Configure o acesso seguro para usar a API REST com Python

Contents

Introdução

Este documento descreve as etapas para configurar o acesso à API e usá-lo para buscar informações de recursos no Secure Access.

Pré-requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

1. Python 3. x
2. API REST
3. Acesso seguro da Cisco

Requisitos

Estes requisitos devem ser cumpridos antes de prosseguir:

- Conta de usuário do Cisco Secure Access com a função de Administrador Completo.
- Conta do Cisco Security Cloud Single Sign On (SCSO) para entrar no Secure Access.

Componentes Utilizados

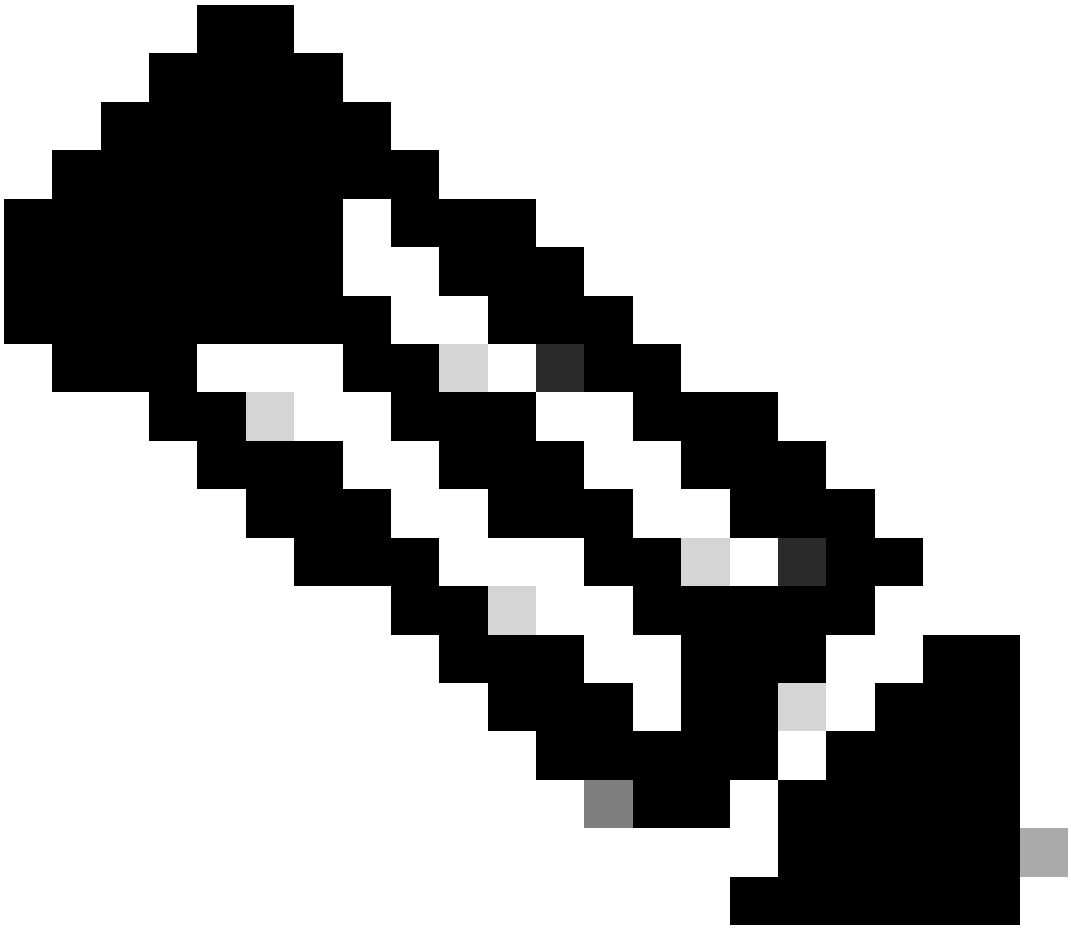
As informações neste documento são baseadas nestas versões de software e hardware:

- Painel de acesso seguro
- Python

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

A API de Acesso Seguro fornece uma interface REST padrão e suporta o Fluxo de Credenciais de Cliente OAuth 2.0. Para começar, entre no Secure Access e crie suas chaves de API do Secure Access. Em seguida, use suas credenciais de API para gerar um token de acesso à API.



> Observação: chaves de API, senhas, segredos e tokens permitem acesso aos seus dados privados. Você nunca deve compartilhar suas credenciais com outro usuário ou outra organização.

