Automazione dell'isolamento di avvio/arresto su più endpoint

Sommario

Introduzione

In questo documento viene descritto come automatizzare l'isolamento stop/start su più endpoint utilizzando l'API per Cisco Secure Endpoint.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Cisco Secure Endpoint
- Cisco Secure Endpoint Console
- API Cisco Secure Endpoint
- Python

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software:

- Cisco Secure Endpoint 8.4.0.30201
- Ambiente Python da endpoint a host
- Python 3.11.7

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali

conseguenze derivanti dall'uso dei comandi.

Premesse

- Utilizzare una richiesta PUT per avviare l'isolamento.
- Una richiesta DELETE viene utilizzata per interrompere l'isolamento.
- Per ulteriori informazioni, consultare la documentazione dell'API.

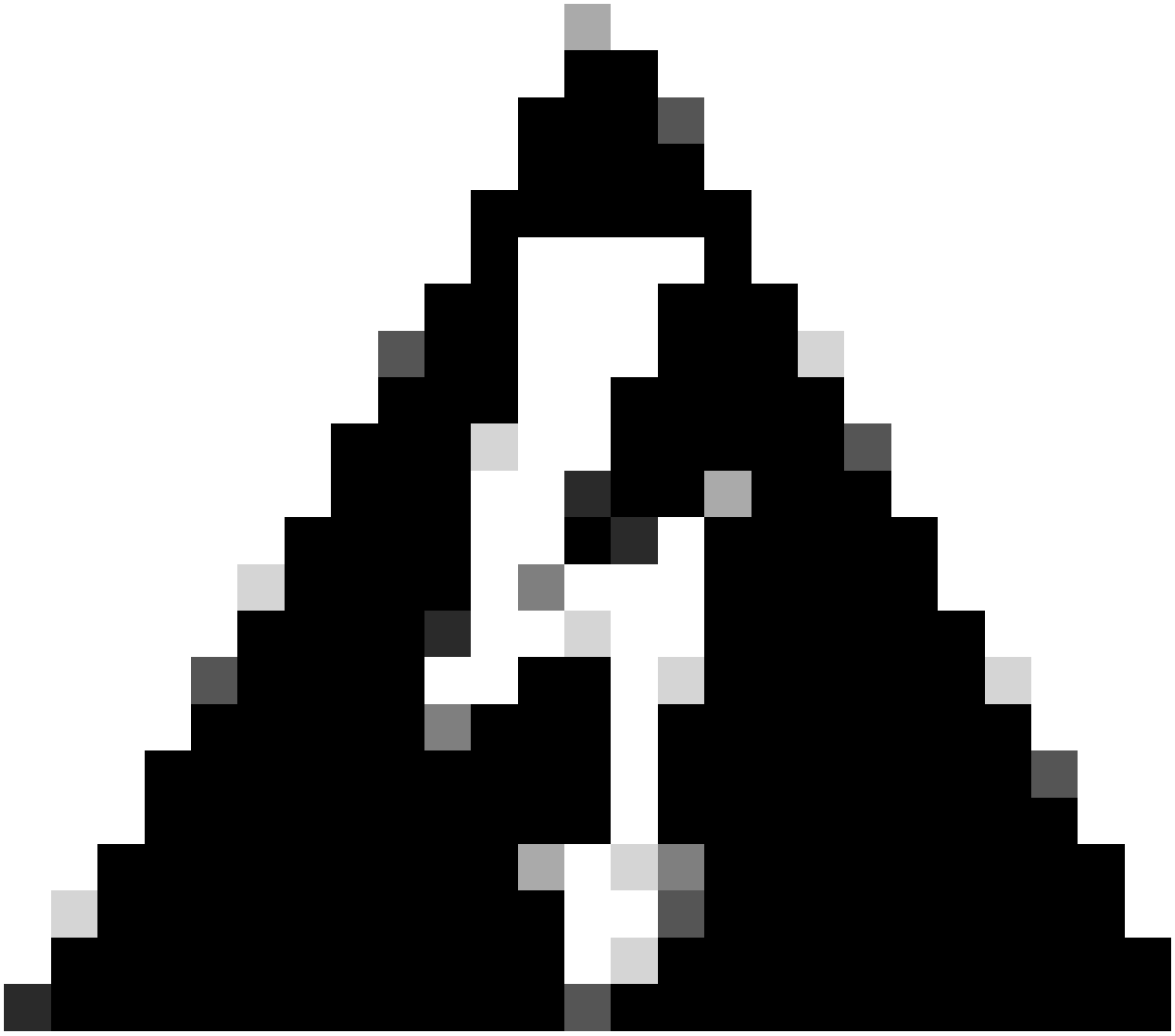
Problema

Cisco Secure Endpoint consente l'isolamento start/stop su un computer alla volta. Tuttavia, durante un problema di sicurezza, è spesso necessario eseguire queste operazioni su più endpoint contemporaneamente per contenere efficacemente le potenziali minacce. L'automazione del processo di isolamento di avvio/arresto per gli endpoint in blocco tramite l'API può migliorare in modo significativo l'efficienza della risposta agli incidenti e ridurre il rischio complessivo per la rete.

Soluzione

- Lo script Python fornito in questo articolo può essere utilizzato per avviare/terminare l'isolamento su più endpoint nell'organizzazione utilizzando le credenziali API Secure Endpoint.
- Per generare le credenziali dell'API AMP, consultare la panoramica dell'API Cisco AMP for Endpoints
- Per utilizzare lo script fornito, è necessario installare pythonsugli endpoint.
- Dopo l'installazione di python, installare il modulo delle richieste

```
pip install requests
```



Avviso: lo script viene fornito solo a scopo illustrativo e ha lo scopo di dimostrare come automatizzare la funzionalità di isolamento dell'endpoint utilizzando l'API. Cisco Technical Assistance Center (TAC) non è coinvolto nella risoluzione dei problemi relativi a questo script. Prima di distribuire lo script in un'impostazione di produzione, gli utenti devono procedere con cautela ed eseguire un test approfondito dello script in un ambiente sicuro.

Script

È possibile utilizzare lo script fornito per avviare l'isolamento su più endpoint nell'azienda:

```
import requests

def read_config(file_path):
    """
    Reads the configuration file to get the API base URL, client ID, and API key.
    """
    config = {}
    try:
```

```
        with open(file_path, 'r') as file:
            for line in file:
                # Split each line into key and value based on '='
                key, value = line.strip().split('=')
                config[key] = value
