER_129:80
  2 match virtual-address 192.168.120.129 tcp eq www
class-map match-all STICKY-IP_115:ANY
  2 match virtual-address 192.168.120.115 any
class-map match-any TCP-REUSE-VIP_141:80
  10 match virtual-address 192.168.120.141 tcp eq www
class-map match-any TCP-REUSE-VIP_141:81
  10 match virtual-address 192.168.120.141 tcp eq 81
class-map type http loadbalance match-all TestHeader
  2 match http header TestHeader header-value ".*"
class-map match-all UDP-VIP_114:UDP
  2 match virtual-address 192.168.120.114 udp any
class-map match-all UDP-VIP_114:UDP-53
  2 match virtual-address 192.168.120.114 udp eq domain
class-map type http loadbalance match-all URL*_L7
  2 match http url .*
class-map match-all URL-MAPS-VIP_130:80
  2 match virtual-address 192.168.120.130 tcp eq www
class-map type http loadbalance match-all URLCOOKIE-MAP1
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE1"
class-map type http loadbalance match-all URLCOOKIE-MAP2
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE2"
class-map match-all WEIGHT_112:80
  2 match virtual-address 192.168.120.112 tcp eq www

policy-map type management first-match P-MGT
  class MGT
    permit

policy-map type loadbalance first-match FTP-LB-SF_FTP
  class class-default
    serverfarm FTP
policy-map type loadbalance first-match MAX-CONN-LB-SF_MAX-CONN2
  class INDEX.HTML
    serverfarm MAX-CONN
  class URL*_L7
    serverfarm MAX-CONN2
policy-map type loadbalance first-match PLBSF-FTP-test
```

```
  class class-default
policy-map type loadbalance first-match PLBSF_ADD-REM-SRV
  class class-default
    serverfarm ADD-REM-SRV
policy-map type loadbalance first-match PLBSF_COOKIE-HASH
  class class-default
    sticky-serverfarm COOKIE-INSERT-GROUP-45
policy-map type loadbalance first-match PLBSF_COOKIE-INS2
  class class-default
    sticky-serverfarm COOKIE-INS2-GROUP-46
policy-map type loadbalance first-match PLBSF_COOKIE-INSERT
  class P-COOKIE-INS
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Destination_IP header-value "%id"
    insert-http Source-IP header-value "%is"
  class P-COOKIE-INS2
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Destination_IP header-value "%id"
    insert-http Source-IP header-value "%is"
policy-map type loadbalance first-match PLBSF_COOKIE-MAP
  class URLCOOKIE-MAP2
    serverfarm COOKIE2
  class URLCOOKIE-MAP1
    serverfarm COOKIE1
  class COOKIE-MAP:80
    serverfarm COOKIE
  class class-default
    serverfarm GEN-80
policy-map type loadbalance first-match PLBSF_FA-PURGE
  class class-default
    serverfarm FA-PURGE
policy-map type loadbalance first-match PLBSF_GEN-443
  class class-default
    serverfarm GEN-443
policy-map type loadbalance first-match PLBSF_GEN-80
  class INDEX.HTML
    sticky-serverfarm STKY-GRP-40
  class class-default
    sticky-serverfarm STKY-GRP-30
policy-map type loadbalance first-match PLBSF_GEN-FTP
  class class-default
    sticky-serverfarm STKY-GRP-32
policy-map type loadbalance first-match PLBSF_GEN-UDP
  class class-default
    sticky-serverfarm STKY-GRP-31
policy-map type loadbalance first-match PLBSF_HDR-IXIA
  class P-HDR-IXIA
    serverfarm HDR-IXIA
policy-map type loadbalance first-match PLBSF_HEADER
  class BROWSER_FIREFOX
    serverfarm CS-MOZILLA
  class BROWSER_MOZILLA40
    serverfarm CS-MSIE
  class BROWSER_MOZILLA50
    serverfarm CS-MOZILLA
  class BROWSER_MSIE
    serverfarm CS-MSIE
  class BROWSER_MOZILLA
    serverfarm CS-MOZILLA
  class class-default
    serverfarm HEADER
policy-map type loadbalance first-match PLBSF_HEADER-INSERT
  class P-HDR-INSERT
    serverfarm HEADER-INSERT
    insert-http Source-IP header-value "%is"
    insert-http Accept header-value "anything"
    insert-http Pragma header-value "Pragma no Pragma that is the question"
    insert-http Destination_iP header-value "%id"
    insert-http
Custom-header_name_size_100bytes_abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHIJKLMNOPQR
S_100BYTES header-value "Size of inserted header value is 100 bytes
abcdefghijklmnopqrstuvwxyz-01234567890_ABCDEFGHI_100BYTES"
policy-map type loadbalance first-match PLBSF_HEADER-INSERT2
  class P-HDR-SRCDST-IP
    serverfarm HEADER-INSERT2
    insert-http Source-IP header-value "%is"
    insert-http Destination_iP header-value "%id"
policy-map type loadbalance first-match PLBSF_ICMP
  class INDEX.HTML
    serverfarm ICMP
policy-map type loadbalance first-match PLBSF_ICMP2
  class class-default
    serverfarm ICMP2
policy-map type loadbalance first-match PLBSF_IDLE-TCP
  class class-default
    serverfarm IDLE-TCP
policy-map type loadbalance first-match PLBSF_IDLE-UDP
  class class-default
    serverfarm IDLE-UDP
policy-map type loadbalance first-match PLBSF_L3
  class class-default
    serverfarm L3
policy-map type loadbalance first-match PLBSF_LENGTHS
  class INDEX.HTML
    serverfarm LENGTHS
  class class-default
    serverfarm LENGTHS-2
policy-map type loadbalance first-match PLBSF_MAX-CONN
  class INDEX.HTML
    sticky-serverfarm STICKY-GROUP-29
  class class-default
    serverfarm MAX-CONN2
policy-map type loadbalance first-match PLBSF_NORM
  class class-default
    serverfarm NORM
policy-map type loadbalance first-match PLBSF_NORM2_L7
  class class-default
    serverfarm NORM2_L7
policy-map type loadbalance first-match PLBSF_PERSISTENT
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm PERSISTENT
policy-map type loadbalance first-match PLBSF_PRED-CONNS
  class class-default
    serverfarm PRED-CONNS
policy-map type loadbalance first-match PLBSF_PRED-CONNS-UDP
  class class-default
    serverfarm PRED-CONNS-UDP
policy-map type loadbalance first-match PLBSF_PREDICTOR
```

```
  class class-default
    serverfarm PREDICTOR
policy-map type loadbalance first-match PLBSF_RADIUS-1812
  class class-default
    serverfarm RADIUS
policy-map type loadbalance first-match PLBSF_RED-ALL-SVRS
  class class-default
    serverfarm RED-ALL-SVRS
policy-map type loadbalance first-match PLBSF_REDIRECT
  class REDIRECT-1K
    serverfarm REDIRECT-1K
  class REDIRECT-10K
    serverfarm REDIRECT-10K
  class REDIRECT-100K
    serverfarm REDIRECT-100K
  class class-default
    serverfarm REDIRECT
policy-map type loadbalance first-match PLBSF_RHI
  class URL*_L7
    serverfarm RHI
policy-map type loadbalance first-match PLBSF_SORRY
  class class-default
    sticky-serverfarm STKY-GRP-50
policy-map type loadbalance first-match PLBSF_STICKY-COOKIE
  class INDEX.HTML
    sticky-serverfarm COOKIE-GROUP
  class URL*_L7
    serverfarm GEN-80
policy-map type loadbalance first-match PLBSF_STICKY-HEADER
  class MSISDN
    sticky-serverfarm HEADER-GROUP-42
  class TestHeader
    sticky-serverfarm HEADER-GROUP-41
  class class-default
    serverfarm DEFAULT
policy-map type loadbalance first-match PLBSF_STICKY-NETMASK
  class class-default
    sticky-serverfarm STKY-GRP-33
policy-map type loadbalance first-match PLBSF_TCP-REUSE
  class URL*_L7
    serverfarm TCP-REUSE
policy-map type loadbalance first-match PLBSF_UDP
  class class-default
    serverfarm UDP
policy-map type loadbalance first-match PLBSF_URL-MAPS
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm URL-MAPS
policy-map type loadbalance first-match PLBSF_WEIGHT
  class class-default
    serverfarm WEIGHT

policy-map type inspect ftp first-match FTP-INSPSF_FTP
  class FTP-L7-MAX-DENY2
    mask-reply
  class FTP-L7-MAX-DENY
    deny

policy-map multi-match NAT_POLICY
  class NAT_CLASS
    nat dynamic 1 vlan 120
policy-map multi-match NORMALIZATION
  class NORM_ALL_TRAFFIC-VIP_ANY
    connection advanced-options NORM_IP
policy-map multi-match SH-Gold-VIPs
  class ADD-REM-SRV-VIP_110:80
    loadbalance vip inservice
    loadbalance policy PLBSF_ADD-REM-SRV
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
  class WEIGHT_112:80
    loadbalance vip inservice
    loadbalance policy PLBSF_WEIGHT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class FA-PURGE-VIP_113:ANY
    loadbalance vip inservice
    loadbalance policy PLBSF_FA-PURGE
    nat dynamic 1 vlan 120
    connection advanced-options INFINITE-IDLE
  class UDP-VIP_114:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_UDP
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    connection advanced-options 1SECOND-IDLE
  class ICMP-UDP-VIP_138
    loadbalance vip inservice
    loadbalance policy PLBSF_ICMP2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options 60SECOND-IDLE
  class COOKIE-HASH-VIP_10.20.30.40:80
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-HASH
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class URL-MAPS-VIP_130:80
    loadbalance vip inservice
    loadbalance policy PLBSF_URL-MAPS
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PERSISTENT-VIP_131:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PERSISTENT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class REDIRECT-VIP_135:80
    loadbalance vip inservice
    loadbalance policy PLBSF_REDIRECT
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class LENGTHS-VIP_136:80
    loadbalance vip inservice
    loadbalance policy PLBSF_LENGTHS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PARSE_LENGTH
  class NORM-VIP_142:80
    loadbalance vip inservice
    loadbalance policy PLBSF_NORM
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class FTP-VIP-NAT_119
    nat dynamic 1 vlan 120
    nat dynamic 1 vlan 29
  class FTP-VIP_119:1111
    loadbalance vip inservice
    loadbalance policy FTP-LB-SF_FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class MAX-CONN-VIP_105
    loadbalance vip inservice
    loadbalance policy MAX-CONN-LB-SF_MAX-CONN2
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class FTP-VIP_119:3333-4444
    loadbalance vip inservice
    loadbalance policy FTP-LB-SF_FTP
    loadbalance vip icmp-reply active
    inspect ftp strict
  class PREDICTOR_117:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PREDICTOR
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class GEN-VIP_120:80
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-80
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class GEN-VIP_120:443
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-443
    loadbalance vip icmp-reply active
  class GEN-VIP_120:21
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-FTP
    loadbalance vip icmp-reply active
    inspect ftp
  class GEN-VIP_120:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_GEN-UDP
    loadbalance vip icmp-reply active
    connection advanced-options 2SECOND-IDLE
  class IDLE-VIP_111:TCP
    loadbalance vip inservice
    loadbalance policy PLBSF_IDLE-TCP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options 60SECOND-IDLE
  class IDLE-VIP_111:UDP
    loadbalance vip inservice
    loadbalance policy PLBSF_IDLE-UDP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options 60SECOND-IDLE
  class STICKY-IP_115:ANY
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-NETMASK
```

```
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
  class STICKY-COOKIE-VIP_127:80
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-COOKIE
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-DELIM
  class STICKY-HEADER_129:80
    loadbalance vip inservice
    loadbalance policy PLBSF_STICKY-HEADER
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class TCP-REUSE-VIP_141:81
    loadbalance vip inservice
    loadbalance policy PLBSF_TCP-REUSE
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options REUSE-REBAL
  class ICMP-URL-VIP_138:80
    loadbalance vip inservice
    loadbalance policy PLBSF_ICMP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class ICMP
    inspect icmp error
  class HEADER-INSERT-VIP_121:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class HDR-IXIA-VIP_123:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HDR-IXIA
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class HEADER-INSERT2-VIP_122:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBAL-4K
  class COOKIE-INSERT-VIP_118:80
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-INSERT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-INSERT-HDR-PARSE
  class COOKIE-MAP-VIP_124:80
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-MAP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-DELIM
  class HEADER-VIP_125:80
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER
    loadbalance vip icmp-reply
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PRED-CONNS-VIP_128:80
    loadbalance vip inservice
    loadbalance policy PLBSF_PRED-CONNS
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PRED-CONNS-UDP-VIP_128:2222
    loadbalance vip inservice
    loadbalance policy PLBSF_PRED-CONNS-UDP
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
    connection advanced-options PRED-CONNS-UDP_CONN
  class MAX-CONN-VIP_126:80
    loadbalance vip inservice
    loadbalance policy PLBSF_MAX-CONN
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    connection advanced-options HALF-CLOSE_4s
  class RED-100K-VIP_132:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RED-10K-VIP_133:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RED-1K-VIP_134:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RED-ALL-SVRS
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class SORRY-VIP_137:80
    loadbalance vip inservice
    loadbalance policy PLBSF_SORRY
    loadbalance vip icmp-reply active
    appl-parameter http advanced-options PERSIST-REBALANCE
  class TCP-REUSE-VIP_141:80
    loadbalance vip inservice
    loadbalance policy PLBSF_TCP-REUSE
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options REUSE-REBAL
  class NORM2_L7-VIP_142:888
    loadbalance vip inservice
    loadbalance policy PLBSF_NORM2_L7
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class COOKIE-INS2-VIP_118:8888
    loadbalance vip inservice
    loadbalance policy PLBSF_COOKIE-INS2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options COOKIE-INSERT-HDR-PARSE
  class RADIUS-NAT_134
    loadbalance vip inservice
    loadbalance policy PLBSF_RADIUS-1812
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    nat dynamic 1 vlan 29
  class UDP-VIP_114:UDP-53
    loadbalance vip inservice
    loadbalance policy PLBSF_UDP
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 120
    inspect dns
    connection advanced-options 1SECOND-IDLE
  class L3_139
    loadbalance vip inservice
    loadbalance policy PLBSF_L3
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
  class GEN-NAT_120
    nat dynamic 1 vlan 120
  class HEADER-INSERT-VIP_121:443
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    nat dynamic 1 vlan 29
    appl-parameter http advanced-options PERSIST-REBAL-4K
    ssl-proxy server ACE_TERM
  class HEADER-INSERT2-VIP_122:443
    loadbalance vip inservice
    loadbalance policy PLBSF_HEADER-INSERT2
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 120
    nat dynamic 1 vlan 29
    appl-parameter http advanced-options PERSIST-REBAL-4K
    ssl-proxy server ACE_TERM
policy-map multi-match SH-Gold-VIPs3
  class RHI_125.100:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.101:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.102:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.103:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.104:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.105:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.106:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.107:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.108:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.109:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.110:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.111:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.112:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.113:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.114:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.115:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.116:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.117:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.118:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.119:80
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE
  class RHI_125.120-127
    loadbalance vip inservice
    loadbalance policy PLBSF_RHI
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    loadbalance vip advertise metric 254
    nat dynamic 1 vlan 120
    appl-parameter http advanced-options PERSIST-REBALANCE

