User and Entity Behavior Analysis

-
- Information About User and Entity Behavior Analysis , on page 1
- Configuring User and Entity Behavior Analysis (Using UDP Collector), on page 1
- Configuring User and Entity Behavior Analysis (Using Stealthwatch Cloud), on page 2
- Mapping Stealthwatch Cloud to Flow Measurements, on page 3
- Example: Stealthwatch Cloud Configuration , on page 4
- Verifying Stealthwatch Cloud Details, on page 5

Information About User and Entity Behavior Analysis

User and Entity Behavior Analysis (UEBA) is a solution that has a number of security techniques, which allow you to profile and track the behavior of users and devices, in order to identify potential inside threats and targeted attacks in networks, when anomalies occur.

For instance, employees of an enterprise may unintentionally download a malicious piece of software that might include some backdoor or leakage in company secrets. This is detected by the change in the pattern of communication from one or more devices or users in the network, compared to an established baseline.

User and Entity Behavior Analysis can be deployed using two methods:

- User Datagram Protocol (UDP) collector (Cisco Digital Network Architecture (DNA) Center is a UDP collector)
- Stealthwatch Cloud (SwC) - The Embedded Wireless Controller (EWC) directly uploads data to SwC.

Configuring User and Entity Behavior Analysis (Using UDP Collector)

In a Cisco DNA Center-based deployment, the controller acts as the collector of NetFlow information that is sent to Cisco DNA Center. In turn, Cisco DNA Center compresses the information for SwC. The controller enables Application Visibility and Control (AVC) on the access points (APs) and maintains the communication channel with Cisco DNA Center.

In EWC, you can also send FnFv9 data through the UDP to a UDP collector.

In the Non-Cisco DNA-C based deployment, the FnF flow records are directly sent to SwC from the controller.

Configuring User and Entity Behavior Analysis (Using Stealthwatch Cloud)

The following sections provide information about configuring the User and Entity Behavior Analysis solution using Stealthwatch Cloud (GUI and CLI).

Configuring User and Entity Behavior Analysis Using Stealthwatch Cloud (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Threat Defense**.

Step 2 Click **Cisco StealthWatch Integration**.

Step 3 On the Stealthwatch page, in the **Service Key** field, enter the Stealthwatch cloud service key.

Step 4 Click the cloud icon to view the detailed statistics of Stealthwatch.

Step 5 In the **Sensor Name** field, enter a sensor name for Stealthwatch Cloud registration.

Step 6 In the **URL** field, enter the Stealthwatch Cloud server URL.

Step 7 Click **Apply**.

Step 8 (Optional) Click **Unconfigure StealthWatch**, to unconfigure Stealthwatch Cloud.

What to do next

You can view and verify the Stealthwatch Cloud's health status in the **Stealthwatch Health Status**

Configuring Stealthwatch Cloud (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**stealthwatch-cloud-monitor** **Example:** `Device(config)# stealthwatch-cloud-monitor`	Configures the Stealthwatch Cloud monitor. Enters the Stealthwatch Cloud Monitor configuration mode.

	Command or Action	Purpose
Step 3	**service-key** *swc-service-key* **Example:** `Device(config-stealthwatch-cloud-monitor)# service-key xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx`	(Optional) Sets the Stealthwatch Cloud service key. Service key is provided by the SwC portal. The alternative to service key is the authentication through the IP address allowed list. For more information about service key and allowed lists, see the appropriate SwC guide.
Step 4	**sensor-name** *swc-sensor-name* **Example:** `Device(config-stealthwatch-cloud-monitor)# sensor-name swc-sensor-name`	(Optional) Provides a sensor name for the Stealthwatch Cloud registration. The device serial number is the default value.
Step 5	**url** *SwC-server-url* **Example:** `Device(config-stealthwatch-cloud-monitor)# url https://sensors.eu-2.obsrvbl.com`	Sets the Stealthwatch Cloud server URL.

Mapping Stealthwatch Cloud to Flow Measurements

There are two options to map Stealthwatch Cloud to flow measurements, namely the flow-exporter configuration and the flow-monitor configuration.

Note At any given period, there can be only one internal and one external active flow exporter. An active flow exporter is an exporter that is bound to the flow monitor that is bound to a wireless profile.

Configuring Flow Exporter for Stealthwatch Cloud

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow exporter** *flow-exporter-name* **Example:**	Defines the flow exporter.

	Command or Action	Purpose
	`Device(config)# flow exporter flow-exporter-name`	**Note** At a given moment, there can be only one internal and one external active flow exporter. An active flow exporter is an exporter that is bound to the flow monitor, which is bound to a wireless profile.
Step 3	**destination stealthwatch-cloud** **Example:** `Device(config-flow-exporter)# destination stealthwatch-cloud`	Exports the flow information to Stealthwatch Cloud.

