Review Secure Endpoint (CSE) Windows Scans

Contents

Introduction

This document describes the different types of scans of a Windows connector.

Prerequisites

The prerequisites for this document are:

- Windows Endpoint
- Secure Endpoint (CSE) version v.8.0.1.21164 or later
- Access to Secure Endpoint Console

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on these software and hardware versions:

- Secure Endpoint Console
- Windows 10 Endpoint
- Secure Endpoint version v.8.0.1.21164

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

The scans were tested on a lab environment with Policy set to debug.
Flash scan on install was enabled via Connector download.

The scans were executed from the Secure Client GUI and from the Scheduler.

Full Scan

This log demonstrates when a Full scan is requested from the CSE Graphic User Interface (GUI).

```
(1407343, +0 ms) Aug 23 18:06:01 [9568]: Processing AMP_UI_SCAN action:
```

Scan from User Interface

Here, the ScanInitiator process begins the Scan process.

```
(1407343, +0 ms) Aug 23 18:06:01 [9568]: ScanInitiator::RequestScan: Attempting to start scan: dConnecte
```

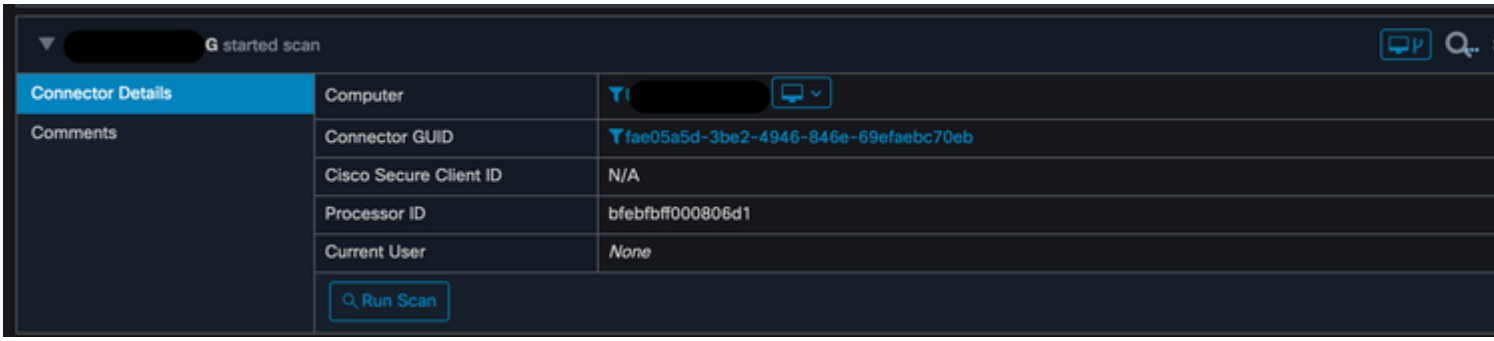
You can see that **Full Scan** is the type of Scan triggered on the GUI as shown in the image.

Next, you have the **Security Identifier** (**SID)**, which is a value of variable length assigned to this particular event, this Security Identifier helps you track the scan in the logs.

```
(1407343, +0 ms) Aug 23 18:06:01 [17268]: imn::CEventManager::PublishEvent: publis
json={"iclsa":"0","sce":108,"scx":"Full Scan","sid":1407343,"sit":2,"sop":0,"stp":
ui64EventId=7135211821471891460
```

Publish Event

You can match this with the event from the CSE console.



Console event

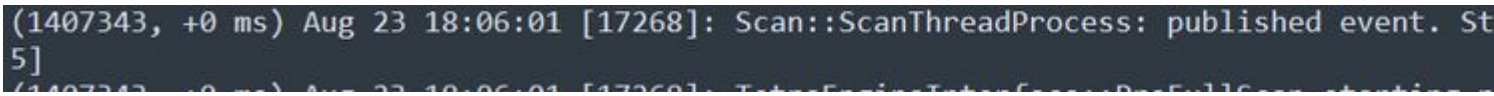
Next, in the logs, you can see this:

```
(1407343, +0 ms) Aug 23 18:06:01 [17268]: PublishScanStartEvent publishing event suc
```

Publish Succeeded

This means that the event has been successfully published to the CSE Cloud.