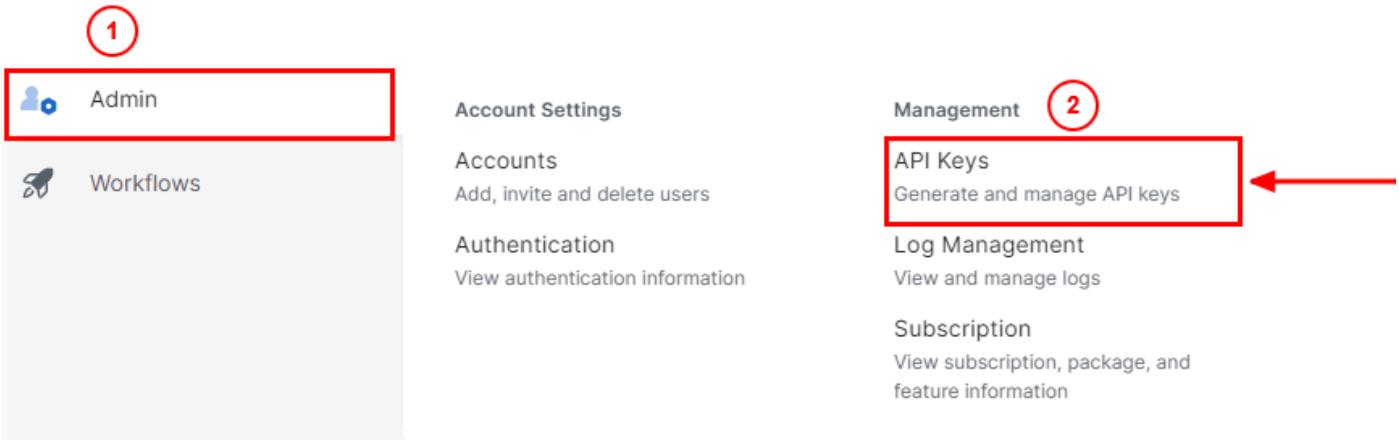
Configure a chave de API no Secure Access Dashboard antes de executar os scripts mencionados neste artigo.

Criar uma chave de API

Crie uma chave de API e um segredo com essas etapas. Entre no Secure Access com o URL: [Secure Access]

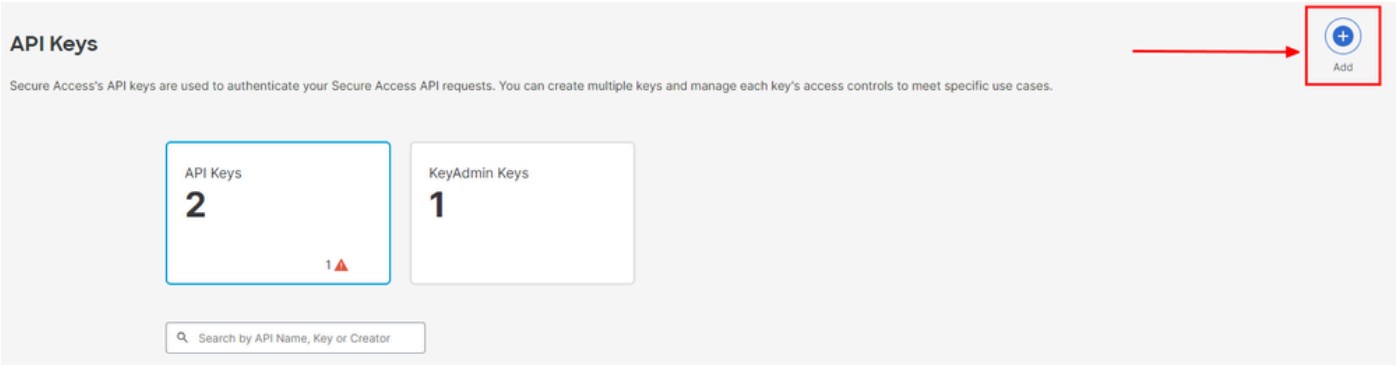
1. Na barra lateral esquerda, selecione a opção Admin.