service-policy input P-MGT

interface vlan 28
interface vlan 29
  ip address 172.29.0.3 255.255.255.0
  alias 172.29.0.1 255.255.255.0
  peer ip address 172.29.0.2 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.71 192.168.120.71 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  no shutdown
interface vlan 99
  ip address 192.168.99.3 255.255.255.0
  peer ip address 192.168.99.2 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 105
  ip address 192.168.105.3 255.255.255.0
  peer ip address 192.168.105.2 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
interface vlan 120
  description Upstream VLAN_120 - Clients and VIPs
  ip address 192.168.120.3 255.255.255.0
  alias 192.168.120.1 255.255.255.0
  peer ip address 192.168.120.2 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  nat-pool 1 192.168.120.70 192.168.120.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  service-policy input NAT_POLICY
  no shutdown

ft track host GW_251-252
  track-host 192.168.16.252
  peer track-host 192.168.16.251
  peer probe HA-ICMP priority 10
  probe HA-ICMP priority 5
  priority 5
  peer priority 110
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 5
  peer priority 110
ft track interface Int_4/37_6/13_V99
  track-interface vlan 99
  peer track-interface vlan 99
  priority 5
  peer priority 110
ft track host RT-241
  peer track-host 172.29.0.241
  peer probe HA-TCP:554 priority 50
  peer probe HA-TCP:1755 priority 60

domain SH-Gold-Domain
  add-object all
domain KALAP_TAG
  add-object class-map GEN-VIP_120:21

role SHAdmin
  rule 1 permit create
  rule 2 permit monitor
  rule 3 permit modify
role SHUser
  rule 1 deny create
  rule 2 permit monitor
  rule 3 deny modify
  rule 4 permit debug
role TEST

ip route 10.1.0.0 255.255.255.0 192.168.120.254
ip route 172.28.0.0 255.255.255.0 172.29.0.253
ip route 192.168.16.251 255.255.255.255 192.168.105.251
ip route 192.168.16.252 255.255.255.255 192.168.105.252
ip route 10.3.0.0 255.255.255.0 192.168.120.254
username admin password 5 $1$hU1iScF8$WmpdK4IcQI2ofTMDm6l.N1  role Admin domain
default-domain
username localadmin password 5 $1$g5rd5HO2$C34zVe3a9f73Dce/WNvbM.  role Admin domain
SH-Gold-Domain default-domain
username localuser password 5 $1$I21oqX4Q$/OqAKTdBbe8xreKwZtWR3.  role Network-Monitor
domain SH-Gold-Domain
username vrtadmgold password 5 $1$.gNJPJS6$UtozYODAuirfw8XHR1FA8/  role Admin domain
SH-Gold-Domain
username vrtnetmongold password 5 $1$InjySHhu$oLsQV267Nu68q3fH6h7Z4.  role Network-Monitor
domain SH-Gold-Domain
username vrtjohndoe password 5 $1$klNcwN8c$3hTNwFSMtned/9k2a5RPg.  role Admin domain
SH-Gold-Domain

snmp-server community ACE-private group Network-Monitor
snmp-server community ACE-public group Network-Monitor

snmp-server host 10.1.0.242 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-2_ACE2-1/SH-Gold# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:23 ---

+++ 09:57:23 6K-2_ACE2-1 ctxExec +++
changeto SH-Gold



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-Gold#


6K-2_ACE2-1/SH-Gold# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:23 ---

+++ 09:57:23 6K-2_ACE2-1 ctxExec +++
changeto SH-LOAD



NOTE: Configuration mode has been disabled on all sessions
```

```
6K-2_ACE2-1/SH-LOAD#


6K-2_ACE2-1/SH-LOAD# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:23 ---

+++ 09:57:23 6K-2_ACE2-1 ctxExec +++
changeto SH-LOAD



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-LOAD# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging history 5
logging buffered 5
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514


aaa authentication login error-enable

access-list everyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 10
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_CLEAR"
probe https GEN_HTTPS
  interval 10
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_SSL"
probe icmp ICMP
  interval 5
  passdetect interval 10


parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version SSL3
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 6
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
```

```
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  key pkey.pem
  cert end-to-end.pem
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  key pkey.pem
  cert term.pem
  ssl advanced-options PARM_ACE_TERM_STRONG_CIPHERS

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
```

```
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host L3
  probe GEN_HTTP
  probe ICMP
  rserver BRG-IIS-4
    inservice
  rserver BRG-IIS-5
    inservice
  rserver BRG-LINUX-4
    inservice
  rserver BRG-LINUX-5
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice

sticky http-cookie SSL_TERM_COOKIE GROUP_10
  cookie insert browser-expire
  serverfarm ACE_TERM_SERVERS_CLEAR
sticky http-cookie SSL_INIT_COOKIE GROUP_20
  cookie insert browser-expire
  serverfarm ACE_INIT_SERVERS_SSL
sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
  cookie insert browser-expire
  serverfarm ACE_END_TO_END_SERVERS_SSL
sticky http-cookie NON_SSL_TESTING GROUP_40
  cookie insert browser-expire
  serverfarm NON-SSL-TEST

class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  4 match request-method rfc get
  5 match request-method rfc head
  6 match request-method rfc options
  7 match request-method rfc post
  8 match request-method rfc put
  9 match request-method rfc trace
  10 match url .*
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
class-map match-all L3_114
  2 match virtual-address 192.168.130.114 any
class-map type http loadbalance match-all LB_CLASS_HTTP
  2 match http url .*
  3 match source-address 10.1.0.0 255.255.0.0
class-map match-all NON-SSL_TEST_114
  description NON-SSL_TEST
  2 match virtual-address 192.168.130.114 tcp eq www
class-map type management match-any REMOTE
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol https any
  7 match protocol snmp any
class-map match-all SSL_END_TO_END_113
  description END to END SSL VIP
  2 match virtual-address 192.168.130.113 tcp eq https
class-map match-all SSL_INIT_112
  description SSL INIT CLEAR VIP
  2 match virtual-address 192.168.130.112 tcp eq www
class-map match-all SSL_TERM_111
  description TERM SSL VIP
  2 match virtual-address 192.168.130.111 tcp eq https
class-map type http loadbalance match-all STICK_ME_TO_SERVER
  description STICKY FOR SSL TESTING
  2 match http url .*.jpg
  3 match source-address 10.1.0.0 255.255.0.0
class-map match-all URL*_L7

policy-map type management first-match POLICY_MGMT
  class REMOTE
    permit

policy-map type loadbalance first-match NON_SSL_TESTING
  class STICK_ME_TO_SERVER
    sticky-serverfarm GROUP_40
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http I_AM header-value "NON_SSL"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
  class LB_CLASS_HTTP
    serverfarm NON-SSL-TEST
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http I_AM header-value "NON_SSL"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match PLBSF_L3
  class class-default
    serverfarm L3
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
  class STICK_ME_TO_SERVER
    sticky-serverfarm GROUP_30
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_END_TO_END"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client ACE_AS_CLIENT
  class LB_CLASS_HTTP
    serverfarm ACE_END_TO_END_SERVERS_SSL
    insert-http I_AM header-value "SSL_END_TO_END"
    insert-http SRC_Port header-value "%ps"
policy-map type loadbalance first-match POLICY_SSL_INIT
  class STICK_ME_TO_SERVER
    sticky-serverfarm GROUP_20
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_INIT"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client ACE_AS_CLIENT
  class LB_CLASS_HTTP
    serverfarm ACE_INIT_SERVERS_SSL
    insert-http SRC_Port header-value "%ps"
    insert-http DEST_IP header-value "%id"
    insert-http DEST_Port header-value "%pd"
    insert-http I_AM header-value "SSL_INIT"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client ACE_AS_CLIENT
policy-map type loadbalance first-match POLICY_SSL_TERM
  class STICK_ME_TO_SERVER
    sticky-serverfarm GROUP_10
    insert-http SRC_IP header-value "is"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM"
  class LB_CLASS_HTTP
    serverfarm ACE_TERM_SERVERS_CLEAR
    insert-http SRC_IP header-value "is"
    insert-http DEST_Port header-value "%pd"
    insert-http DEST_IP header-value "%id"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM"

policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_TERM_111
    loadbalance vip inservice
    loadbalance policy POLICY_SSL_TERM
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server ACE_TERM
    connection advanced-options TCP_PARAM
  class SSL_INIT_112
    loadbalance vip inservice
    loadbalance policy POLICY_SSL_INIT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class SSL_END_TO_END_113
    loadbalance vip inservice
    loadbalance policy POLICY_SSL_END_TO_END
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server ACE_END_TO_END
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST_114
    loadbalance vip inservice
    loadbalance policy NON_SSL_TESTING
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281
    inspect http policy INSPECT_GOOD_HTTP
  class L3_114
    loadbalance vip inservice
    loadbalance policy PLBSF_L3
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 130
    nat dynamic 1 vlan 281

service-policy input POLICY_MGMT

interface vlan 130
  ip address 192.168.130.9 255.255.255.0
  alias 192.168.130.7 255.255.255.0
  peer ip address 192.168.130.8 255.255.255.0
  fragment chain 256
  fragment min-mtu 68
  access-group input everyone
  nat-pool 1 192.168.130.70 192.168.130.70 netmask 255.255.255.0 pat
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.9 255.255.255.0
  alias 172.28.1.7 255.255.255.0
  peer ip address 172.28.1.8 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input everyone
  nat-pool 1 192.168.130.71 192.168.130.71 netmask 255.255.255.0 pat
  no shutdown

ip route 10.1.0.0 255.255.255.0 192.168.130.254
username admin password 5 $1$wGwStjD1$n2XN6XsZ5uB50mwxvwQHA.  role Admin domain
default-domain

snmp-server community ACE-private group Network-Monitor
snmp-server community ACE-public group Network-Monitor
```

```
snmp-server host 10.1.0.236 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-2_ACE2-1/SH-LOAD# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:23 ---

+++ 09:57:23 6K-2_ACE2-1 ctxExec +++
changeto SH-LOAD



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-LOAD#


6K-2_ACE2-1/SH-LOAD# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:23 ---

+++ 09:57:23 6K-2_ACE2-1 ctxExec +++
changeto SH-Silver



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-Silver#


6K-2_ACE2-1/SH-Silver# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:24 ---

+++ 09:57:24 6K-2_ACE2-1 ctxExec +++
changeto SH-Silver
```

```
NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-Silver# show running-config

Generating configuration....


logging enable
logging standby
logging timestamp
logging trap 5
logging history 7
logging buffered 5
logging monitor 5
logging device-id hostname
logging host 10.86.83.85 udp/514
logging message 251006 level 7
logging message 302022 level 7


tacacs-server key 7 "vwjjzamggu"
tacacs-server host 10.86.83.215 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-silver

crypto chaingroup CHAIN1
  cert term.pem
  cert init.pem
  cert end-to-end.pem
aaa authentication login default group sh-silver local
aaa accounting default group sh-silver local
aaa authentication login error-enable

access-list acl1 line 8 extended permit ip host 172.28.1.6 host 172.28.1.23
access-list eveyone line 10 extended permit ip any any


probe http GEN_HTTP
  interval 5
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_CLEAR"
probe https GEN_HTTPS
  interval 5
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "ACE_SSL"


parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 30
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
  urgent-flag clear
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version SSL3
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  version SSL3
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  session-cache timeout 180
  version TLS1
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow

action-list type modify http Header-Deletions
  header insert response Cache-Control header-value "max-age=901"
  header delete response Etag
  header delete response ETag
  header delete response Pragma
  header delete response Set-Cookie2
action-list type modify http HTTP-HTTPS-Rewrite
  header insert response Cache-Control header-value "max-age=901"
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header insert request Pragma header-value "no-cache, no-cache is the only pragma value I
have ever seen"
  header delete both Content-Type
  header delete response ETag
  header delete response Set-Cookie
  header rewrite response Keep-Alive header-value "timeout=150" replace "timeout=300"
  header rewrite request User-Agent header-value ".*curl.*" replace "Mozilla/4.0
(compatible; cURL; Runs on Linux)"
  ssl url rewrite location "192.168.130.11"
  ssl url rewrite location "www.ssl-term-ace.com"
action-list type modify http Header-Deletions-Cache=2592008
  header insert response Cache-Control header-value "max-age=2592008"
  header delete response Etag
  header delete response ETag
  header delete response Pragma
  header delete response Set-Cookie2
action-list type modify http HTTP-HTTPS-Rewrite-E2E
  header insert response Cache-Control header-value "max-age=901"
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header delete both Content-Type
  header delete response ETag
  header delete response Set-Cookie
  ssl url rewrite location "192.168.130.13" sslport 444
  ssl url rewrite location "www.ssl-end-to-end-ace.com" sslport 444