Then, the next action is actually to perform the scan:

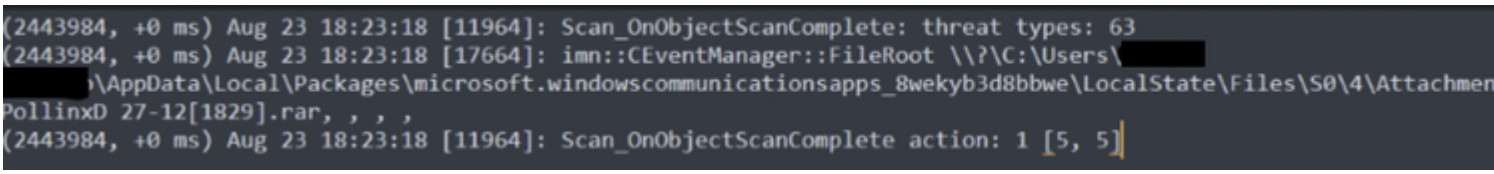


Scan start

As you can notice, the SID is the same, so you are under the stream of SID **1407343.**

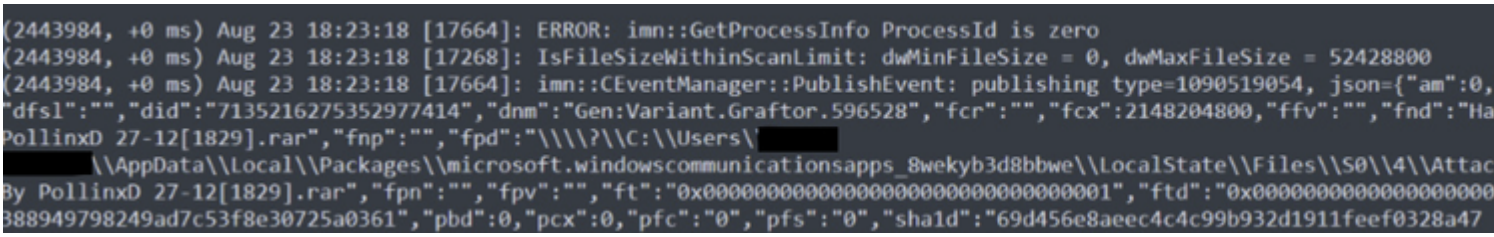
These are the steps that the connector performs when a Threat is detected during the Scan.

Step 1. The connector tells you the File that caused the detection, in this example, it is caused by **Hacksantana Trainer GLS**.

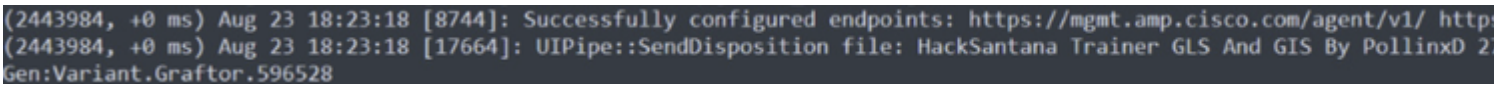


File detected

Step 2. The event is published to the CSE console with the Threat Detection name and the path where it is found.



Detection name



Threat event publish

After the scan finishes, you can take a look at the Event viewer, for a summary of the Scan.

In this example, you can see when the Scan starts, and like previously, a SID is given, this time, with a value of **2458015.**

```
(2458015, +0 ms) Aug 24 19:21:19 [17500]: Scan::ScanThreadProcess: beginning scan id: 2458015, [type: 1, opt
```

Flash scan start

The next action is to publish the event to the CSE cloud.

```
(2458015, +0 ms) Aug 24 19:21:19 [17500]: imn::CEventManager::PublishEvent: publishing type=554696714, json={"ic
Scan","sid":2458015,"sit":2,"sop":3,"stp":1}, ui64EventId=7135602311308509188
```

