Resolución de problemas de flujo de eventos en la nube privada

Contenido

Introducción

Este documento describe cómo resolver problemas de flujos de eventos en Advanced Malware Protection Secure Endpoint Private Cloud.

Prerequisites

Requirements

Cisco recomienda que conozca los siguientes temas:

- Nube privada de terminal seguro
- consulta de API

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Nube privada de terminal seguro v3.9.0
- cURL v7.87.0
- cURL v8.0.1

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Configuración

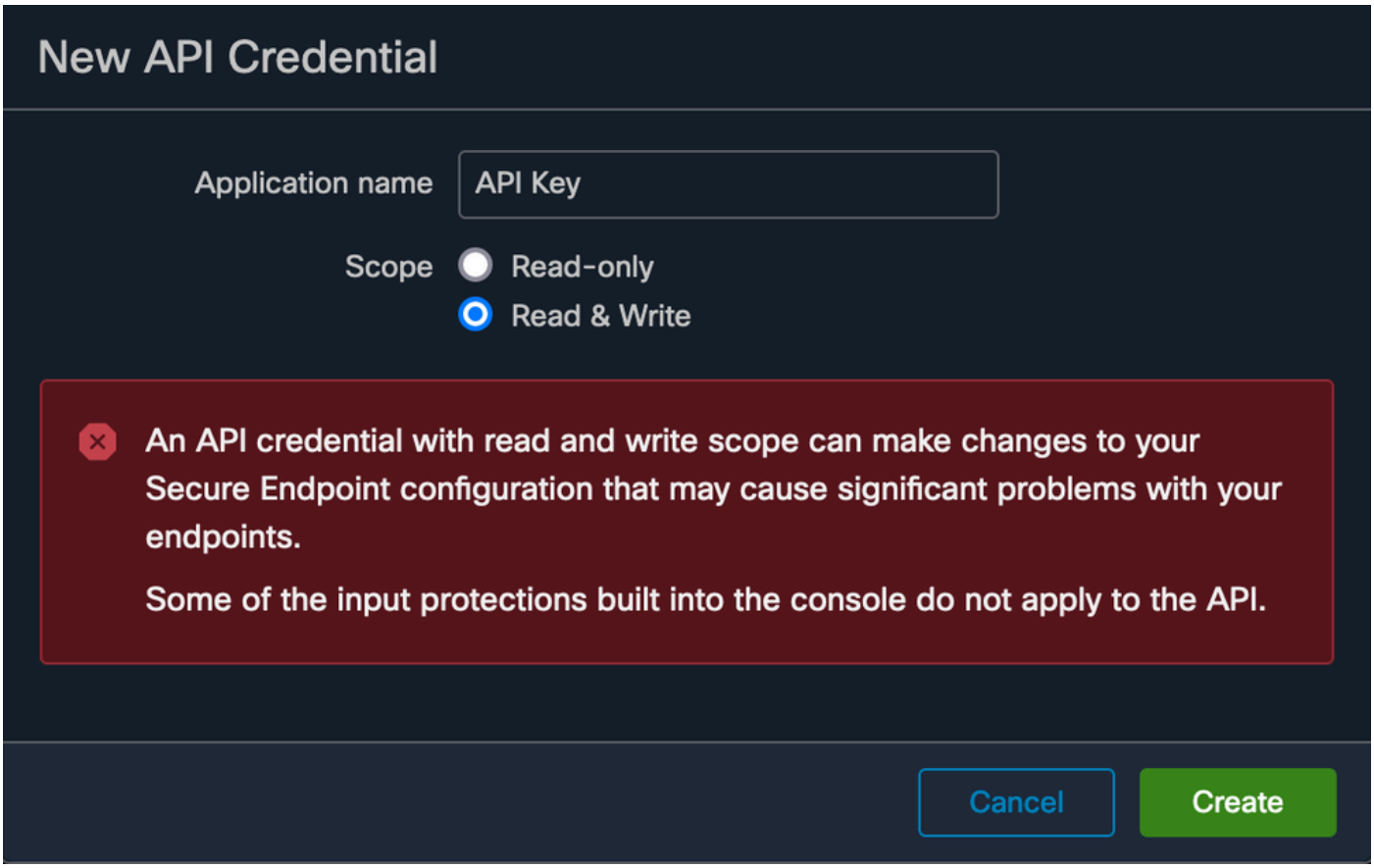
Crear clave de API

Paso 1. Inicie sesión en la Consola de nube privada.

Paso 2. Desplácese hasta Accounts > API Credentials.

Paso 3. Haga clic en New API Credential.