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice
rserver redirect http-to-https
  webhost-redirection https://192.168.130.11
  inservice

ssl-proxy service ACE_AS_CLIENT
  ssl advanced-options PARM_ACE_AS_CLIENT
ssl-proxy service ACE_END_TO_END
  key pkey.pem
  cert end-to-end.pem
  ssl advanced-options PARM_ACE_TERM
ssl-proxy service ACE_TERM
  key pkey.pem
  cert term.pem
  ssl advanced-options PARM_ACE_TERM_EXPORT_CIPHERS

serverfarm host ACE_END_TO_END_SERVERS_SSL
  description SERVERS FOR END TO END SSL TESTING
  failaction purge
  predictor leastconns
  probe GEN_HTTPS
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_INIT_SERVERS_SSL
  description SERVERS FOR SSL INIT TESTING
  failaction purge
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-IIS-2 443
    inservice
  rserver BRG-IIS-3 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-2 443
    inservice
  rserver BRG-LINUX-3 443
    inservice
serverfarm host ACE_TERM_SERVERS_CLEAR
  description SERVERS FOR SSL TERM TESTING
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host L4
  probe GEN_HTTP
  rserver BRG-IIS-1
    inservice
  rserver BRG-LINUX-1
    inservice
serverfarm host L7
  rserver BRG-IIS-2
    inservice
  rserver BRG-LINUX-2
```

```
    inservice
serverfarm host NON-SSL-TEST
  description SERVERS FOR NON SSL TESTING
  failaction purge
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2 80
    inservice
  rserver BRG-IIS-3 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-2 80
    inservice
  rserver BRG-LINUX-3 80
    inservice
serverfarm host ONE-IIS-SERVER
  rserver BRG-IIS-1
    inservice
serverfarm redirect REDIRECT-GET-SLASH
  rserver http-to-https
    inservice
serverfarm host TCP-NORM-FARM
  predictor leastconns
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-IIS-3
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver BRG-LINUX-3
    inservice
serverfarm host TELNET-NORM-TEST
  rserver BRG-LINUX-1
    inservice

sticky http-cookie SSL_TERM_COOKIE GROUP_10
  cookie insert browser-expire
  serverfarm ACE_TERM_SERVERS_CLEAR
sticky http-cookie SSL_INIT_COOKIE GROUP_20
  cookie insert browser-expire
  serverfarm ACE_INIT_SERVERS_SSL
sticky http-cookie SSL_END_TO_END_COOKIE GROUP_30
  cookie insert browser-expire
  timeout 30
  serverfarm ACE_END_TO_END_SERVERS_SSL
sticky http-cookie NON_SSL_TESTING GROUP_40
  cookie insert browser-expire
  serverfarm NON-SSL-TEST
sticky ip-netmask 255.255.255.0 address both NEW_GROUP
  serverfarm NON-SSL-TEST

class-map match-all GENERIC
class-map match-all L4-VIP_20:80
class-map match-all NON-SSL_TEST
  description NON-SSL_TEST
class-map match-all SSL_END_TO_END_13
  description STICKY FOR SSL TESTING
class-map match-all TCP-NORM-TEST
```

```
  description TCP NORM TEST
  2 match virtual-address 192.168.130.17 tcp eq 22

policy-map type management first-match POLICY_MGMT

policy-map type loadbalance first-match GENERIC
policy-map type loadbalance first-match NON_SSL_TESTING
policy-map type loadbalance first-match PLBSF_L4
  class class-default
    serverfarm L4
policy-map type loadbalance first-match POLICY_SSL_END_TO_END
policy-map type loadbalance first-match POLICY_SSL_INIT
policy-map type loadbalance first-match POLICY_SSL_TERM
policy-map type loadbalance first-match TCP-NORM-TESTING
policy-map type loadbalance first-match TELNET-NORM_TESTING
  class class-default
    serverfarm TELNET-NORM-TEST

policy-map type inspect http all-match INSPECT_GOOD_HTTP

policy-map multi-match SSL_TEST_SUITE_VIPS
  class SSL_END_TO_END_13
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server ACE_END_TO_END
    connection advanced-options TCP_PARAM
  class NON-SSL_TEST
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
    connection advanced-options TCP_PARAM
  class TCP-NORM-TEST
    loadbalance vip inservice
    loadbalance policy TCP-NORM-TESTING
    loadbalance vip icmp-reply
    nat dynamic 1 vlan 281
    connection advanced-options NORMALIZE_MY_TCP_TRAFFIC
  class GENERIC
    nat dynamic 1 vlan 281
    appl-parameter http advanced-options HTTP_PARAM
  class L4-VIP_20:80
    nat dynamic 1 vlan 281

interface vlan 130
  ip address 192.168.130.3 255.255.255.0
  alias 192.168.130.1 255.255.255.0
  peer ip address 192.168.130.2 255.255.255.0
  fragment min-mtu 80
  access-group input eveyone
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown
interface vlan 281
  ip address 172.28.1.6 255.255.255.0
  alias 172.28.1.4 255.255.255.0
  peer ip address 172.28.1.5 255.255.255.0
  fragment min-mtu 80
  access-group input eveyone
  nat-pool 1 192.168.130.201 192.168.130.201 netmask 255.255.255.0 pat
  service-policy input POLICY_MGMT
  service-policy input SSL_TEST_SUITE_VIPS
  no shutdown

domain SH-Silver-Domain
  add-object all
```

```
ip route 10.1.0.0 255.255.255.0 192.168.130.254
ip route 192.168.120.0 255.255.255.0 192.168.130.254
ip route 10.3.0.0 255.255.255.0 192.168.130.254
username admin password 5 $1$5SD7.E4T$/xUH04GK/gFCx8PdOKZ.L/  role Admin domain
default-domain
username vrtnetmonsilver password 5 $1$SGEScCRK$qW1k3UCNiR/jIFdcYsIZx.  role
Network-Monitor domain SH-Silver-Domain
username vrtadmsilver password 5 $1$c64ET9q4$fYPmJa5OVQvntr1quXN0O.  role Admin domain
SH-Silver-Domain
username vrtjohndoe password 5 $1$Ej3RvPBx$aMHRl4SocCBTriiB9nOmf/  role Admin domain
SH-Silver-Domain
username netmon password 5 $1$hPvrGd2M$Usu6WSw9RdIFxP.qec0vp1  role Network-Monitor domain
default-domain




6K-2_ACE2-1/SH-Silver# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:24 ---

+++ 09:57:24 6K-2_ACE2-1 ctxExec +++
changeto SH-Silver



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/SH-Silver#


6K-2_ACE2-1/SH-Silver# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:24 ---

+++ 09:57:24 6K-2_ACE2-1 ctxExec +++
changeto SH-Bridge


6K-2_ACE2-1/SH-Bridge#


6K-2_ACE2-1/SH-Bridge# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
```

```
--- 09:57:24 ---

+++ 09:57:24 6K-2_ACE2-1 ctxExec +++
changeto SH-Bridge


6K-2_ACE2-1/SH-Bridge# show running-config

Generating configuration....


logging enable
logging standby
logging console 5
logging timestamp
logging trap 5
logging history 5
logging buffered 5
logging device-id context-name
logging host 10.1.0.242 udp/514
logging host 10.1.0.236 udp/514


tacacs-server key 7 "vwjjzamggu"
tacacs-server host 172.29.0.235 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.236 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.237 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-bridge
  server 172.29.0.236
  server 172.29.0.237

arp 192.168.120.80 00.00.11.11.22.22
arp 172.29.0.200 00.00.22.22.44.44
arp learned-interval 60
aaa authentication login default group sh-bridge local
aaa accounting default group sh-bridge
access-list BPDU-ALLOW ethertype permit bpdu

access-list ICMP-ONLY line 8 extended permit icmp any any
access-list NAT_ACCESS line 20 extended permit tcp any host 172.28.3.121
access-list NAT_ACCESS line 30 extended permit tcp any host 172.28.3.122
access-list NAT_ACCESS line 40 extended permit tcp any host 172.28.3.140
access-list anyone-ip line 10 extended permit ip any any
access-list anyone-tcp line 10 extended permit tcp any any

script file 1 FTP_PROBE_SCRIPT
script file 2 TFTP_PROBE
script file 3 LDAP_PROBE


probe icmp FA-PURGE-ICMP
  ip address 172.28.4.253 routed
  interval 5
  passdetect interval 2
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
probe ftp FTP
  interval 10
```

```
    faildetect 2
    passdetect interval 10
    receive 5
    expect status 220 220
    open 3
  probe icmp HA-ICMP
    interval 2
    faildetect 2
    passdetect interval 2
  probe tcp HA-TCP:1755
    port 1755
    interval 2
    faildetect 2
    passdetect interval 2
  probe tcp HA-TCP:554
    port 554
    interval 2
    faildetect 2
    passdetect interval 2
  probe http HTTP
    interval 5
    passdetect interval 10
    receive 5
    expect status 200 200
    open 3
  probe icmp ICMP
    interval 10
    faildetect 2
    passdetect interval 10
  probe dns PRB-DNS1
    description "all good addresses"
    interval 5
    passdetect interval 2
    domain www1.safeharbor.com
    expect address 1.1.1.1
    expect address 1.1.1.2
    expect address 1.1.1.3
  probe dns PRB-DNS2
    description "one good address"
    interval 5
    passdetect interval 2
    domain www2.safeharbor.com
    expect address 2.1.1.1
  probe dns PRB-DNS3
    description "2 good addresses, bumpy case"
    interval 5
    passdetect interval 2
    domain WwW3.SaFeHaRbOr.CoM
    expect address 3.1.1.3
    expect address 3.1.1.2
  probe dns PRB-DNS4
    description "one good and one bad address"
    interval 5
    passdetect interval 2
    domain www4.safeharbor.com
    expect address 192.168.1.4
    expect address 4.1.1.1
  probe dns PRB-DNS5
    description "all bad addresses"
    interval 5
    passdetect interval 2
    domain www4.safeharbor.com
    expect address 192.168.1.5
    expect address 192.168.1.6
    expect address 1.1.1.1
  probe dns PRB-DNS6:2222
    description "dns not running on this port, but addresses good"
    port 2222
    interval 5
    passdetect interval 2
    domain www1.safeharbor.com
    expect address 1.1.1.1
    expect address 1.1.1.2
    expect address 1.1.1.3
  probe http PRB-HTTP:84
    port 84
    interval 5
    passdetect interval 4
    passdetect count 10
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:85
    description Server RST 1byte data
    port 85
    interval 5
    passdetect interval 2
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:86
    description Server RST 3200byte data
    port 86
    interval 5
    passdetect interval 2
    expect status 200 200
    connection term forced
  probe http PRB-HTTP:87
    description Server FIN 1byte data
    port 87
    interval 5
    passdetect interval 2
    expect status 200 200
  probe http PRB-HTTP:88
    description Server FIN 3200byte data
    port 88
    interval 5
    passdetect interval 2
    expect status 200 200
  probe https PRB-SSL:443
    interval 5
    passdetect interval 10
    request method get url /index.txt
    expect status 200 200
    header Via header-value "PRB-SSL:443 Probe Header"
    hash
  probe icmp PRED-PING
    ip address 172.28.4.243 routed
    interval 5
    faildetect 2
    passdetect interval 2
  probe radius RADIUS
    interval 2
    faildetect 2
    passdetect interval 2
    credentials lab labtest1 secret ace
  probe scripted SCRIPT_FTP:21
    interval 10
    passdetect interval 2
    passdetect count 2
    receive 5
    script FTP_PROBE_SCRIPT /home/lab/ftp-files/file01.log lab labtest1
  probe scripted SCRIPT_LDAP
    interval 10
    passdetect interval 5
    passdetect count 2
    receive 5
    script LDAP_PROBE
  probe scripted SCRIPT_TFTP
    interval 10
    passdetect interval 5
    passdetect count 2
    receive 5
    script TFTP_PROBE "large file name to test the tftp scripted probe.exe"
  probe https SSL
    interval 5
    passdetect interval 10
    expect status 200 299
    header Via header-value "ACE_Gold_SSL"
    connection term forced
  probe https SSL-445:FIN
    port 445
    interval 5
    passdetect interval 10
    expect status 200 200
  probe https SSL-445:RST
    port 445
    interval 5
    passdetect interval 10
    expect status 200 200
    connection term forced
  probe tcp TCP
    interval 5
    faildetect 2
    passdetect interval 10
    open 3
  probe udp UDP
    interval 5
    passdetect interval 2
  probe udp UDP:2222
    port 2222
    interval 5
    passdetect interval 2


  parameter-map type connection 120SECOND-IDLE
    set timeout inactivity 120
    set tcp timeout half-closed 30
  parameter-map type connection 1SECOND-IDLE
    set timeout inactivity 1
  parameter-map type connection 2SECOND-IDLE
    set timeout inactivity 2
  parameter-map type connection 60SECOND-IDLE
    set timeout inactivity 60
    set tcp timeout half-closed 30
  parameter-map type http COOKIE-DELIM
    persistence-rebalance
    set secondary-cookie-delimiters @$
  parameter-map type http COOKIE-INSERT-HDR-PARSE
    persistence-rebalance
    set header-maxparse-length 4000
  parameter-map type ssl EXPORT_CIPHERS
    cipher RSA_EXPORT_WITH_RC4_40_MD5
    cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
    cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
    cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
    cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
  parameter-map type connection HALF-CLOSE_4s
    set tcp timeout half-closed 4
  parameter-map type connection INFINITE-IDLE
    set timeout inactivity 0
    set tcp timeout half-closed 30
  parameter-map type connection LLFlows
    set timeout inactivity 0
  parameter-map type connection NORM
    reserved-bits clear
  parameter-map type connection NORM_TCP
    tcp-options timestamp allow
    reserved-bits drop
    syn-data drop
    urgent-flag clear
  parameter-map type http PARSE_LENGTH
    persistence-rebalance
  parameter-map type http PERSIST-REBAL-4K
    persistence-rebalance
  parameter-map type http PERSIST-REBALANCE
    persistence-rebalance
  parameter-map type connection PRED-CONNS-UDP_CONN
    set timeout inactivity 300
  parameter-map type ssl RC4_128_MD5_CIPHER
    cipher RSA_WITH_RC4_128_MD5
    version TLS1
  parameter-map type http REUSE-REBAL
    server-conn reuse
    persistence-rebalance
  parameter-map type ssl STRONG_CIPHERS
    cipher RSA_WITH_3DES_EDE_CBC_SHA
    cipher RSA_WITH_AES_128_CBC_SHA priority 2
    cipher RSA_WITH_AES_256_CBC_SHA priority 3
  parameter-map type ssl WEAK_CIPHERS
    cipher RSA_WITH_RC4_128_MD5
    cipher RSA_WITH_RC4_128_SHA priority 2
    cipher RSA_WITH_DES_CBC_SHA priority 3
  parameter-map type connection wan-opt
    set tcp wan-optimization rtt 0

  rserver host BRG-11
    ip address 172.28.3.11
    inservice
  rserver host BRG-12
    ip address 172.28.3.12
    inservice
  rserver host BRG-13
    ip address 172.28.3.13
    inservice
  rserver host BRG-14
    ip address 172.28.3.14
    inservice
  rserver host BRG-15
    ip address 172.28.3.15
    inservice
  rserver host LOCAL-151
    ip address 172.28.3.151
    inservice
  rserver host LOCAL-152
    ip address 172.28.3.152
    inservice
  rserver host LOCAL-153
```