   - Em Admin, selecione a opção **API Keys**:



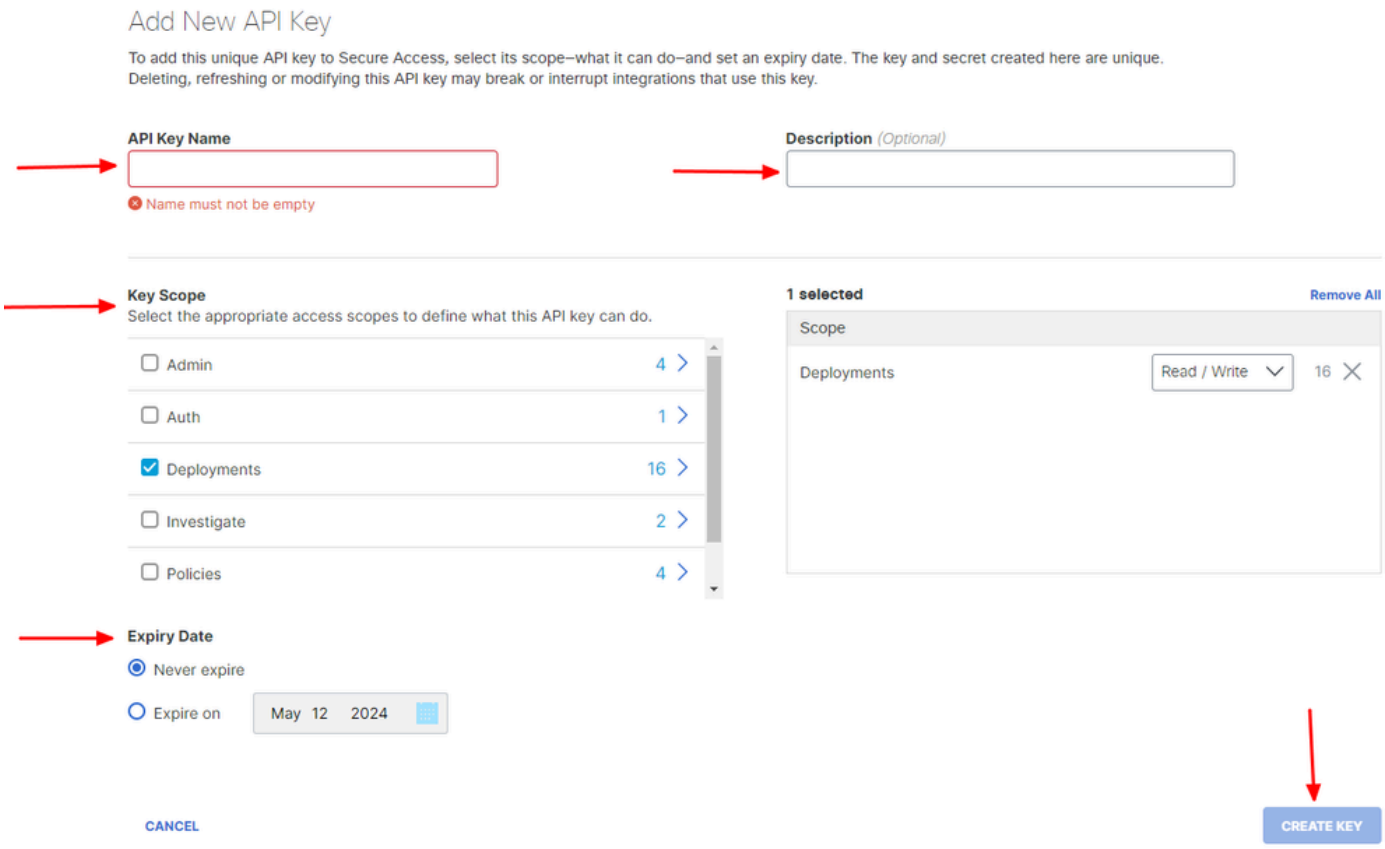
Administrador do painel de controle de acesso seguro - Chaves de API

3. No canto superior direito, clique no + botão Adicionar uma nova chave de API:



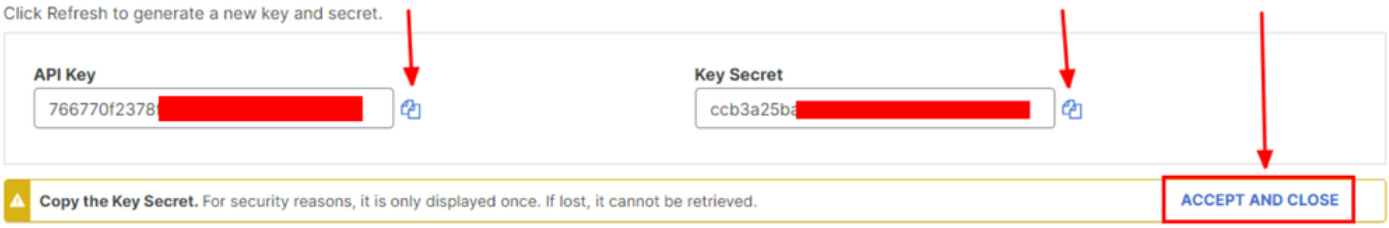
Acesso seguro - Adicionar chave de API

4. Forneça o **API Key Name, Description**(Opcional) e selecione o Key scope e Expiry date conforme sua necessidade. Ao terminar, clique no botão **Create**:

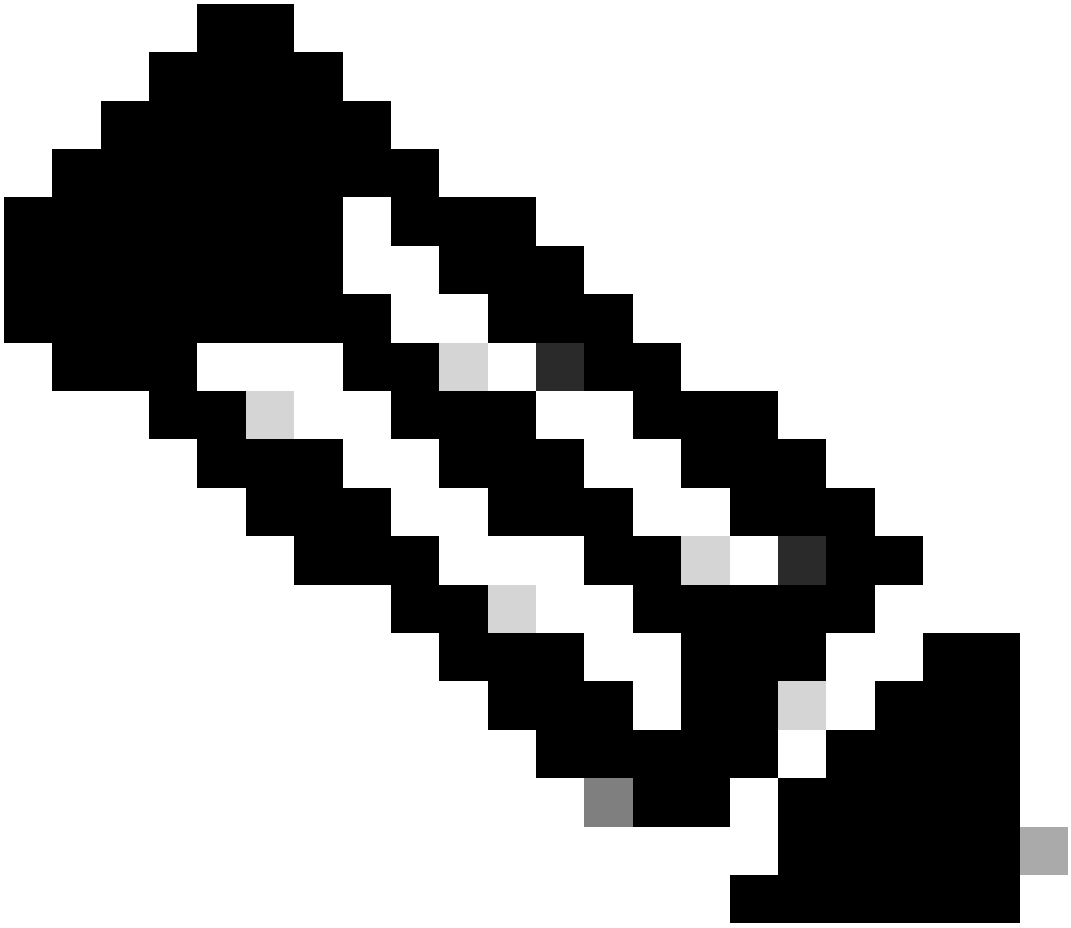


Acesso seguro - Detalhes da chave de API

5. Copie o API Keye o **Key Secret** e clique em ACCEPT AND CLOSE:



Acesso seguro - chave e segredo de API



Observação: há apenas uma oportunidade para copiar o segredo da API. O Secure Access não salva o segredo da API e você não pode recuperá-lo após sua criação inicial.

Código Python

Há várias maneiras de gravar esse código considerando que o token gerado é válido por 3600 segundos (1 hora). Você pode criar 2 scripts separados nos quais o primeiro script pode ser usado para gerar o Token de Portador e, em seguida, um segundo script no qual o Token de Portador pode ser usado para fazer a chamada de API (busca/atualização ou exclusão) para o recurso no qual você está interessado, ou escrever um único script para realizar ambas as ações, garantindo que, se um token de portador já tiver sido gerado, uma condição será mantida no código de que um novo token de portador não será gerado toda vez que o script for executado.