When the Scan finishes, the Event is published to the cloud.

```
(2458015, +0 ms) Aug 24 19:21:19 [17500]: imn::CEventManager::PublishEvent: publishing type=554696714, json={"ic
Scan","sid":2458015,"sit":2,"sop":3,"stp":1}, ui64EventId=7135602311308509188
```

Scan Finish Publish

The event can be seen in the Windows Event viewer. As you can notice, the information is the same as the information presented in the logs.

```
- <EventData>
    <Data Name="JsonEvent">{"dios":0,"ds":0,"hi":0,"scx":"Flash Scan","sdds":0,"sdfs":10951,"sdps":215,"sid":2458015,"s
      </Data>
    <Data Name="EventTypeId">554696715</Data>
    <Data Name="TimeStamp">133058605022030000</Data>
    <Data Name="EventId">7135602410092756997</Data>
    <Data Name="Description">EVENT_SCAN_COMPLETED_CLEAN</Data>
  </EventData>
</Event>
```

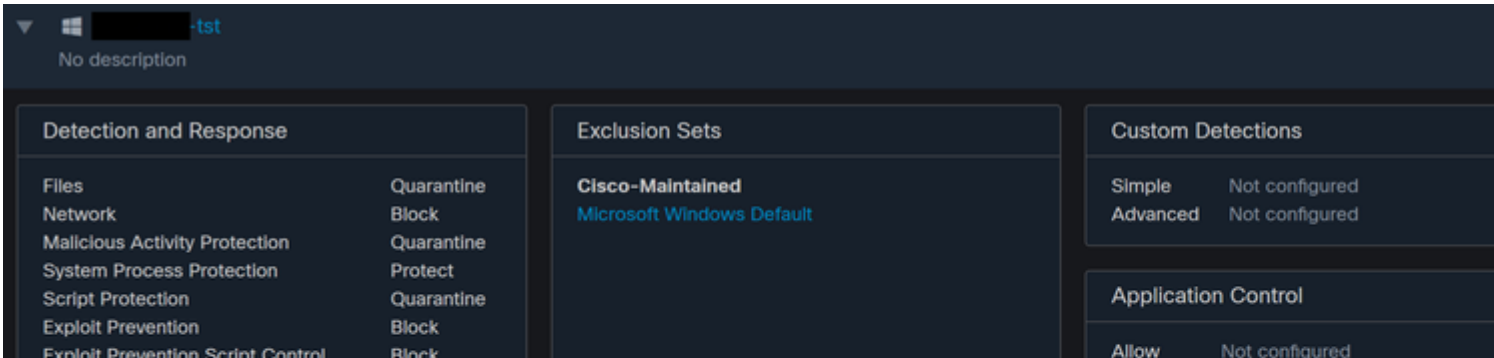
JSON Event

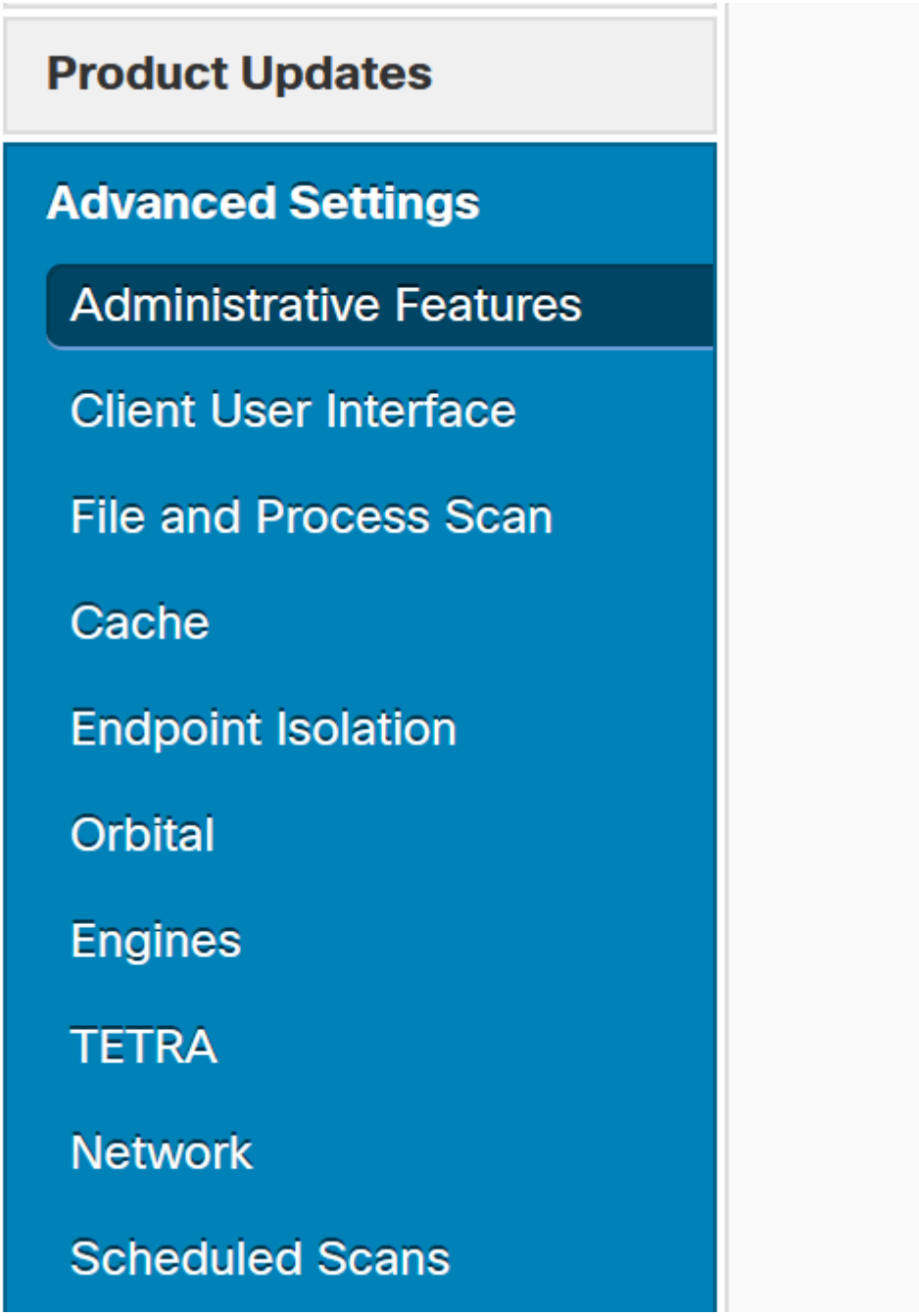
Scheduled Scans

When it comes to Scheduled scans, you must be aware of a set of aspects.

After a Scan is scheduled, a change in the Serial number occurs.

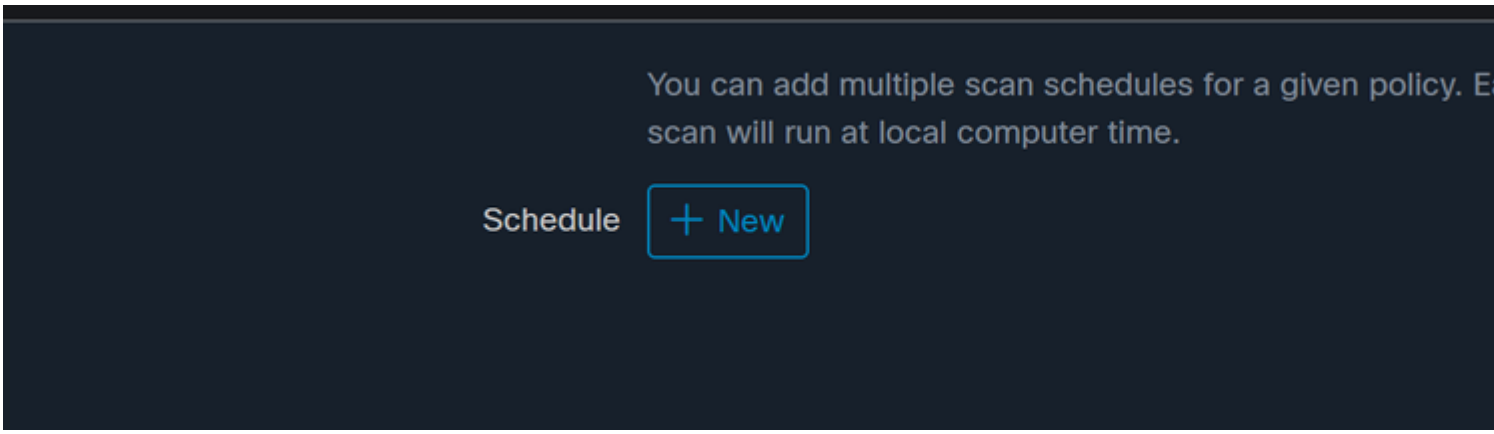
Here, the test policy does not have any Scheduled Scans.