```
    ip address 172.28.3.153
    inservice
  rserver host LOCAL-154
    ip address 172.28.3.154
    inservice
  rserver redirect REDIRECT-100K
    webhost-redirection http://172.28.3.132/redirect-100k.html 302
    inservice
  rserver redirect REDIRECT-10K
    webhost-redirection http://172.28.3.133/redirect-10k.html 302
    inservice
  rserver redirect REDIRECT-1K
    webhost-redirection http://172.28.3.134/redirect-1k.html 302
    inservice
  rserver host RT-239-FRAGRTR
    ip address 172.28.4.239
    inservice
  rserver host RT-240
    ip address 172.28.4.240
    inservice
  rserver host RT-241
    ip address 172.28.4.241
    inservice
  rserver host RT-242
    ip address 172.28.4.242
    inservice
  rserver host RT-243
    ip address 172.28.4.243
    inservice
  rserver host RT-244
    ip address 172.28.4.244
    inservice
  rserver host RT-245
    ip address 172.28.4.245
    inservice
  rserver host WEIGHT-80
    ip address 10.1.0.235
    weight 80
    inservice

  serverfarm host ADD-REM-SRV
    predictor leastconns
    probe TCP
    rserver BRG-13
      inservice
    rserver BRG-14
      inservice
    rserver LOCAL-153
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-243
      inservice
    rserver RT-244
      inservice
  serverfarm host COOKIE
    probe ICMP
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver RT-244
      inservice
  serverfarm host COOKIE-HASH
    predictor leastconns
    rserver RT-240
      inservice
    rserver RT-240 90
      inservice
    rserver RT-241
      inservice
    rserver RT-241 90
      inservice
    rserver RT-242
      inservice
    rserver RT-242 90
      inservice
    rserver RT-243
      inservice
    rserver RT-243 90
      inservice
  serverfarm host COOKIE-INSERT
    rserver BRG-14
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-245
      inservice
  serverfarm host COOKIE-INSERT2
    probe HTTP
    rserver BRG-14
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-245
      inservice
  serverfarm host COOKIE1
    probe HTTP
    rserver BRG-12
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-240
      inservice
  serverfarm host COOKIE2
    probe TCP
    rserver BRG-11
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-241
      inservice
  serverfarm host CS-COOKIES
    probe TCP
    rserver BRG-12
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-240
      inservice
  serverfarm host CS-MOZILLA
    rserver LOCAL-151
      inservice
    rserver RT-240
      inservice
  serverfarm host CS-MSIE
    rserver RT-242
      inservice
    rserver RT-243
      inservice
  serverfarm host DEFAULT
    probe ICMP
    rserver BRG-15
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-245
      inservice
  serverfarm host FA-PURGE
    failaction purge
    rserver BRG-11
      inservice
    rserver BRG-14
      probe FA-PURGE-ICMP
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-154
      probe FA-PURGE-ICMP
      inservice
    rserver RT-242
      inservice
    rserver RT-244
      probe FA-PURGE-ICMP
      inservice
  serverfarm host FTP
    probe FTP
    rserver BRG-11 21
      inservice
    rserver BRG-12 21
      inservice
    rserver LOCAL-151 21
      inservice
    rserver LOCAL-152 21
      inservice
    rserver RT-240 21
      inservice
    rserver RT-241 21
      inservice
  serverfarm host GEN-443
    probe SSL
    rserver BRG-12
      inservice
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm host GEN-80
    probe TCP
    rserver BRG-12
      inservice
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-152
```

```
      inservice
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm host GEN-FTP
    probe FTP
    rserver BRG-13
      inservice
    rserver LOCAL-152
      inservice
    rserver RT-240
      inservice
  serverfarm host GEN-UDP
    probe ICMP
    rserver BRG-11
      inservice
    rserver LOCAL-151
      inservice
    rserver RT-244
      inservice
  serverfarm host GEN2-80
    probe TCP
    rserver BRG-11
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-241
      inservice
  serverfarm host HDR-IXIA
    rserver BRG-14
      inservice
    rserver RT-241
      inservice
  serverfarm host HEADER
    probe HTTP
    rserver LOCAL-152
      inservice
    rserver RT-240
      inservice
    rserver RT-244
      inservice
  serverfarm host HEADER-INSERT
    rserver BRG-13
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-153
      inservice
  serverfarm host HEADER-INSERT2
    rserver BRG-12
      inservice
    rserver BRG-14
      inservice
    rserver LOCAL-153
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-240
      inservice
    rserver RT-241
      inservice
    rserver RT-244
      inservice
    rserver RT-245
      inservice
  serverfarm host ICMP
    probe ICMP
    rserver BRG-11
      inservice
    rserver LOCAL-152
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-241
      inservice
    rserver RT-242
      inservice
  serverfarm host ICMP2
    rserver BRG-11 7777
      inservice
    rserver LOCAL-152
      inservice
    rserver LOCAL-153 7777
      inservice
    rserver RT-241
      inservice
    rserver RT-242 7777
      inservice
  serverfarm host IDLE-TCP
    probe TCP
    rserver BRG-15
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-244
      inservice
  serverfarm host IDLE-UDP
    probe UDP:2222
    rserver BRG-15
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-244
      inservice
  serverfarm host LDAP
    rserver BRG-15
      inservice
    rserver LOCAL-151
      inservice
    rserver RT-244
      inservice
  serverfarm host LENGTHS
    rserver RT-241
      inservice
    rserver RT-244
      inservice
  serverfarm host LENGTHS-2
    rserver RT-240
      inservice
    rserver RT-245
      inservice
  serverfarm host MAX-CONN
    probe HTTP
    rserver LOCAL-151
      conn-limit max 4 min 2
      inservice
  serverfarm host MAX-CONN2
```

```
  probe HTTP
  rserver LOCAL-151
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 90
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 91
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 92
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 93
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 94
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-151 95
    conn-limit max 500 min 2
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-243
    conn-limit max 500 min 2
    inservice
serverfarm host NORM
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  rserver BRG-11
    inservice
  rserver LOCAL-153
    inservice
  rserver RT-241
    inservice
serverfarm host NORM2_L7
  failaction purge
  predictor leastconns slowstart 10
  probe TCP
  retcode 100 599 check count
  rserver BRG-11 80
    inservice
  rserver LOCAL-153 80
    inservice
  rserver RT-241 80
    inservice
serverfarm host PERSISTENT
  rserver LOCAL-154
    inservice
  rserver RT-240
    inservice
  rserver RT-242
    inservice
  rserver RT-243
    inservice
serverfarm host PRED-CONNS
  predictor leastconns
  rserver BRG-11
    inservice
  rserver BRG-12
    inservice
  rserver BRG-13
```

```
      inservice
    rserver BRG-14
      inservice
    rserver BRG-15
      inservice
    rserver LOCAL-151
      inservice
    rserver LOCAL-152
      inservice
    rserver LOCAL-153
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-240
      inservice
    rserver RT-241
      inservice
    rserver RT-242
      inservice
    rserver RT-243
      inservice
    rserver RT-244
      inservice
  serverfarm host PRED-CONNS-UDP
    failaction purge
    predictor leastconns
    rserver BRG-11 2222
      inservice
    rserver LOCAL-151 2222
      inservice
    rserver LOCAL-153 2222
      inservice
    rserver LOCAL-154 2222
      inservice
    rserver RT-240 2222
      inservice
    rserver RT-242 2222
      inservice
    rserver RT-244 2222
      probe PRED-PING
      inservice
  serverfarm host PREDICTOR
    predictor leastconns
    probe TCP
    rserver BRG-13
      inservice
    rserver BRG-14
      inservice
    rserver LOCAL-152
      inservice
    rserver LOCAL-153
      inservice
    rserver RT-243
      inservice
    rserver RT-244
      inservice
  serverfarm host PROBES
    predictor leastconns
    rserver BRG-13
      inservice
    rserver LOCAL-154
      inservice
    rserver RT-244
      inservice
```

```
serverfarm host PROBES-2
  predictor leastconns
  rserver BRG-11
    inservice
  rserver LOCAL-152
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-241
    inservice
  rserver RT-244
    inservice
serverfarm host PROBES-MANY
  predictor leastconns
  probe SCRIPT_TFTP
  rserver BRG-11
    probe RADIUS
    inservice
  rserver LOCAL-152
    inservice
  rserver LOCAL-154
    probe RADIUS
    inservice
  rserver RT-241
    inservice
  rserver RT-243
    inservice
  rserver RT-244
    probe RADIUS
    inservice
serverfarm host RADIUS
  rserver BRG-11
    inservice
  rserver LOCAL-151
    inservice
  rserver RT-244
    inservice
serverfarm host RED-ALL-SVRS
  rserver RT-240
    inservice
  rserver RT-241
    inservice
serverfarm host REDIRECT
  rserver RT-242
    inservice
  rserver RT-243
    inservice
  rserver RT-244
    inservice
  rserver RT-245
    inservice
serverfarm redirect REDIRECT-100K
  rserver REDIRECT-100K
    inservice
serverfarm redirect REDIRECT-10K
  rserver REDIRECT-10K
    inservice
serverfarm redirect REDIRECT-1K
  rserver REDIRECT-1K
    inservice
serverfarm host RHI
  rserver BRG-11
    inservice
  rserver LOCAL-154
```

```
    inservice
  rserver RT-241
    inservice
serverfarm host SORRY
  rserver RT-240
    inservice
serverfarm host SORRY-BACK
  rserver LOCAL-151
    inservice
  rserver RT-243
    inservice
serverfarm host STICKY-COOKIE
  probe ICMP
  rserver BRG-11
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-241
    inservice
  rserver RT-242
    inservice
serverfarm host STICKY-HEADER
  probe HTTP
  rserver BRG-12
    inservice
  rserver LOCAL-151
    inservice
  rserver RT-243
    inservice
serverfarm host STICKY-HEADER2
  probe HTTP
  rserver BRG-13
    inservice
  rserver LOCAL-152
    inservice
  rserver RT-244
    inservice
serverfarm host STICKY-NETMASK
  probe ICMP
  rserver BRG-12
    inservice
  rserver LOCAL-153
    inservice
  rserver RT-242
    inservice
serverfarm host TCP-REUSE
  rserver BRG-15
    inservice
  rserver LOCAL-154
    inservice
  rserver RT-245
    inservice
serverfarm host UDP
  probe UDP
  rserver BRG-11
    inservice
  rserver LOCAL-151
    inservice
  rserver RT-241
    inservice
serverfarm host URL-MAP-128K
  rserver RT-241
    inservice
  rserver RT-242
```

```
    inservice
serverfarm host URL-MAP-16K
  rserver BRG-12
    inservice
  rserver RT-242
    inservice
serverfarm host URL-MAP-32K
  rserver RT-244
    inservice
  rserver RT-245
    inservice
serverfarm host URL-MAP-512K
  rserver RT-242
    inservice
  rserver RT-243
    inservice
serverfarm host URL-MAP-64K
  rserver RT-242 91
    inservice
  rserver RT-244
    inservice
serverfarm host URL-MAPS
  rserver RT-241
    inservice
  rserver RT-242
    inservice
  rserver RT-244
    inservice
serverfarm host WEIGHT
  probe HTTP
  rserver BRG-11
    weight 10
    inservice
  rserver LOCAL-152
    weight 40
    inservice
  rserver RT-240
    weight 20
    inservice
  rserver RT-243
    weight 30
    inservice

sticky ip-netmask 255.255.255.255 address source STKY-GRP-30
  timeout 40
  replicate sticky
  serverfarm GEN-80
sticky http-cookie cookie-gold-grp40 STKY-GRP-40
  cookie insert browser-expire
  timeout 1
  replicate sticky
  serverfarm GEN2-80
sticky ip-netmask 255.255.255.255 address both STKY-GRP-31
  timeout 40
  replicate sticky
  serverfarm GEN-UDP
sticky ip-netmask 255.255.255.255 address both STKY-GRP-32
  timeout 40
  replicate sticky
  serverfarm GEN-FTP
sticky http-cookie COOKIE_TEST COOKIE-GROUP
  cookie secondary URLCOOKIE
  timeout 40
  replicate sticky
  serverfarm STICKY-COOKIE
  16 static cookie-value "PATRIOTS0" rserver LOCAL-151
sticky http-header MSISDN HEADER-GROUP-42
  timeout 30
  replicate sticky
  serverfarm STICKY-HEADER
sticky http-header TestHeader HEADER-GROUP-41
  header offset 15 length 7
  timeout 30
  replicate sticky
  serverfarm STICKY-HEADER2
sticky http-cookie COOKIE_INSERT COOKIE-INSERT-GROUP-45
  cookie insert
  timeout 1
  replicate sticky
  serverfarm COOKIE-HASH
sticky http-cookie COOKIE_TEST COOKIE-MAP-GROUP
  replicate sticky
  serverfarm CS-COOKIES
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-29
  timeout 30
  replicate sticky
  serverfarm MAX-CONN
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-11
  timeout 30
  replicate sticky
  serverfarm URL-MAP-16K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-13
  timeout 30
  replicate sticky
  serverfarm URL-MAP-64K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-12
  timeout 30
  replicate sticky
  serverfarm URL-MAP-32K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-14
  timeout 30
  replicate sticky
  serverfarm URL-MAP-128K
sticky ip-netmask 255.255.255.255 address both STICKY-GROUP-15
  timeout 30
  replicate sticky
  serverfarm URL-MAP-512K
sticky http-cookie Safeharbor-Cookie1 COOKIE-GROUP-42
  cookie insert
  timeout 30
  replicate sticky
sticky http-cookie Cookie-Insert-Name-maxsize-is-63-bytes-abcdefghilmnopqr-63bytes
COOKIE-INSERT-GROUP-46
  cookie insert
  timeout 1
  replicate sticky
  serverfarm COOKIE-INSERT
sticky http-cookie COOKIE_TEST STKY-GRP-43
  replicate sticky
  serverfarm PREDICTOR
sticky ip-netmask 255.255.255.255 address both STKY-GRP-33
  timeout 20
  replicate sticky
  serverfarm STICKY-NETMASK

class-map type http loadbalance match-all 128K-FORWARDING
  2 match http url .*128k.*
class-map type http loadbalance match-all 16K-FORWARDING
```