    except FileNotFoundError:
        print(f"Error: Configuration file '{file_path}' not found.")
        exit(1)  # Exit the script if the file is not found
    except ValueError:
        print(f"Error: Configuration file '{file_path}' is incorrectly formatted.")
        exit(1)  # Exit the script if the file format is invalid
    return config

def read_guids(file_path):
    """
    Reads the file containing GUIDs for endpoints to be isolated.
    """
    try:
        with open(file_path, 'r') as file:
            # Read each line, strip whitespace, and ignore empty lines
            return [line.strip() for line in file if line.strip()]
    except FileNotFoundError:
        print(f"Error: GUIDs file '{file_path}' not found.")
        exit(1)  # Exit the script if the file is not found
    except Exception as e:
        print(f"Error: An unexpected error occurred while reading the GUIDs file: {e}")
        exit(1)  # Exit the script if an unexpected error occurs

def isolate_endpoint(base_url, client_id, api_key, connector_guid):
    """
    Sends a PUT request to isolate an endpoint identified by the connector GUID.
    Args:
        base_url (str): The base URL for the API.
        client_id (str): The API client ID for authentication.
        api_key (str): The API key for authentication.
        connector_guid (str): The GUID of the connector to be isolated.
    """
    url = f"{base_url}/{connector_guid}/isolation"
    try:
        # Send PUT request with authentication
        response = requests.put(url, auth=(client_id, api_key))
        response.raise_for_status()  # Raise an HTTPError for bad responses (4xx and 5xx)

        if response.status_code == 200:
            print(f"Successfully isolated endpoint: {connector_guid}")
        else:
            print(f"Failed to isolate endpoint: {connector_guid}. Status Code: {response.status_code},
    except requests.RequestException as e:
        print(f"Error: An error occurred while isolating the endpoint '{connector_guid}': {e}")

if __name__ == "__main__":
    # Read configuration values from the config file
    config = read_config('config.txt')

    # Read list of GUIDs from the GUIDs file
    connector_guids = read_guids('guids.txt')

    # Extract configuration values
    base_url = config.get('BASE_URL')
    api_client_id = config.get('API_CLIENT_ID')
    api_key = config.get('API_KEY')
```

```
    # Check if all required configuration values are present
    if not base_url or not api_client_id or not api_key:
        print("Error: Missing required configuration values.")
        exit(1)  # Exit the script if any configuration values are missing

    # Process each GUID by isolating the endpoint
    for guid in connector_guids:
        isolate_endpoint(base_url, api_client_id, api_key, guid)
```