Advanced Settings

Click **New.**

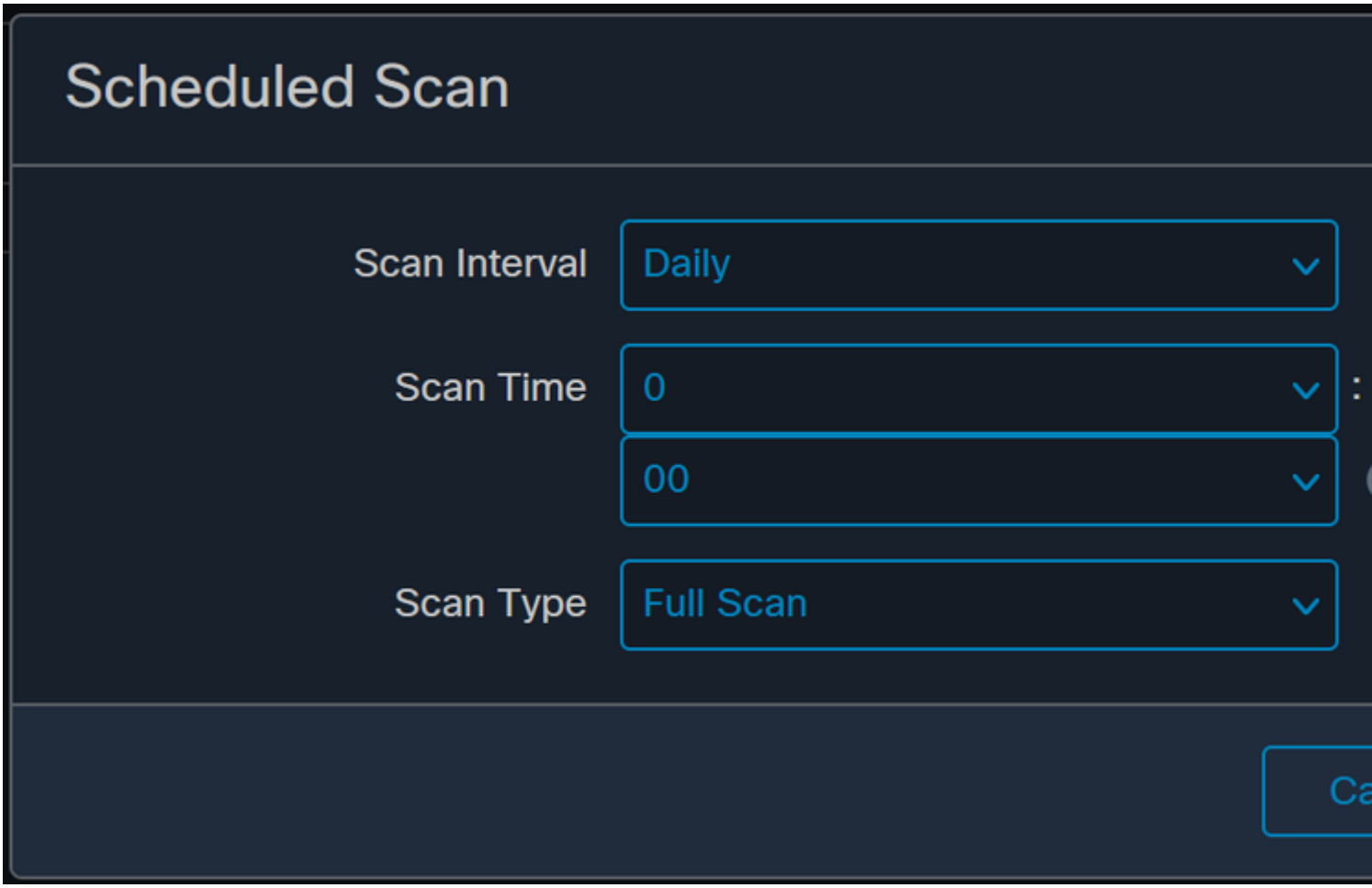


New Scan Configuration

The options are:

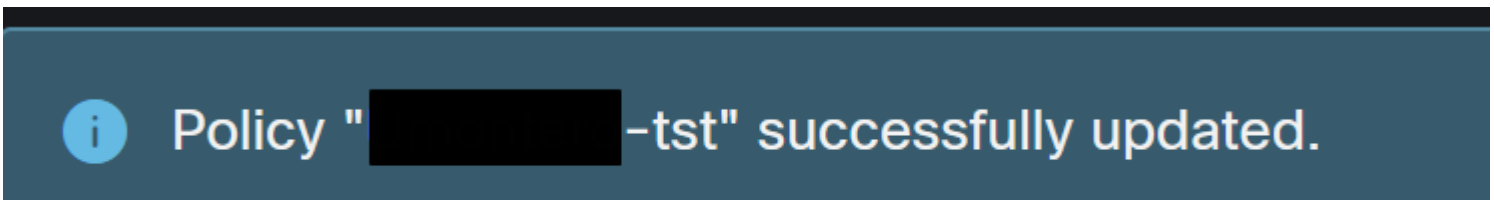
- Scan Interval
- Scan Time
- Scan Type

After you have configured your Scan, click **Add**.



Scheduled Scan Configuration

Save your policy changes, a pop-up appears that confirms your changes.



```
- <EventData>
    <Data Name="JsonEvent">{"iclsa":"0","sce":108,"scx":"Flash Scan","sid":86616093,"sit":4,"sop
    <Data Name="EventTypeId">554696714</Data>
    <Data Name="TimeStamp">133059446390220000</Data>
    <Data Name="EventId">7135963775756140548</Data>
    <Data Name="Description">EVENT_SCAN_STARTED</Data>
  </EventData>
</Event>
```


Cloud View

Once the Scan finishes, you can see the event published to the cloud.


```
(86641515, +0 ms) Aug 25 18:44:24 [3116]: imn::CEventManager::PublishEvent: publishing type=554696715, json={"
Scan","sdds":0,"sdfs":11575,"sdps":218,"sid":86616093,"sios":0,"sit":4,"sop":3,"sspc":0,"stp":1}, ui64EventId=