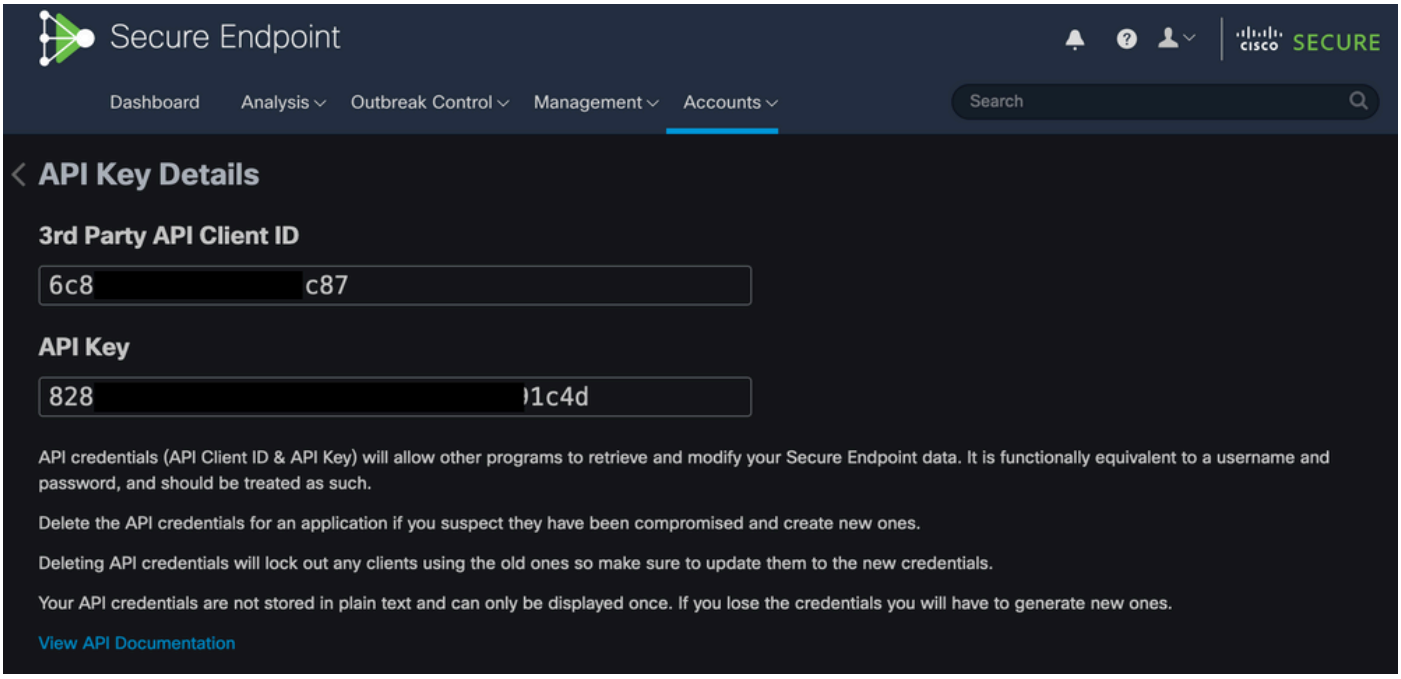
Paso 4. Agregue el Application name y haga clic en Read & Write alcance.



Crear clave de API

Paso 5. Haga clic en Create.

Paso 6. Guarde las credenciales de API.



Clave de API

Precaución: la clave de la API no se puede recuperar si sale de esta página.

Crear flujo de eventos

Esto crea una nueva secuencia de mensajes del protocolo avanzado de Message Queue Server (AMQP) para la información de eventos.

Puede crear un flujo de eventos para los tipos y grupos de eventos especificados:

```
--data '{"name":"EVENT_STREAM_NAME","event_type":["EVENT_TYPE_1", "EVENT_TYPE_2"],"group_guid":["GROUP_
```

Puede crear un flujo de eventos para todos los tipos de eventos y todos los grupos mediante:

```
--data '{"name":"EVENT_STREAM_NAME","event_type":[],"group_guid":[]}'
```

MacOS/Linux

Puede crear un flujo de eventos en MacOS/Linux con el uso de:

```
curl -X POST -k -H 'Accept: application/json' -H 'Content-Type: application/json' -u 'CLIENT_ID:API_KEY
```

Windows:

Puede crear un flujo de eventos en Windows con el uso de:

```
curl -X POST -k -H "Accept: application/json" -H "Content-Type: application/json" -u "CLIENT_ID:API_KEY
```

Respuesta

```
HTTP/1.1 201 Created
(...)
"data": {
  "id": 17,
  "name": "EVENT_STREAM_NAME",
  "amqp_credentials": {
    "user_name": "17-1bfXXXXXXXXX",
```

```
    "queue_name": "event_stream_17",
    "password": "3961XXXXXXXXXXXXXXXXXXXX814a77",
    "host": "FMC_SERVICE_URL",
    "port": 443,
    "proto": "https"
  }
}
```

Lista de flujos de eventos

Muestra una lista de los flujos de eventos creados en la nube privada.