```
  2 match http url .*16k.*
class-map type http loadbalance match-all 32K-FORWARDING
  2 match http url .*32k.*
class-map type http loadbalance match-all 512K-FORWARDING
  2 match http url .*512k.*
class-map type http loadbalance match-all 64K-FORWARDING
  2 match http url .*64k.*
class-map match-all ADD-REM-SRV-VIP_110:80
class-map type http loadbalance match-all BROWSER_MOZILLA
  2 match http header User-Agent header-value ".*Mozilla.*"
class-map type http loadbalance match-all BROWSER_MOZILLA40
  2 match http header User-Agent header-value ".*Mozilla/4.0.*"
class-map type http loadbalance match-all BROWSER_MOZILLA50
  2 match http header User-Agent header-value ".*Mozilla/5.0.*"
class-map type http loadbalance match-all BROWSER_MSIE
  2 match http header User-Agent header-value ".*MSIE.*"
class-map match-all COOKIE-HASH-VIP_10.20.30.40:80
class-map match-all COOKIE-MAP:80
class-map match-all FA-PURGE-VIP_113:ANY
class-map type ftp inspect match-any FTP-L7-MAX-DENY2
  2 match request-method syst
class-map type ftp inspect match-any FTP-L7-MIN-DENY
  2 match request-method mkd
  3 match request-method rmd
class-map match-all FTP-VIP-NAT_119
  2 match destination-address 172.28.3.119 255.255.255.255
class-map match-all FTP-VIP_119:1111
  2 match access-list ICMP-ONLY
class-map match-all ICMP-UDP-VIP_138
class-map match-all LENGTHS-VIP_136:80
  description remote-access-traffic-match
class-map type http loadbalance match-all MSISDN
  2 match http header MSISDN header-value ".*"
class-map match-any NAT_CLASS
  2 match access-list NAT_ACCESS
class-map match-any NORM-VIP_142:80
  2 match any
class-map type http loadbalance match-any P-COOKIE-INS
  2 match http url /index.html* method GET
class-map type http loadbalance match-any P-COOKIE-INS2
  2 match http url .*
class-map type http loadbalance match-all P-HDR-INSERT
  2 match http url .*
class-map type http loadbalance match-all P-HDR-IXIA
  2 match http url .*
class-map type http loadbalance match-all P-HDR-SRCDST-IP
  2 match http url .*
class-map match-all PERSISTENT-VIP_131:80
class-map type http loadbalance match-all REDIRECT-10K
  2 match http url .*redirect-10k.html
class-map type http loadbalance match-all REDIRECT-1K
  2 match http url .*redirect-1k.html
class-map match-all REDIRECT-VIP_135:80
class-map match-all UDP-VIP_114:UDP
class-map match-all URL*_L7
class-map match-all URL-MAPS-VIP_130:80
class-map type http loadbalance match-all URLCOOKIE-MAP2
  2 match http cookie secondary URLCOOKIE cookie-value "VALUE2"
class-map match-all WEIGHT_112:80
  2 match virtual-address 172.28.3.112 tcp eq www

policy-map type management first-match P-MGT

policy-map type loadbalance first-match FTP-LB-SF_FTP
```

```
  class class-default
    serverfarm FTP
policy-map type loadbalance first-match MAX-CONN-LB-SF_MAX-CONN2
policy-map type loadbalance first-match PLBSF-FTP-test
  class class-default
policy-map type loadbalance first-match PLBSF_ADD-REM-SRV
  class class-default
    serverfarm ADD-REM-SRV
policy-map type loadbalance first-match PLBSF_COOKIE-HASH
  class class-default
    sticky-serverfarm COOKIE-INSERT-GROUP-45
policy-map type loadbalance first-match PLBSF_COOKIE-INSERT
  class P-COOKIE-INS
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Source-IP header-value "%is"
    insert-http Destination_IP header-value "%id"
  class P-COOKIE-INS2
    sticky-serverfarm COOKIE-INSERT-GROUP-46
    insert-http Source-IP header-value "%is"
    insert-http Destination_IP header-value "%id"
policy-map type loadbalance first-match PLBSF_COOKIE-MAP
  class URLCOOKIE-MAP2
    serverfarm COOKIE2
policy-map type loadbalance first-match PLBSF_FA-PURGE
  class class-default
    serverfarm FA-PURGE
policy-map type loadbalance first-match PLBSF_GEN-443
  class class-default
    serverfarm GEN-443
policy-map type loadbalance first-match PLBSF_GEN-80
  class class-default
    sticky-serverfarm STKY-GRP-30
policy-map type loadbalance first-match PLBSF_GEN-FTP
  class class-default
    sticky-serverfarm STKY-GRP-32
policy-map type loadbalance first-match PLBSF_GEN-UDP
  class class-default
    sticky-serverfarm STKY-GRP-31
policy-map type loadbalance first-match PLBSF_HDR-IXIA
  class P-HDR-IXIA
    serverfarm HDR-IXIA
policy-map type loadbalance first-match PLBSF_HEADER
policy-map type loadbalance first-match PLBSF_HEADER-INSERT
  class P-HDR-INSERT
    serverfarm HEADER-INSERT
    insert-http Destination_iP header-value "%id"
    insert-http Pragma header-value "Pragma no Pragma that is the question"
    insert-http Accept header-value "anything"
    insert-http Source-IP header-value "%is"
policy-map type loadbalance first-match PLBSF_HEADER-INSERT2
  class P-HDR-SRCDST-IP
    serverfarm HEADER-INSERT2
    insert-http Destination_iP header-value "%id"
    insert-http Source-IP header-value "%is"
policy-map type loadbalance first-match PLBSF_ICMP
policy-map type loadbalance first-match PLBSF_ICMP2
  class class-default
    serverfarm ICMP2
policy-map type loadbalance first-match PLBSF_IDLE-TCP
  class class-default
    serverfarm IDLE-TCP
policy-map type loadbalance first-match PLBSF_IDLE-UDP
  class class-default
    serverfarm IDLE-UDP
```

```
policy-map type loadbalance first-match PLBSF_LENGTHS
policy-map type loadbalance first-match PLBSF_MAX-CONN
  class class-default
    serverfarm MAX-CONN2
policy-map type loadbalance first-match PLBSF_NORM
  class class-default
    serverfarm NORM
policy-map type loadbalance first-match PLBSF_NORM2_L7
  class class-default
    serverfarm NORM2_L7
policy-map type loadbalance first-match PLBSF_PERSISTENT
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm PERSISTENT
policy-map type loadbalance first-match PLBSF_PRED-CONNS
  class class-default
    serverfarm PRED-CONNS
policy-map type loadbalance first-match PLBSF_PRED-CONNS-UDP
  class class-default
    serverfarm PRED-CONNS-UDP
policy-map type loadbalance first-match PLBSF_PREDICTOR
  class class-default
    serverfarm PREDICTOR
policy-map type loadbalance first-match PLBSF_RED-ALL-SVRS
  class class-default
    serverfarm RED-ALL-SVRS
policy-map type loadbalance first-match PLBSF_REDIRECT
  class REDIRECT-1K
    serverfarm REDIRECT-1K
  class REDIRECT-10K
    serverfarm REDIRECT-10K
policy-map type loadbalance first-match PLBSF_RHI
policy-map type loadbalance first-match PLBSF_SORRY
  class class-default
    serverfarm SORRY backup SORRY-BACK
policy-map type loadbalance first-match PLBSF_STICKY-COOKIE
policy-map type loadbalance first-match PLBSF_STICKY-HEADER
  class MSISDN
    sticky-serverfarm HEADER-GROUP-42
  class class-default
    serverfarm DEFAULT
policy-map type loadbalance first-match PLBSF_STICKY-NETMASK
  class class-default
    sticky-serverfarm STKY-GRP-33
policy-map type loadbalance first-match PLBSF_TCP-REUSE
policy-map type loadbalance first-match PLBSF_UDP
  class class-default
    serverfarm UDP
policy-map type loadbalance first-match PLBSF_URL-MAPS
  class 16K-FORWARDING
    sticky-serverfarm STICKY-GROUP-11
  class 32K-FORWARDING
    sticky-serverfarm STICKY-GROUP-12
  class 64K-FORWARDING
    sticky-serverfarm STICKY-GROUP-13
  class 128K-FORWARDING
    sticky-serverfarm STICKY-GROUP-14
  class 512K-FORWARDING
    sticky-serverfarm STICKY-GROUP-15
  class class-default
    serverfarm URL-MAPS
policy-map type loadbalance first-match PLBSF_WEIGHT
  class class-default
    serverfarm WEIGHT

policy-map type inspect ftp first-match FTP-INSPSF_FTP
  class FTP-L7-MAX-DENY2
    mask-reply

policy-map multi-match NAT_POLICY
  class NAT_CLASS
    nat dynamic 1 vlan 2830
policy-map multi-match NORMALIZATION
policy-map multi-match SH-Gold-VIPs
  class FTP-VIP-NAT_119
    nat dynamic 1 vlan 2830
    nat dynamic 1 vlan 283
  class FTP-VIP_119:1111
  class ADD-REM-SRV-VIP_110:80
    nat dynamic 1 vlan 2830
  class WEIGHT_112:80
    loadbalance vip inservice
    loadbalance policy PLBSF_WEIGHT
    loadbalance vip icmp-reply active
    nat dynamic 1 vlan 2830
    appl-parameter http advanced-options PERSIST-REBALANCE
  class FA-PURGE-VIP_113:ANY
    nat dynamic 1 vlan 2830
    connection advanced-options INFINITE-IDLE
  class UDP-VIP_114:UDP
    nat dynamic 1 vlan 2830
    connection advanced-options 1SECOND-IDLE
  class ICMP-UDP-VIP_138
    nat dynamic 1 vlan 2830
    connection advanced-options 60SECOND-IDLE
  class COOKIE-HASH-VIP_10.20.30.40:80
    appl-parameter http advanced-options PERSIST-REBALANCE
  class URL-MAPS-VIP_130:80
    nat dynamic 1 vlan 2830
    appl-parameter http advanced-options PERSIST-REBALANCE
  class PERSISTENT-VIP_131:80
    nat dynamic 1 vlan 2830
    appl-parameter http advanced-options PERSIST-REBALANCE
  class REDIRECT-VIP_135:80
    appl-parameter http advanced-options PERSIST-REBALANCE
  class LENGTHS-VIP_136:80
    appl-parameter http advanced-options PARSE_LENGTH
  class NORM-VIP_142:80
    nat dynamic 1 vlan 2830
policy-map multi-match SH-Gold-VIPs3

service-policy input P-MGT

interface vlan 99
  ip address 192.168.99.6 255.255.255.0
  peer ip address 192.168.99.5 255.255.255.0
  access-group input anyone-ip
  no shutdown
interface vlan 106
```

```
  ip address 192.168.106.3 255.255.255.0
  peer ip address 192.168.106.2 255.255.255.0
  fragment chain 20
  fragment min-mtu 68
  access-group input anyone-ip
  service-policy input SH-Gold-VIPs3
  no shutdown
interface vlan 283
  description Downstream Bridge VLAN_283-2830
  bridge-group 283
  fragment chain 20
  fragment min-mtu 68
  access-group input BPDU-ALLOW
  access-group input anyone-ip
  nat-pool 1 172.28.3.71 172.28.3.71 netmask 255.255.255.0 pat
  no shutdown
interface vlan 2830
  description Upstream Bridge VLAN_2830-283
  bridge-group 283
  fragment chain 20
  fragment min-mtu 68
  access-group input BPDU-ALLOW
  access-group input anyone-ip
  nat-pool 1 172.28.3.70 172.28.3.70 netmask 255.255.255.0 pat
  service-policy input SH-Gold-VIPs
  service-policy input NAT_POLICY
  no shutdown

interface bvi 283
  ip address 172.28.3.3 255.255.255.0
  alias 172.28.3.1 255.255.255.0
  peer ip address 172.28.3.2 255.255.255.0
  no shutdown

ft track host GW_251-252
  track-host 192.168.16.252
  peer track-host 192.168.16.251
  peer probe HA-ICMP priority 10
  probe HA-ICMP priority 5
  priority 5
  peer priority 110
ft track hsrp HSRP_120
  track-hsrp hsrp-Vl120-120
  peer track-hsrp hsrp-Vl120-120
  priority 5
  peer priority 110
ft track interface Int_4/37_6/13_V99
  track-interface vlan 99
  peer track-interface vlan 99
  priority 5
  peer priority 110
ft track host LOCAL-241
  peer track-host 172.28.4.241
  peer probe HA-TCP:554 priority 50
  peer probe HA-TCP:1755 priority 60

domain SH-Gold-Domain
  add-object all

role SHAdmin
  rule 1 permit create
  rule 2 permit monitor
  rule 3 permit modify
role SHUser
```