Para fazê-lo funcionar em Python, certifique-se de instalar estas bibliotecas:

```
pip install oauthlib pip install requests_oauthlib
```

Script 1:

Certifique-se de mencionar o client_ide correto client_secretneste script:

```
import requests from oauthlib.oauth2 import BackendApplicationClient from oauthlib.oauth2 import TokenE
```

Saída:

A saída deste script deve ser semelhante a:

```
Token: {'token_type': 'bearer', 'access_token': 'eyJhbGciOiJSUzI1NiIsImtpZCI6IjcyNmI5MGUzLWxxxxxxxxxxxxx
```

O access_tokené muito longo com milhares de caracteres e, portanto, para manter a saída legível, ele foi encurtado apenas para este exemplo.

Script 2:

O access_token do Script 1 pode ser usado nesse script para fazer chamadas à API. Como exemplo, use o Script 2 para buscar as informações sobre os grupos de túneis de rede usando o recurso /deployments/v2/networktunnelgroups:

```
import requests import pprint import json url = "https://api.sse.cisco.com/deployments/v2/networktunnel
```

Saída:

A saída deste script deve ser semelhante a:

```
{'data': [{'createdAt': '2023-11-01T10:17:09Z',
           'deviceType': 'ASA',
           'hubs': [{'authId': '[redacted]-sse.cisco.com',
                     'createdAt': '2023-11-01T10:17:09Z',
                     'datacenter': {'name': '[redacted]'},
                     'id': [redacted],
                     'isPrimary': True,
                     'modifiedAt': '2023-11-01T10:17:09Z',
                     'status': None,
                     'tunnelsStatus': None},
                    {'authId': '[redacted]-sse.cisco.com',
                     'createdAt': '2023-11-01T10:17:09Z',
                     'datacenter': {'name': '[redacted]'},
                     'id': [redacted],
                     'isPrimary': False,
                     'modifiedAt': '2023-11-01T10:17:09Z',
                     'status': None,
                     'tunnelsStatus': None}],
           'id': [redacted],
           'modifiedAt': '2024-02-12T03:09:14Z',
           'name': 'DMZ ASA Tunnel NC',
           'organizationId': [redacted],
           'region': '[redacted]',
           'routing': {'data': {'networkCIDRs': ['[redacted]']},
                       'type': 'static'},
           'status': 'connected'}],
 'limit': 10,
 'offset': 0,
 'total': 1}
```


Saída em Python - Grupos de túnel de rede

Você também pode buscar informações sobre Diretivas, Computadores em Roaming, Relatórios etc. com o Guia do Usuário de Desenvolvedores do Secure Access.

Troubleshooting

Os pontos de extremidade da API Secure Access usam códigos de resposta HTTP para indicar o sucesso ou a falha de uma solicitação da API. Em geral, os códigos no intervalo 2xx indicam êxito, os códigos no intervalo 4xx indicam um erro resultante das informações fornecidas e os códigos no intervalo 5xx indicam erros de servidor. A abordagem para resolver o problema dependeria do código de resposta recebido:

200	**OK**	Success. Everything worked as expected.
201	**Created**	New resource created.
202	**Accepted**	Success. Action is queued.
204	**No Content**	Success. Response with no message body.
400	**Bad Request**	Likely missing a required parameter or malformed JSON. The syntax of your query may need to be revised. Check for any spaces preceding, trailing, or in the domain name of the domain you are trying to query.
401	**Unauthorized**	The authorization header is missing or the key and secret pair is invalid. Ensure your API token is valid.
403	**Forbidden**	The client is unauthorized to access the content.
404	**Not Found**	The requested resource doesn't exist. Check the syntax of your query or ensure the IP and domain are valid.
409	**Conflict**	The client requests that the server create the resource, but the resource already exists in the collection.
429	**Exceeded Limit**	Too many requests received in a given amount of time. You may have exceeded the rate limits for your organization or package.
413	**Content Too Large**	The request payload is larger than the limits defined by the server.

API REST - Códigos de resposta 1

500	**Internal Server Error**	Something wrong with the server.
503	**Service Unavailable**	Server is unable to complete request.

API REST - Códigos de resposta 2

Informações Relacionadas

- Guia do usuário do Cisco Secure Access
- Suporte técnico e downloads da Cisco
- Adicionar chaves de API de acesso seguro
- Guia do usuário para desenvolvedores

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.