Istruzione

- Per generare le credenziali dell'API AMP, consultare la panoramica dell'API Cisco AMP for Endpoints
- Utilizzare BASE_URL menzionato per il proprio paese:

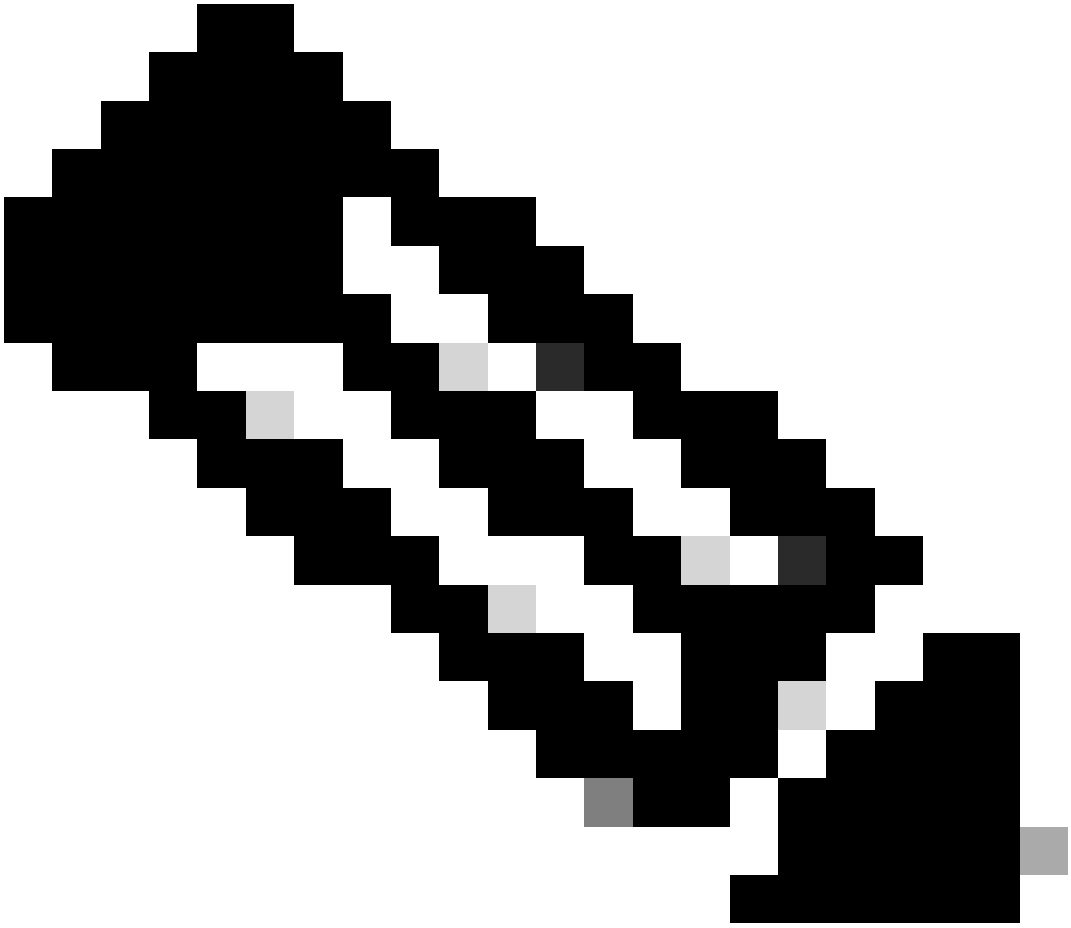
```
NAM - https://api.amp.cisco.com/v1/computers/
EU - https://api.eu.amp.cisco.com/v1/computers/
APJC - https://api.apjc.amp.cisco.com/v1/computers/
```

- Creare un file config.txt nella stessa directory dello script con il contenuto indicato. Esempio di file config.txt:

```
BASE_URL=https://api.apjc.amp.cisco.com/v1/computers/
API_CLIENT_ID=xxxxxxxxxxxxxxxxxxxx
API_KEY=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
```

- Creare un file guids.txt nella stessa directory dello script con l'elenco dei GUID dei connettori, uno per riga. Aggiungere tutti i GUID necessari. Esempio di file guids.txt:

```
abXXXXXXXXXXXXcd-XefX-XghX-X12X-XXXXXX567XXXXXXX
yzXXXXXXXXXXXXlm-XprX-XmnX-X34X-XXXXXX618XXXXXXX
```



Nota: è possibile raccogliere i GUID degli endpoint tramite l'API GET /v1/computers o da Cisco Secure Endpoint Console passando a Gestione > Computer, espandendo la voce per un endpoint specifico e copiando il GUID del connettore.