```
  rule 1 deny create
  rule 2 permit monitor
  rule 3 deny modify
  rule 4 permit debug
role TEST

ip route 10.1.0.0 255.255.255.0 172.28.3.254
ip route 172.28.4.0 255.255.255.0 172.28.3.253
ip route 172.28.0.0 255.255.254.0 172.28.3.253
ip route 192.168.16.251 255.255.255.255 192.168.106.251
ip route 192.168.16.252 255.255.255.255 192.168.106.252
username admin password 5 $1$hU1iScF8$WmpdK4IcQI2ofTMDm6l.N1  role Admin domain
default-domain
username localadmin password 5 $1$g5rd5HO2$C34zVe3a9f73Dce/WNvbM.  role Admin domain
SH-Gold-Domain default-domain
username localuser password 5 $1$I21oqX4Q$/OqAKTdBbe8xreKwZtWR3.  role Network-Monitor
domain SH-Gold-Domain
username vrtadmgold password 5 $1$.gNJPJS6$UtozYODAuirfw8XHR1FA8/  role Admin domain
SH-Gold-Domain
username vrtnetmongold password 5 $1$InjySHhu$oLsQV267Nu68q3fH6h7Z4.  role Network-Monitor
domain SH-Gold-Domain
username vrtjohndoe password 5 $1$klNcwN8c$3hTNwFSMtned/9k2a5RPg.  role Admin domain
SH-Gold-Domain

snmp-server community ACE-public group Network-Monitor
snmp-server community ACE-private group Network-Monitor

snmp-server host 10.1.0.236 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-2_ACE2-1/SH-Bridge# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:24 ---

+++ 09:57:24 6K-2_ACE2-1 ctxExec +++
changeto SH-Bridge


6K-2_ACE2-1/SH-Bridge#


6K-2_ACE2-1/SH-Bridge# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:24 ---
```

```
+++ 09:57:24 6K-2_ACE2-1 ctxExec +++
changeto A2



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/A2#


6K-2_ACE2-1/A2# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:25 ---

+++ 09:57:25 6K-2_ACE2-1 ctxExec +++
changeto A2



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/A2# show running-config

Generating configuration....


logging enable
logging standby
logging console 6
logging timestamp
logging trap 5
logging history 5
logging buffered 6
logging monitor 5
logging queue 200
logging device-id context-name
logging host 10.86.83.236 udp/514
logging host 10.86.83.39 udp/514
logging message 302022 level 7
logging message 302023 level 7
logging message 302024 level 7
logging message 302025 level 7
logging message 302026 level 7
logging message 302027 level 7


tacacs-server key 7 "vwjjzamggu"
tacacs-server host 172.29.0.235 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.236 key 7 "vwjjzamggu"
tacacs-server host 172.29.0.237 key 7 "vwjjzamggu"
aaa group server tacacs+ sh-gold
  server 172.29.0.236
  server 172.29.0.237

arp 192.168.120.81 00.00.11.11.22.33
arp 172.29.0.201 00.00.22.22.44.55
arp learned-interval 60


crypto authgroup CAUTH_GRP
  cert init.pem
  cert bigcert.pem
  cert SH-ACE-CA.cer
crypto authgroup CAUTH:444_GRP
  cert init.pem
  cert bigcert.pem
  cert SH-ACE-CA.cer
  cert CA-57.cer
crypto authgroup CAUTH_CRP
  cert CA-57.cer
crypto crl SH_ACE_CRL http://sh-ace-ca.cisco.com/CertEnroll/SH-ACE-CA.crl
crypto crl SH_ACE_CRL4 http://sh-ace-ca.cisco.com/CertEnroll/NO-FILE.crl
crypto crl SH_ACE_CRL2 http://10.86.83.127/CertEnroll/SH-ACE-CA-expired.crl
crypto crl SH_ACE_CRL3 http://sh-ace-ca/CertEnroll/CA-57.crl
aaa authentication login error-enable

object-group service OBJ_SERV_1-100
  tcp
  icmp
  igmp
  3
  ip-in-ip
  5
  7
  8
  9
  10
  11
  12
  13
  14
  15
  16
  18
  19
  20
  21
  22
  23
  24
  25
  26
  27
  28
  29
  30
  31
  32
  33
  34
  35
  36
  37
  38
  39
  40
  41
  42
  43
  44
```

```
    45
    46
    gre
    48
    49
    esp
    ah
    52
    53
    54
    55
    56
    57
    58
    59
    60
    61
    62
    63
    64
    65
    66
    67
    68
    69
    70
    71
    72
    73
    74
    75
    76
    77
    78
    79
    80
    81
    82
    83
    84
    85
    86
    87
    eigrp
    ospf
    90
    91
    92
    93
    94
    95
    96
    97
    98
    99
    100
    udp
  object-group network OBJ_NET_10
    host 10.1.0.234
    host 10.10.0.2
    host 10.10.0.3
    host 10.10.0.4
    host 10.10.0.5
    host 10.10.0.6
    host 10.10.0.7
    host 10.10.0.8
    host 10.10.0.9
    host 10.10.0.10
  object-group network OBJ_NET_50
    host 10.10.0.51
    host 10.10.0.52
    host 10.10.0.53
    host 10.10.0.54
    host 10.10.0.55
    host 10.10.0.56
    host 10.10.0.57
    host 10.10.0.58
    host 10.10.0.59
    host 10.10.0.60
    host 10.10.0.61
    host 10.10.0.62
    host 10.10.0.63
    host 10.10.0.64
    host 10.10.0.65
    host 10.10.0.66
    host 10.10.0.67
    host 10.10.0.68
    host 10.10.0.69
    host 10.10.0.70
    host 10.10.0.71
    host 10.10.0.72
    host 10.10.0.73
    host 10.10.0.74
    host 10.10.0.75
    host 10.10.0.76
    host 10.10.0.77
    host 10.10.0.78
    host 10.10.0.79
    host 10.10.0.80
    host 10.10.0.81
    host 10.10.0.82
    host 10.10.0.83
    host 10.10.0.84
    host 10.10.0.85
    host 10.10.0.86
    host 10.10.0.87
    host 10.10.0.88
    host 10.10.0.89
    host 10.10.0.90
    host 10.10.0.91
    host 10.10.0.92
    host 10.10.0.93
    host 10.10.0.94
    host 10.10.0.95
    host 10.10.0.96
    host 10.10.0.97
    host 10.10.0.98
    host 10.10.0.99
    host 192.168.140.107

  access-list NAT_ACCESS line 8 extended permit ip any any
  access-list anyone-ip line 10 extended permit ip any any

  script file 1 FTP_PROBE_SCRIPT
  script file 2 TFTP_PROBE
  script file 3 LDAP_PROBE
```

```
probe http FORCED-FAIL
  interval 10
  faildetect 2
  passdetect interval 5
  receive 5
  request method get url /notthere.html
  expect status 200 299
  open 3
probe ftp FTP
  interval 10
  faildetect 2
  passdetect interval 10
  receive 5
  expect status 220 220
  open 3
probe http GEN_HTTP
  interval 10
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "A2_ctx_ACE_CLEAR"
probe https GEN_HTTPS
  interval 10
  passdetect interval 2
  passdetect count 2
  expect status 200 200
  header Via header-value "A2_ctx_ACE_SSL"
probe icmp HA-ICMP
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:1755
  port 1755
  interval 2
  faildetect 2
  passdetect interval 2
probe tcp HA-TCP:554
  port 554
  interval 2
  faildetect 2
  passdetect interval 2
probe http HTTP
  interval 10
  passdetect interval 10
  receive 5
  expect status 200 200
  open 3
probe icmp ICMP
  interval 10
  faildetect 2
  passdetect interval 10
probe https SSL
  interval 10
  passdetect interval 7
  passdetect count 2
  expect status 200 200
  header Via header-value "A2_ctx_SSL_Prb"
probe tcp TCP
  interval 5
  faildetect 2
  passdetect interval 10
  open 3
probe tcp TCP:443
  port 443
  interval 5
  passdetect interval 10
  connection term forced
  open 3
probe udp UDP:2222
  port 2222
  interval 5
  passdetect interval 2
probe udp UDP:53
  interval 5
  passdetect interval 2
probe icmp WAE_ICMP
  interval 2
  faildetect 1
  passdetect interval 2
  passdetect count 1
probe snmp linuxCpu
  interval 5
  passdetect interval 2
  passdetect count 2
  community ace-public
  oid 1.3.6.1.4.1.2021.10.1.3.1
    weight 8000
  oid 1.3.6.1.4.1.2021.10.1.3.2
    weight 8000
probe snmp windowsCpu
  interval 5
  passdetect interval 2
  passdetect count 2
  community ace-public
  oid .1.3.6.1.2.1.25.3.3.1.2.1

ip domain-lookup
ip domain-list cisco.com
ip name-server 172.28.0.152
ip name-server 161.44.124.122
ip name-server 10.86.83.121

parameter-map type ssl CLIENT_SSL
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 0
  close-protocol disabled
parameter-map type http HTTP_PARAM
  case-insensitive
  persistence-rebalance
parameter-map type http HTTP_REBAL_REUSE
  server-conn reuse
  case-insensitive
  persistence-rebalance
parameter-map type ssl INIT_SSL
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 3660
parameter-map type connection NORMALIZE_MY_TCP_TRAFFIC
  nagle
  slowstart
  set timeout inactivity 30
  tcp-options timestamp allow
  syn-data drop
  exceed-mss allow
  urgent-flag clear
parameter-map type ssl PARM_ACE_AS_CLIENT
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  version TLS1
parameter-map type ssl PARM_ACE_AS_CLIENT_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_AS_CLIENT_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_AS_CLIENT_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM
  cipher RSA_WITH_RC4_128_MD5 priority 6
  version TLS1
parameter-map type ssl PARM_ACE_TERM_EXPORT_CIPHERS
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5 priority 3
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA priority 2
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA priority 5
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA priority 4
parameter-map type ssl PARM_ACE_TERM_STRONG_CIPHERS
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA priority 2
  cipher RSA_WITH_AES_256_CBC_SHA priority 3
parameter-map type ssl PARM_ACE_TERM_WEAK_CIPHERS
  cipher RSA_WITH_RC4_128_MD5
  cipher RSA_WITH_RC4_128_SHA priority 2
  cipher RSA_WITH_DES_CBC_SHA priority 3
  version TLS1
  close-protocol disabled
parameter-map type connection RL_BW
  rate-limit bandwidth 800000
```

```
parameter-map type ssl SSL-INIT
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 3660
parameter-map type generic SSL-PARSE
  set max-parse-length 60
parameter-map type ssl SSL_CAUTH
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 3660
parameter-map type connection TCP_PARAM
  syn-data drop
  exceed-mss allow
parameter-map type http TEMP_HTTP_REBAL_REUSE
  server-conn reuse
  case-insensitive
  persistence-rebalance
parameter-map type connection TEMP_TCP_PARAM
  syn-data drop
  exceed-mss allow
parameter-map type ssl TERM_SSL
  cipher RSA_WITH_RC4_128_MD5 priority 5
  cipher RSA_WITH_RC4_128_SHA
  cipher RSA_WITH_DES_CBC_SHA
  cipher RSA_WITH_3DES_EDE_CBC_SHA
  cipher RSA_WITH_AES_128_CBC_SHA
  cipher RSA_WITH_AES_256_CBC_SHA priority 10
  cipher RSA_EXPORT_WITH_RC4_40_MD5
  cipher RSA_EXPORT1024_WITH_RC4_56_MD5
  cipher RSA_EXPORT_WITH_DES40_CBC_SHA
  cipher RSA_EXPORT1024_WITH_DES_CBC_SHA
  cipher RSA_EXPORT1024_WITH_RC4_56_SHA
  session-cache timeout 3660

action-list type modify http HTTP-HTTPS-Rewrite-Term
  header insert response Cache-Control header-value "max-age=901"
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header insert request Pragma header-value "no-cache, no-cache is the only pragma value I 
have ever seen"
  header delete response ETag
  header delete response Set-Cookie
  header rewrite response Keep-Alive header-value "timeout=150" replace "timeout=300"
  header rewrite request User-Agent header-value ".*petescape.*" replace "Mozilla/4.0 
(compatible; Tarantula Client; Runs on Linux)"
  ssl url rewrite location "www.ssl-term-ace.com"
  ssl url rewrite location "192.168.140.103"
action-list type modify http HTTP-HTTPS-Rewrite-E2E
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header insert response Cache-Control header-value "max-age=2592008"
  header insert request Pragma header-value "cache, no-cache is the only pragma value I
have ever seen but on this one we will use cache"
  header delete response ETag
  header delete response Set-Cookie
  header rewrite request User-Agent header-value ".*petescape.*" replace "Mozilla/5.0
(compatible; Tarantula Client; Does not run on a MAC)"
  ssl url rewrite location "www.ssl-end-to-end-ace.com" sslport 444
  ssl url rewrite location "192.168.140.105" sslport 444
action-list type modify http HTTP-HTTPS-Rewrite-Init
  header insert response Cache-Control header-value "max-age=2008"
  header insert response SourceIP header-value "%is"
  header insert response DestIP header-value "%id"
  header delete response ETag
  header delete response Set-Cookie
  header rewrite response Keep-Alive header-value "timeout=150" replace "timeout=300"
  header rewrite request User-Agent header-value ".*petescape.*" replace "Mozilla/4.0
(compatible; Tarantula Client; Runs on Linux)"

rserver host BRG-IIS-1
  ip address 172.28.1.26
  inservice
rserver host BRG-IIS-2
  ip address 172.28.1.27
  inservice
rserver host BRG-IIS-3
  ip address 172.28.1.28
  inservice
rserver host BRG-IIS-4
  ip address 172.28.1.29
  inservice
rserver host BRG-IIS-5
  ip address 172.28.1.30
  inservice
rserver host BRG-LINUX-1
  ip address 172.28.1.21
  inservice
rserver host BRG-LINUX-11
  ip address 192.168.120.11
  inservice
rserver host BRG-LINUX-12
  ip address 192.168.120.12
  inservice
rserver host BRG-LINUX-13
  ip address 192.168.120.13
  inservice
rserver host BRG-LINUX-14
  ip address 192.168.120.14
  inservice
rserver host BRG-LINUX-15
  ip address 192.168.120.15
  inservice
rserver host BRG-LINUX-2
  ip address 172.28.1.22
  inservice
rserver host BRG-LINUX-3
  ip address 172.28.1.23
  inservice
rserver host BRG-LINUX-4
  ip address 172.28.1.24
  inservice
rserver host BRG-LINUX-5
  ip address 172.28.1.25
  inservice
rserver host LOCAL-239-FRAGRTR
  ip address 172.29.0.239
  inservice
rserver host LOCAL-IIS-241
  ip address 172.29.0.241
  inservice
rserver host LOCAL-IIS-243
  ip address 172.29.0.243
  inservice
rserver host LOCAL-IIS-245
  ip address 172.29.0.245
  inservice
rserver host LOCAL-LINUX-240
  ip address 172.29.0.240
  inservice
rserver host LOCAL-LINUX-242
  ip address 172.29.0.242
  inservice
rserver host LOCAL-LINUX-244
  ip address 172.29.0.244
  inservice
rserver host RT-IIS-152
  ip address 172.28.0.152
  inservice
rserver host RT-LINUX-151
  ip address 172.28.0.151
  inservice
rserver host RT-LINUX-153
  ip address 172.28.0.153
  inservice
rserver host RT-LINUX-154
  ip address 172.28.0.154
  inservice
rserver host WAE_1
  description WAE 1
  ip address 172.16.0.30
rserver host WAE_2
  description WAE 2
  ip address 172.16.0.31
  inservice

ssl-proxy service ACE_TERM
  ssl advanced-options PARM_ACE_TERM_EXPORT_CIPHERS
ssl-proxy service CAAUTH
ssl-proxy service CAUTH
  key term-wc.key
  cert term-wc.cer
  authgroup CAUTH_GRP
  crl SH_ACE_CRL4
  ssl advanced-options SSL_CAUTH
ssl-proxy service CAUTH:444
  key term-wc.key
  cert term-wc.cer
  authgroup CAUTH:444_GRP
  crl best-effort
  ssl advanced-options SSL_CAUTH
ssl-proxy service CAUTHL4
  authgroup CAUTH_CRP
  crl SH_ACE_CRL4
ssl-proxy service E2E_SSL
```