```


Scan Finish Publish

Scheduled Full Scan

The Windows event viewer shows **Event Scan Started**, as shown in the image.


```
- <EventData>
    <Data Name="JsonEvent">{"iclsa":"0","sce":108,"scx":"Full Scan","sid":87216125,"sit":4,"sop":0,
    <Data Name="EventTypeId">554696714</Data>
    <Data Name="TimeStamp">133059452390500000</Data>
    <Data Name="EventId">7135966352736518152</Data>
    <Data Name="Description">EVENT_SCAN_STARTED</Data>
  </EventData>
</Event>
```


Once it finishes, you can compare the published event.


```
(88165093, +0 ms) Aug 25 19:09:48 [18536]: imn::CEventManager::PublishEvent: publishing type=1091567628, json={
Scan","sdds":46012,"sdfs":280196,"sdps":224,"sid":87216125,"sios":0,"sit":4,"sop":0,"sspc":0,"stp":5}, ui64Even
```


You can see this in the event viewer from Windows.


```
- <EventData>
    <Data Name="JsonEvent">{"dios":0,"ds":2,"hi":0,"scx":"Full
      Scan","sdds":46012,"sdfs":280196,"sdps":224,"sid":87216125,"sios":0,"sit":4,"sop":0,"sspc":0,
```