Configuring Flow Monitor for Stealthwatch Cloud

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow monitor** *flow-monitor-name* **Example:** `Device(config)# flow monitor flow-monitor-name`	Defines the flow monitor.
Step 3	**exporter** *flow-exporter-name* **Example:** `Device(config-flow-monitor)# exporter flow-exporter-name`	Exports the flow information to the exporter.
Step 4	**record wireless avc basic** **Example:** `Device(config-flow-monitor)# record wireless avc basic`	Specifies the flow record with basic IPv4 wireless AVC template.
Step 5	**end** **Example:** `Device(config-flow-monitor)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Example: Stealthwatch Cloud Configuration

The following example shows a complete CLI configuration of Stealthwatch Cloud:

```
stealthwatch-cloud-monitor
  service-key XXXXXXXXXXXXXXXXXXXXXXXXXXXX
  sensor-name ewc-sensor
  url https://sensors.eu-2.obsrvbl.com

flow exporter fexp-swc
 destination stealthwatch-cloud

flow monitor fm-avc-swc
 exporter fexp-swc
 record wireless avc basic

wireless profile policy swc-policy-profile
 ipv4 flow monitor fm-avc-swc input
 ipv4 flow monitor fm-avc-swc output
 ipv6 flow monitor fm-avc-swc input
 ipv6 flow monitor fm-avc-swc output

wlan my-wlan 1 my-wlan

wireless tag policy swc-policy-tag
 wlan my-wlan policy swc-policy-profile

ap 0000.0000.0001
 policy-tag swc-policy-tag
```

Verifying Stealthwatch Cloud Details

To verify the state and statistics of Stealthwatch Cloud, use the **show stealthwatch-cloud wireless-shim** command:

```
Device# show stealthwatch-cloud wireless-shim
Stealthwatch-Cloud wireless shim

Total
 RX records        : 15
 RX bytes          : 2345
 TX records        : 10
 TX bytes          : 1234
 TX batches        : 1
 Failed batches    : 0
 Non-SWC records   : 5

Buffers
 Status        : TX
 Size          : 1272000
 Compressed    : 8
 Uncompressed  : 0
 Records       : 8

 Status        : Filling
 Size          : 1272000
 Compressed    : 2
 Uncompressed  : 0
 Records       : 2
```

To verify the Stealthwatch Cloud connection details, use the **show stealthwatch-cloud connection** command.

```
Device# show stealthwatch-cloud connection
Stealthwatch-Cloud details
  Registration
    #ID         : 0xe6000001
```

```
    URL         : https://sensors.eu-2.obsrvbl.com
    Service Key : XXXXXXXXXXXXXXXXXXXXXXXXXXXX
    Sensor Name : ewc-sensor
    Registered  : Yes
  Connection
    Status               : UP
    Last status update   : 03/17/2020 21:44:55
    # Flaps              : 0
    # Heartbeats         : 9
    # Lost heartbeats    : 1
    Total RX bytes       : 4567
    Total TX bytes       : 1234
    Upload Speed (B/s)   : 247
    Download Speed (B/s) : 269
    # Open sessions      : 0
    # Redirections       : 0
    # Timeouts           : 0

  HTTP Events
    GET response                  : 1
    GET request                   : 1
    GET Status Code 2XX           : 1
    PUT response                  : 1
    PUT request                   : 1
    PUT Status Code 2XX           : 1
    POST response                 : 12
    POST request                  : 12
    POST Status Code 2XX          : 11
    POST Status Code 4XX          : 1
API Events
    Abort                         : 1

  Event History
Timestamp               #Times   Event               RC Context
----------------------- -------- ------------------- --
----------------------------------------
03/21/2020 10:42:06.161 9        HEARTBEAT_OK         0

03/20/2020 06:49:05.717 1        HEARTBEAT_FAIL       0 HTTPCON_EV_TIMEOUT (6)

03/20/2020 06:47:05.717 1        SEND_START           0 ID:0001
03/20/2020 06:49:05.717 3        SIGNAL_DATA_FAIL     0 ID:0001, attempt : 3
03/18/2020 09:23:39.375 1        REGISTER_OK          0

03/18/2020 09:23:13.276 1        REGISTER_SEND        0

03/18/2020 09:23:12.154 1        SEND_ABORT_ALL       0 config change

03/18/2020 09:23:12.154 1       OPTIONS_CONFIG      0 URL https://sensor.staging.obsrvbl.com

03/18/2020 09:23:12.154 1       OPTIONS_CONFIG      0 Service-key XXXXXXXXXXXXXXXXXXXXXXXX
03/18/2020 09:23:12.154 1        OPTIONS_CONFIG       0 Host ewc-sensor => reset

03/18/2020 09:23:12.154 1        OPTIONS_CONFIG       0 cfg-mode manual => reset
```