```
  key pkey.pem
  cert end-to-end.pem
  ssl advanced-options TERM_SSL
ssl-proxy service INIT_E2E_SSL
  ssl advanced-options CLIENT_SSL
ssl-proxy service INIT_SSL
  ssl advanced-options INIT_SSL
ssl-proxy service RL_E2E_SSL
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options TERM_SSL
ssl-proxy service RL_INIT_E2E_SSL
  ssl advanced-options CLIENT_SSL
ssl-proxy service SSL-INIT
  ssl advanced-options SSL-INIT
ssl-proxy service TERM_SSL
  key term-wc.key
  cert term-wc.cer
  ssl advanced-options PARM_ACE_TERM

serverfarm host E2E
  failaction purge
  predictor leastconns
  probe SSL
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-IIS-241 443
    inservice
  rserver LOCAL-IIS-245 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-IIS-152 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host E2E_CHUNK
  failaction purge
  predictor leastconns
  probe TCP:443
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host E2E_POST
  failaction purge
  predictor leastconns
  probe TCP:443
  retcode 100 599 check count
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-LINUX-151 443
```

```
    inservice
serverfarm host FTP
  failaction purge
  predictor leastconns
  probe FTP
  rserver BRG-LINUX-11
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
serverfarm host INIT
  failaction purge
  predictor leastconns
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-IIS-241 443
    inservice
  rserver LOCAL-IIS-245 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-IIS-152 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host INIT2
  failaction purge
  predictor leastconns
  probe TCP
  rserver BRG-IIS-1 443
    inservice
  rserver BRG-LINUX-1 443
    inservice
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-IIS-241 443
    inservice
  rserver LOCAL-IIS-245 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
  rserver RT-IIS-152 443
    inservice
  rserver RT-LINUX-151 443
    inservice
serverfarm host L4
  failaction purge
  probe HTTP
  probe SSL
  rserver BRG-LINUX-11
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
serverfarm host L7
  failaction purge
  predictor leastconns
  probe TCP
  retcode 100 599 check count
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host LEASTLOAD
  predictor least-loaded probe linuxCpu
    weight connection
    autoadjust average
  rserver LOCAL-LINUX-240
    inservice
  rserver LOCAL-LINUX-242
    inservice
serverfarm host LEASTLOAD-2
  predictor least-loaded probe windowsCpu
    weight connection
    autoadjust average
  rserver LOCAL-IIS-243
    inservice
  rserver LOCAL-IIS-245
    inservice
serverfarm host ORIGIN-SF-WAAS
  description SERVER SF FOR WAAS RETURN TRAFFIC
  predictor leastconns
  probe TCP
  rserver BRG-IIS-1
    conn-limit max 6500 min 5000
    inservice
  rserver BRG-LINUX-1
    conn-limit max 6500 min 5000
    inservice
serverfarm host PREDICTOR
  failaction purge
  predictor least-bandwidth samples 2 assess-time 1
  probe HTTP
  rserver BRG-IIS-1
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-11
```

```
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-IIS-245
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
serverfarm host RL_SSL
  failaction purge
  probe SSL
  rserver BRG-LINUX-11 443
    inservice
  rserver LOCAL-LINUX-240 443
    inservice
serverfarm host RL_WWW
  failaction purge
  probe HTTP
  retcode 100 599 check count
  rserver BRG-LINUX-11
    inservice
  rserver LOCAL-IIS-241
    inservice
serverfarm host SSL-SESSION
  failaction purge
  predictor leastconns
  probe TCP:443
  rserver BRG-IIS-1
    inservice
  rserver BRG-IIS-2
    inservice
  rserver BRG-LINUX-1
    inservice
  rserver BRG-LINUX-11
    inservice
  rserver BRG-LINUX-2
    inservice
  rserver LOCAL-IIS-241
    inservice
  rserver LOCAL-IIS-245
    inservice
  rserver LOCAL-LINUX-240
    inservice
  rserver RT-IIS-152
    inservice
  rserver RT-LINUX-151
    inservice
serverfarm host SSL_CAUTH
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host SSL_CAUTH2
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host TERM
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 81
    inservice
  rserver BRG-LINUX-11 81
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 81
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 81
    inservice
serverfarm host TERM2
  failaction purge
  predictor leastconns
  probe GEN_HTTP
  rserver BRG-IIS-1 80
    inservice
  rserver BRG-LINUX-1 80
    inservice
  rserver BRG-LINUX-11 80
    inservice
  rserver LOCAL-IIS-241 80
    inservice
  rserver LOCAL-IIS-245 80
    inservice
  rserver LOCAL-LINUX-240 80
    inservice
  rserver RT-IIS-152 80
    inservice
  rserver RT-LINUX-151 80
    inservice
serverfarm host WAAS
  description WAAS SF TRANSPARENT MODE
  transparent
  predictor leastconns
  probe WAE_ICMP
  rserver WAE_1
    conn-limit max 6500 min 5000
    inservice
  rserver WAE_2
    conn-limit max 6500 min 5000
    inservice

sticky http-cookie TERM_SSL STKY-CKY_TERM
  cookie insert
  timeout 30
  replicate sticky
  serverfarm TERM
sticky http-cookie INIT_SSL STKY-CKY_INIT
  cookie insert
  timeout 30
  replicate sticky
  serverfarm INIT
sticky http-cookie E2E_SSL STKY-CKY_E2E_ANY
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E
sticky http-cookie E2E_SSL STKY-CKY_E2E_POST
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E_POST
sticky http-cookie E2E_SSL STKY-CKY_E2E_CHUNK
  cookie insert
  timeout 30
  replicate sticky
  serverfarm E2E_CHUNK
sticky layer4-payload SSL-SESSION_STKY
  timeout 600
  replicate sticky
  serverfarm SSL-SESSION
  response sticky
  layer4-payload offset 43 length 32 begin-pattern "\x20"
sticky http-cookie SSL_CAUTH STKY-CKY_CAUTH
  cookie insert
  timeout 30
  replicate sticky
  serverfarm SSL_CAUTH
sticky ip-netmask 255.255.255.255 address source STKY-SRCIP-GRP30
  timeout 30
  replicate sticky
  serverfarm RL_SSL
sticky http-cookie RL_E2E_SSL RL-COOKIE
  cookie insert
  timeout 30
  replicate sticky
  serverfarm RL_SSL

class-map type http inspect match-all CHUNK-HDR_INSPECT
  2 match header Transfer-Encoding header-value "chunked"
class-map type http inspect match-all CHUNK-XFR_INSPECT
  2 match transfer-encoding chunked
class-map type http loadbalance match-all CHUNK.PL_URL
  2 match http url .*.chunk.pl
class-map type http loadbalance match-all CHUNK_HDR
  2 match http header Transfer-Encoding header-value "chunked"
```

```
class-map type http loadbalance match-any CHUNK_POST.PL_URL
  2 match http url .*.chunk.pl
  3 match http url .*cgipostform.pl
class-map match-all E2E-VIP_105:443
  2 match virtual-address 192.168.140.105 tcp eq https
class-map match-all E2E-VIP_105:444
  2 match virtual-address 192.168.140.105 tcp eq 444
class-map match-all FTP-VIP_107
  2 match virtual-address 192.168.140.107 tcp eq ftp
class-map match-all HEADER-INSERT-VIP_121:80
  2 match virtual-address 192.168.120.121 tcp eq www
class-map match-all INIT-VIP_104:80
  2 match virtual-address 192.168.140.104 tcp eq www
class-map type http inspect match-any INSPECT_HTTP_GOOD
  2 match request-method rfc connect
  3 match request-method rfc delete
  5 match request-method rfc head
  6 match request-method rfc options
  8 match request-method rfc put
  9 match request-method rfc trace
  10 match url .*
  11 match request-method ext copy
  12 match request-method ext edit
  13 match request-method ext getattr
  14 match request-method ext getattrname
  15 match request-method ext getprops
  16 match request-method ext index
  17 match request-method ext lock
  18 match request-method ext mkdir
  19 match request-method ext move
  20 match request-method ext revadd
  21 match request-method ext revlabel
  22 match request-method ext revlog
  23 match request-method ext revnum
  24 match request-method ext save
  25 match request-method ext setattr
  26 match request-method ext startrev
  27 match request-method ext stoprev
  28 match request-method ext unedit
  29 match request-method ext unlock
  30 match request-method rfc post
  31 match request-method rfc get
class-map match-all L4-VIP_107:NAT
  2 match destination-address 192.168.140.107 255.255.255.255
class-map match-all L4-VIP_107:TCP_80-443
  2 match virtual-address 192.168.140.107 tcp range 80 443
class-map match-all L4-VIP_107:UDP_ANY
  2 match virtual-address 192.168.140.107 udp any
class-map match-all L7-VIP_110:80
  2 match virtual-address 192.168.140.110 tcp eq www
class-map match-all LL_VIP_111:80
  2 match virtual-address 192.168.140.111 tcp eq www
class-map match-all LL_VIP_112:80
  2 match virtual-address 192.168.140.112 tcp eq www
class-map type management match-any MGT
  description remote-access-traffic-match
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol snmp any
  7 match protocol https any
class-map type http loadbalance match-any POST.PL_URL
  2 match http url .*cgipostform.pl
```

```
class-map type http inspect match-all POST_INSPECT
  2 match request-method rfc post
class-map match-all PRED-VIP_108:80
  2 match virtual-address 192.168.140.108 tcp eq www
class-map type management match-any REMOTE
  2 match protocol telnet any
  3 match protocol ssh any
  4 match protocol icmp any
  5 match protocol http any
  6 match protocol https any
class-map match-all RL_E2E-VIP_109:443
  2 match virtual-address 192.168.140.109 tcp eq https
class-map match-all RL_WWW-VIP_109
  2 match virtual-address 192.168.140.109 tcp eq www
class-map match-any SSL-SESSION-VIP_106:443
  2 match virtual-address 192.168.140.106 tcp eq https
class-map type generic match-any SSLID-32_REGEX
  2 match layer4-payload regex "\x16\x03[\x00\x01]..[\x01\x02].*"
  3 match layer4-payload regex "\x80\x4c.*"
class-map match-all SSL_CAUTH-VIP_102:443
  2 match virtual-address 192.168.140.102 tcp eq https
class-map match-all SSL_CAUTH-VIP_102:444
  2 match virtual-address 192.168.140.102 tcp eq 444
class-map match-all TERM-VIP_103:443
  2 match virtual-address 192.168.140.103 tcp eq https
class-map type http loadbalance match-all URL*_L7
  2 match http url .*
class-map type http loadbalance match-all URL_*.GIF
  2 match http url .*.gif
class-map type http loadbalance match-all URL_*.JPG
  2 match http url .*.jpg
class-map match-any WAAS-TO-ACE-VIP_100:ANY
  2 match virtual-address 192.168.140.100 tcp any
class-map match-any WAAS-VIP_100:80
  2 match virtual-address 192.168.140.100 tcp eq www

