Configurar o AMP para a característica do córrego do evento dos valores-limite

Índice

Introdução

Este original descreve como configurar e consumir a característica do córrego do evento para a proteção avançada do malware (AMP) para valores-limite.

Pré-requisitos

Requisitos

Cisco recomenda que você tem o conhecimento dos seguintes assuntos:

- AMP para endpoints
- Conhecimento básico da programação do pitão

Componentes Utilizados

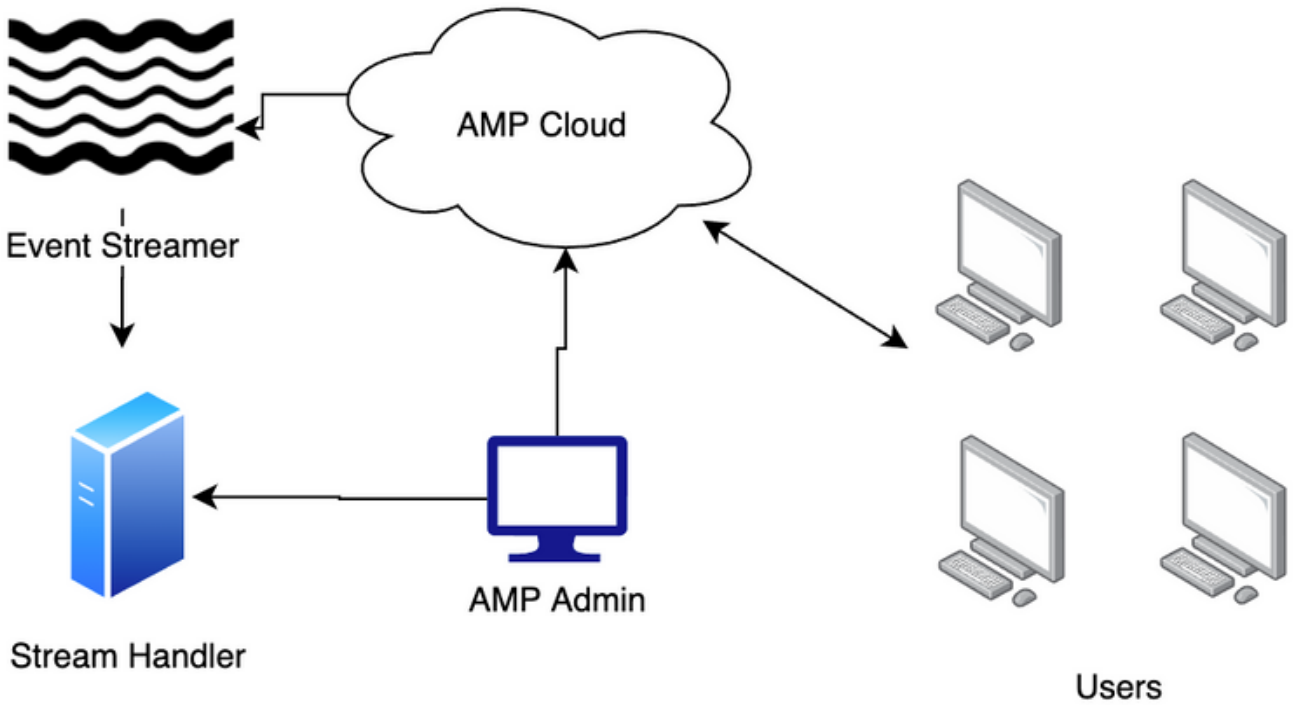
A informação neste documento é baseada no pitão 3.7 com o **pika** (versão 1.1.0) e as bibliotecas externos dos **pedidos** (versão 2.22.0).

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Configurar

Diagrama de Rede

Esta imagem fornece um exemplo de arranjar em sequência do córrego do evento:



Configurações

Crie credenciais API

1. Navegue a seu AMP para o portal e o início de uma sessão dos valores-limite
2. Sob **contas,** escolha **credenciais API**
3. Clique **credenciais novas API**
4. Incorpore um valor ao campo de **nome do aplicativo**
5. Selecione **lido & escreva** para o **espaço**
6. O clique **cria**
7. Armazene estas credenciais em um gerente da senha ou em um arquivo cifrado

Crie o córrego do evento

1. Abra um shell do pitão e importe o **json**, o **SSL**, **pika** e **pede** bibliotecas.

```
import json
import pika
import requests
import ssl
```

2. Armazene os valores para a URL, o client_id, e o api_key. Sua URL pode variar se você não está usando a nuvem norte-americana. Também, seus client_id e api_key são originais a seu ambiente.

```
url = "https://api.amp.cisco.com/v1/event_streams"
client_id = "d16aff14860af496e848"
api_key = "d01ed435-b00d-4a4d-a299-1806ac117e72"
```

3. Crie o objeto de dados para passar ao pedido. Isto deve incluir o nome, e pode incluir o event_type e o group_guid para restringir os eventos e os grupos incluídos no córrego. Se nenhum group_guid ou event_type são passados, o córrego do evento incluirá todos os grupos e tipos de evento.

```
data = {
    "name": "Event Stream for ACME Inc",
    "group_guid": ["5cdf70dd-1b14-46a0-be90-e08da14172d8"],
    "event_type": [1090519054]
}
```

4. Faça o CARGO pedir o atendimento, e armazene o valor em uma variável.

```
r = requests.post(
    url = url,
    data = data,
    auth = (client_id, api_key)
)
```

5. Imprima o código de status. Confirme que o código é 201.

```
print(r.status_code)
```

6. Carregue o índice da resposta em um objeto do json, e armazene esse objeto em uma variável.

```
j = json.loads(r.content)
```

7. Reveja os índices dos dados da resposta.

```
for k, v in j.items():
    print(f"{k}: {v}")
```

8. Os dados avançados do protocolo da fila de mensagens (AMQP) são dentro da resposta. Extraia os dados em variáveis respectivas.

```
user_name = j["data"]["amqp_credentials"]["user_name"]
queue_name = j["data"]["amqp_credentials"]["queue_name"]
password = j["data"]["amqp_credentials"]["password"]
host = j["data"]["amqp_credentials"]["host"]
port = j["data"]["amqp_credentials"]["port"]
proto = j["data"]["amqp_credentials"]["proto"]
```

9. Defina uma função de chamada de volta com estes parâmetros. Nesta instalação, você imprime o corpo do evento à tela. Contudo, você pode mudar este índice desta função para serir seus objetivos.

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

```
    print(body)
```

10. Prepare o AMQP URL das variáveis que você criou.

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

11. Prepare o contexto SSL

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

12. Prepare o córrego AMQP com os métodos da biblioteca do pika.

```
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
)
```

13. Inicie o córrego.

```
channel.start_consuming()
```

14. O córrego é agora vivo e esperando eventos.

Verificar

Provoque um evento em um valor-limite em seu ambiente. Por exemplo, inicie uma varredura instantânea. Observe que o córrego imprime os dados de evento à tela.

Pressione o **Ctrl+C** (Windows) ou o **comando C** (Mac) interromper o córrego.

Troubleshooting

Códigos de status

- Um código de status de 401 indica que há uma edição com autorização. Verifique seus **client_id** e **api_key**, ou gerencia chaves novas.
- Um código de status de 400 indica que há uma edição ruim do pedido. Certifique-se de você não tenha um córrego do evento criado com esse nome, ou que você não tem mais do que os córregos do evento 5 criados.