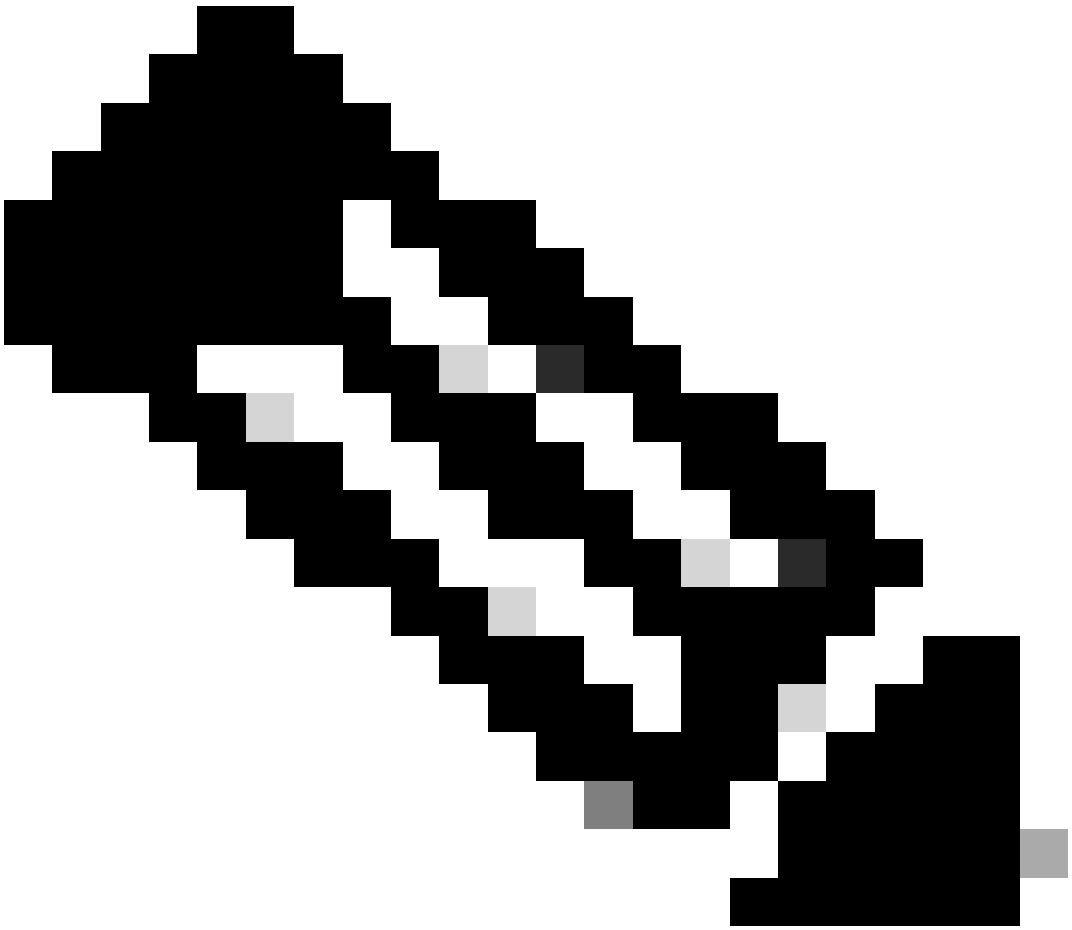
- Aprire un terminale o un prompt dei comandi. Passare alla directory in cui si trova start_isolation_script.py.
- Eseguire lo script eseguendo il comando indicato:

```
python start_isolation_script.py
```

Verifica

- Lo script tenta di isolare ogni endpoint specificato nel file guids.txt.
- Controllare il terminale o il prompt dei comandi per i messaggi di riuscita o di errore per

ciascun endpoint.



Nota: lo script allegato start_isolation.py può essere utilizzato per avviare l'isolamento sugli endpoint, mentre stop_isolation.py è progettato per arrestare l'isolamento sugli endpoint. Tutte le istruzioni per l'esecuzione dello script rimangono invariate.

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).