policy-map type management first-match P-MGT
  class MGT
    permit

policy-map type loadbalance first-match L3
  class class-default
    serverfarm SSL-SESSION
policy-map type loadbalance first-match PLBSF_FTP
  class class-default
    serverfarm FTP
policy-map type loadbalance first-match PLBSF_INIT
  class URL_*.GIF
    serverfarm INIT2
    insert-http I_AM header-value "SSL_INIT.GIF"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client INIT_SSL
  class URL_*.JPG
    serverfarm INIT2
    insert-http I_AM header-value "SSL_INIT.JPG"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client INIT_SSL
  class URL*_L7
    sticky-serverfarm STKY-CKY_INIT
    action HTTP-HTTPS-Rewrite-Init
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_INIT_COOKIE_INS"
    ssl-proxy client INIT_SSL
policy-map type loadbalance first-match PLBSF_L4
  class class-default
    serverfarm L4
policy-map type loadbalance first-match PLBSF_L7
  class URL*_L7
    serverfarm L7
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SF_L7"
policy-map type loadbalance first-match PLBSF_LL
  class class-default
    serverfarm LEASTLOAD
policy-map type loadbalance first-match PLBSF_LL-2
  class class-default
    serverfarm LEASTLOAD-2
policy-map type loadbalance first-match PLBSF_LL_E2E
  class POST.PL_URL
    sticky-serverfarm STKY-CKY_E2E_POST
    insert-http I_AM header-value "E2E_SSL_POST"
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    ssl-proxy client INIT_E2E_SSL
  class CHUNK.PL_URL
    sticky-serverfarm STKY-CKY_E2E_CHUNK
    insert-http I_AM header-value "E2E_SSL_CHUNK_XFER"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client INIT_E2E_SSL
  class class-default
    sticky-serverfarm STKY-CKY_E2E_ANY
    action HTTP-HTTPS-Rewrite-E2E
    insert-http I_AM header-value "E2E_SSL_ANY"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client INIT_E2E_SSL
policy-map type loadbalance first-match PLBSF_ORIGIN-SF-WAAS
  class class-default
    serverfarm ORIGIN-SF-WAAS
policy-map type loadbalance first-match PLBSF_PREDICTOR
  class URL*_L7
    serverfarm PREDICTOR
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SF_PREDICTOR"
policy-map type loadbalance first-match PLBSF_RL_SSL
  class URL*_L7
    sticky-serverfarm RL-COOKIE
    insert-http I_AM header-value "Rate_Limit_E2E_SSL"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
    ssl-proxy client RL_INIT_E2E_SSL
policy-map type loadbalance first-match PLBSF_RL_WWW
  class class-default
    serverfarm RL_WWW
policy-map type loadbalance first-match PLBSF_SSL_CAUTH
  class URL_*.GIF
    serverfarm SSL_CAUTH2
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_CAUTH.GIF"
  class URL_*.JPG
    serverfarm SSL_CAUTH2
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_CAUTH.JPG"
  class URL*_L7
    sticky-serverfarm STKY-CKY_CAUTH
    insert-http I_AM header-value "SSL_CAUTH_COOKIE_INS"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
policy-map type loadbalance first-match PLBSF_TERM
  class URL_*.GIF
    serverfarm TERM2
    action HTTP-HTTPS-Rewrite-Term
    insert-http I_AM header-value "SSL_TERM.GIF"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
  class URL_*.JPG
    serverfarm TERM2
    action HTTP-HTTPS-Rewrite-Term
    insert-http I_AM header-value "SSL_TERM.JPG"
    insert-http SRC_Port header-value "%ps"
    insert-http SRC_IP header-value "%is"
  class URL*_L7
    sticky-serverfarm STKY-CKY_TERM
    action HTTP-HTTPS-Rewrite-Term
    insert-http SRC_IP header-value "%is"
    insert-http SRC_Port header-value "%ps"
    insert-http I_AM header-value "SSL_TERM_COOKIE_INS"
policy-map type loadbalance first-match PLBSF_WAE
  class class-default
    serverfarm WAAS backup ORIGIN-SF-WAAS

policy-map type loadbalance generic first-match PLBSF_SSL-SESSION
  class SSLID-32_REGEX
    sticky-serverfarm SSL-SESSION_STKY

policy-map type inspect http all-match INSPECT_GOOD_HTTP
  class INSPECT_HTTP_GOOD
    permit

policy-map multi-match A2-VIPS
  class E2E-VIP_105:443
    loadbalance vip inservice
    loadbalance policy PLBSF_LL_E2E
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_REBAL_REUSE
    ssl-proxy server E2E_SSL
    connection advanced-options TCP_PARAM
  class SSL_CAUTH-VIP_102:443
    loadbalance vip inservice
    loadbalance policy PLBSF_SSL_CAUTH
    loadbalance vip icmp-reply active
    loadbalance vip advertise active
    nat dynamic 1 vlan 29
    nat dynamic 1 vlan 106
    nat dynamic 1 vlan 120
    inspect http policy INSPECT_GOOD_HTTP
    appl-parameter http advanced-options HTTP_PARAM
    ssl-proxy server CAUTH
    connection advanced-options TCP_PARAM
```

```
class L4-VIP_107:NAT
  nat dynamic 1 vlan 29
  nat dynamic 1 vlan 106
  nat dynamic 1 vlan 120
class L4-VIP_107:TCP_80-443
  loadbalance vip inservice
  loadbalance policy PLBSF_L4
  loadbalance vip icmp-reply active
  loadbalance vip advertise active
class PRED-VIP_108:80
  loadbalance vip inservice
  loadbalance policy PLBSF_PREDICTOR
  loadbalance vip icmp-reply active
  loadbalance vip advertise active
  nat dynamic 1 vlan 29
  nat dynamic 1 vlan 106
  nat dynamic 1 vlan 120
  appl-parameter http advanced-options HTTP_REBAL_REUSE
class RL_E2E-VIP_109:443
  loadbalance vip inservice
  loadbalance policy PLBSF_RL_SSL
  loadbalance vip icmp-reply active
  loadbalance vip advertise active
  nat dynamic 1 vlan 29
  nat dynamic 1 vlan 106
  nat dynamic 1 vlan 120
  inspect http policy INSPECT_GOOD_HTTP
  appl-parameter http advanced-options HTTP_REBAL_REUSE
  ssl-proxy server RL_E2E_SSL
  connection advanced-options TCP_PARAM
class TERM-VIP_103:443
  loadbalance vip inservice
  loadbalance policy PLBSF_TERM
  loadbalance vip icmp-reply active
  loadbalance vip advertise active
  nat dynamic 1 vlan 29
  nat dynamic 1 vlan 106
  nat dynamic 1 vlan 120
  inspect http policy INSPECT_GOOD_HTTP
  appl-parameter http advanced-options HTTP_PARAM
  ssl-proxy server TERM_SSL
  connection advanced-options TCP_PARAM
class INIT-VIP_104:80
  loadbalance vip inservice
  loadbalance policy PLBSF_INIT
  loadbalance vip icmp-reply active
  loadbalance vip advertise active
  nat dynamic 1 vlan 29
  nat dynamic 1 vlan 106
  nat dynamic 1 vlan 120
  appl-parameter http advanced-options HTTP_PARAM
class L7-VIP_110:80
  loadbalance vip inservice
  loadbalance policy PLBSF_L7
  loadbalance vip icmp-reply active
  loadbalance vip advertise active
  nat dynamic 1 vlan 29
  nat dynamic 1 vlan 106
  nat dynamic 1 vlan 120
class SSL-SESSION-VIP_106:443
  loadbalance vip inservice
  loadbalance policy PLBSF_SSL-SESSION
  loadbalance vip icmp-reply active
  loadbalance vip advertise active
```

```
      nat dynamic 1 vlan 29
      nat dynamic 1 vlan 106
      nat dynamic 1 vlan 120
      appl-parameter generic advanced-options SSL-PARSE
    class SSL_CAUTH-VIP_102:444
      loadbalance vip inservice
      loadbalance policy PLBSF_SSL_CAUTH
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 29
      nat dynamic 1 vlan 106
      nat dynamic 1 vlan 120
      inspect http policy INSPECT_GOOD_HTTP
      appl-parameter http advanced-options HTTP_PARAM
      ssl-proxy server CAUTH:444
      connection advanced-options TCP_PARAM
    class FTP-VIP_107
      loadbalance vip inservice
      loadbalance policy PLBSF_FTP
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      inspect ftp
    class E2E-VIP_105:444
      loadbalance vip inservice
      loadbalance policy PLBSF_LL_E2E
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 29
      nat dynamic 1 vlan 106
      nat dynamic 1 vlan 120
      inspect http policy INSPECT_GOOD_HTTP
      appl-parameter http advanced-options HTTP_REBAL_REUSE
      ssl-proxy server E2E_SSL
      connection advanced-options TCP_PARAM
    class L4-VIP_107:UDP_ANY
      loadbalance vip inservice
      loadbalance policy PLBSF_L4
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
    class LL_VIP_111:80
      loadbalance vip inservice
      loadbalance policy PLBSF_LL
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 29
      nat dynamic 1 vlan 106
      nat dynamic 1 vlan 120
    class LL_VIP_112:80
      loadbalance vip inservice
      loadbalance policy PLBSF_LL-2
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 29
      nat dynamic 1 vlan 106
      nat dynamic 1 vlan 120
    class RL_WWW-VIP_109
      loadbalance vip inservice
      loadbalance policy PLBSF_RL_WWW
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 29
      nat dynamic 1 vlan 106
      nat dynamic 1 vlan 120
  policy-map multi-match CLIENTS_TO_WAAS
    class WAAS-VIP_100:80
      loadbalance vip inservice
      loadbalance policy PLBSF_WAE
      loadbalance vip icmp-reply active
      loadbalance vip advertise active
      nat dynamic 1 vlan 29
  policy-map multi-match WAAS-TO-ACE
    class WAAS-TO-ACE-VIP_100:ANY
      loadbalance vip inservice
      loadbalance policy PLBSF_ORIGIN-SF-WAAS
      loadbalance vip icmp-reply active
      nat dynamic 1 vlan 29

  service-policy input P-MGT

  interface vlan 29
    description SERVER VLAN-29
    ip address 172.29.0.6 255.255.255.0
    alias 172.29.0.4 255.255.255.0
    peer ip address 172.29.0.5 255.255.255.0
    fragment chain 20
    fragment min-mtu 68
    access-group input anyone-ip
    nat-pool 1 192.168.106.75 192.168.106.75 netmask 255.255.255.0 pat
    no shutdown
  interface vlan 83
    ip address 10.86.83.212 255.255.255.0
    peer ip address 10.86.83.209 255.255.255.0
    access-group input anyone-ip
    no shutdown
  interface vlan 99
    ip address 192.168.99.8 255.255.255.0
    peer ip address 192.168.99.9 255.255.255.0
    access-group input anyone-ip
    no shutdown
  interface vlan 106
    description CLIENT VLAN-106
    ip address 192.168.106.6 255.255.255.0
    alias 192.168.106.4 255.255.255.0
    peer ip address 192.168.106.5 255.255.255.0
    fragment chain 20
    fragment min-mtu 68
    no icmp-guard
    access-group input anyone-ip
    nat-pool 1 192.168.106.70 192.168.106.70 netmask 255.255.255.0 pat
    service-policy input CLIENTS_TO_WAAS
    service-policy input A2-VIPS
    no shutdown
  interface vlan 120
    description CLIENT VLAN-120
    ip address 192.168.120.6 255.255.255.0
    alias 192.168.120.4 255.255.255.0
    peer ip address 192.168.120.5 255.255.255.0
    fragment chain 20
    fragment min-mtu 68
    access-group input anyone-ip
    nat-pool 1 192.168.120.80 192.168.120.80 netmask 255.255.255.0 pat
    no shutdown
  interface vlan 160
    description WAE VLAN-160
    ip address 172.16.0.3 255.255.255.0
    alias 172.16.0.1 255.255.255.0
    peer ip address 172.16.0.2 255.255.255.0
    no normalization
    fragment chain 20
    fragment min-mtu 68
    no icmp-guard
    access-group input anyone-ip
    nat-pool 1 192.168.106.85 192.168.106.85 netmask 255.255.255.0 pat
    service-policy input WAAS-TO-ACE
    no shutdown

  ft track host GW_251-252
    track-host 192.168.16.252
    peer track-host 192.168.16.251
    peer probe HA-ICMP priority 10
    probe HA-ICMP priority 5
    priority 5
    peer priority 110
  ft track hsrp HSRP_120
    track-hsrp hsrp-Vl120-120
    peer track-hsrp hsrp-Vl120-120
    priority 5
    peer priority 110
  ft track interface Int_4/37_6/13_V99
    track-interface vlan 99
    peer track-interface vlan 99
    peer priority 110
  ft track host RT-241
    peer track-host 172.29.0.241
    peer probe HA-TCP:1755 priority 60
    peer probe HA-TCP:554 priority 50

  domain A2
    add-object all

  role SHAdmin
    rule 1 permit create
    rule 2 permit monitor
    rule 3 permit modify
  role SHUser
    rule 1 deny create
    rule 2 permit monitor
    rule 3 deny modify
    rule 4 permit debug

  ip route 172.28.0.0 255.254.0.0 172.29.0.253
  ip route 192.168.16.251 255.255.255.255 192.168.106.251
  ip route 192.168.16.252 255.255.255.255 192.168.106.252
  ip route 10.1.0.0 255.255.255.0 192.168.106.254
  ip route 10.3.0.0 255.255.255.0 192.168.106.254
  ip route 0.0.0.0 0.0.0.0 192.168.106.254
  username admin password 5 $1$hU1iScF8$WmpdK4IcQI2ofTMDm6l.N1  role Admin domain
  default-domain
  username localadmin password 5 $1$g5rd5HO2$C34zVe3a9f73Dce/WNvbM.  role Admin domain
  default-domain
  username localuser password 5 $1$I21oqX4Q$/OqAKTdBbe8xreKwZtWR3.  role Network-Monitor
  domain default-domain
  username vrtadma2 password 5 $1$.gNJPJS6$UtozYODAuirfw8XHR1FA8/  role Admin domain
  default-domain
  username vrtnetmongold password 5 $1$InjySHhu$oLsQV267Nu68q3fH6h7Z4.  role Network-Monitor
  domain default-domain
  username vrtjohndoe password 5 $1$klNcwN8c$3hTNwFSMtned/9k2a5RPg.  role Admin domain
  default-domain
  username jw password 5 $1$yemV0ad1$4oqyG73BYLG3Q4yGk6Udq.  role Admin domain
  default-domain

  snmp-server community ACE-public group Network-Monitor
```

```
snmp-server community ACE-private group Network-Monitor

snmp-server host 10.1.0.236 traps version 2c ACE-public

snmp-server enable traps slb vserver
snmp-server enable traps slb real
snmp-server enable traps syslog
snmp-server enable traps snmp authentication
snmp-server enable traps snmp linkup
snmp-server enable traps snmp linkdown


6K-2_ACE2-1/A2# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:26 ---

+++ 09:57:26 6K-2_ACE2-1 ctxExec +++
changeto A2



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/A2#


6K-2_ACE2-1/A2# changeto Admin



NOTE: Configuration mode has been disabled on all sessions


6K-2_ACE2-1/Admin#
--- 09:57:26 ---

+++ 09:57:28 6K-2_ACE2-1 ctxExec +++
term width 512


6K-2_ACE2-1/Admin#
--- 09:57:28 ---

+++ 09:57:28 6K-2_ACE2-1 ctxExec +++
term length 0


6K-2_ACE2-1/Admin#
--- 09:57:28 ---

+++ 09:57:28 6K-2_ACE2-1 ctxConfig +++
config terminal


Configuration mode is currently disabled
```

```
6K-2_ACE2-1/Admin#


6K-2_ACE2-1/Admin#
--- 09:58:08 ---

+++ 09:58:08 6K-2_ACE2-1 destroy +++
```

Return to 6K-2 ACE1 All Context, page 416

Go to Basic Topology: Test Device Configurations, page 212

Go to Enterprise Lab Topology, page 3

Go to Test Cases, page 11