MacOS/Linux

Puede enumerar los flujos de eventos en MacOS/Linux con el uso de:

```
curl -k -H 'Accept: application/json' -H 'Content-Type: application/json' -u 'CLIENT_ID:API_KEY' -i 'ht
```

Windows:

Puede enumerar los flujos de eventos en Windows con el uso de:

```
curl -k -H "Accept:application/json" -H "Content-Type:application/json" -u "CLIENT_ID:API_KEY" -i "http
```

Respuesta

```
HTTP/1.1 200 OK
(...)
"data": {
  "id": 17,
  "name": "EVENT_STREAM_NAME",
  "amqp_credentials": {
    "user_name": "17-1bfXXXXXXXXX",
    "queue_name": "event_stream_17",
    "host": "FMC_SERVICE_URL",
    "port": 443,
    "proto": "https"
  }
}
```

Eliminar flujos de eventos

Elimina una secuencia de eventos activa.

MacOS/Linux

Puede eliminar Event Streams en MacOS/Linux con el uso de:

```
curl -X DELETE -k -H 'Accept: application/json' -H 'Content-Type: application/json' -u 'CLIENT_ID:API_K
```

Windows:

Puede eliminar flujos de eventos en Windows con el uso de:

```
curl -X DELETE -k -H "Accept:application/json" -H "Content-Type:application/json" -u "CLIENT_ID:API_KEY
```

Respuesta

```
HTTP/1.1 200 OK
(...)
 "data": {}
```

Verificación

Paso 1. Copie el script Python en su dispositivo y guárdelo como EventStream.py.

```
import pika
import ssl

user_name = "USERNAME"
queue_name = "QUEUE_NAME"
password = "PASSWORD"
host = "FMC_SERVICE_URL"
port = 443
proto = "https"

def callback(channel, method, properties, body):
print(body)

amqp_url = f"amqps://{user_name}:{password}@{host}:{port}"

context = ssl.SSLContext(ssl.PROTOCOL_TLSv1_2)
amqp_ssl = pika.SSLOptions(context)
```

```
params = pika.URLParameters(amqp_url)
params.ssl_options = amqp_ssl

connection = pika.BlockingConnection(params)
channel = connection.channel()

channel.basic_consume(
   queue_name,
   callback,
   auto_ack = False
)

channel.start_consuming()
```

Paso 2. Ejecutarlo en el terminal como `python3 EventStream.py`.

Paso 3. Desencadenar cualquier evento que se agregue a la cola de flujo de eventos.

Paso 4. Verifique si los eventos aparecen en el terminal.

Resolución de problemas

Para ejecutar estos comandos, debe iniciar sesión a través de SSH en la nube privada.

Compruebe el servicio AMQP

Verifique si el servicio está habilitado:

```
[root@fireamp rabbitmq]# amp-ctl service status rabbitmq
running enabled rabbitmq
```

Verifique si el servicio se está ejecutando:

```
[root@fireamp ~]# svstat /service/rabbitmq
/service/rabbitmq: up (pid 25504) 7402137 seconds
```

Comprobar la conexión con el receptor de flujo de eventos

Ejecute el comando:

```
tail /data/log/rabbitmq/rabbit@fireamp.log
```

Se establece la conexión:

```
=INFO REPORT==== 19-Apr-2023::08:40:12 ===
accepting AMQP connection <0.17588.27> (127.0.0.1:32946 -> 127.0.0.1:5672)
```

La conexión está cerrada:

```
=WARNING REPORT==== 19-Apr-2023::08:41:52 ===
closing AMQP connection <0.17588.27> (127.0.0.1:32946 -> 127.0.0.1:5672):
connection_closed_abruptly
```

Comprobar los eventos de la cola

Los eventos de la cola están listos para enviarse en esta secuencia de eventos al receptor después de establecer la conexión. En este ejemplo, hay 14 eventos para Event Stream ID 23.

<#root>

```
[root@fireamp rabbitmq]# rabbitmqctl list_queues
Listing queues ...
1acb0eb6-39f7-4b11-bd9b-fc4dd0e3bd77_60b15rn8mpftaico6or6l8zxavl1usm 26
1acb0eb6-39f7-4b11-bd9b-fc4dd0e3bd77_61984nlu8p11eeopmgmtcjra1v8gf5p 26
1acb0eb6-39f7-4b11-bd9b-fc4dd0e3bd77_iesRAgVoOh287mO_DetOx9PdDu8MxkS6kL4oSTeBm9s 26
event_decoration 0
event_log_store 0
```

event_stream_23 14

```
event_streams_api 0
events_delayed 0
events_retry 0
mongo_event_consumer 0
out_events_q1 0
tevent_listener 0
```

Recopilar archivo de tráfico de red

Para verificar el tráfico de flujo de eventos desde la nube privada, puede recopilar la captura con un tcpdump herramienta:

Paso 1. SSH en la nube privada.

Paso 2. Ejecute el comando:

```
tcpdump -vvv -i eth1 host <Event_Stream_Receiver_IP> -w file.pcap
```

Paso 3. Detenga la captura con Ctrl+C (Windows) o Command-C (Mac).

Paso 4. Extraiga el pcap desde la nube privada.

Información Relacionada

- Configuración de la función AMP para terminales Event Stream
- Soporte Técnico y Documentación - Cisco Systems

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).