API クロスサイト リクエスト フォージェリの防止

（注） 簡素化と一貫性を実現するために、Cisco SD-WAN ソリューションは Cisco Catalyst SD-WAN としてブランド名が変更されました。さらに、Cisco IOS XE SD-WAN リリース 17.12.1a および Cisco Catalyst SD-WAN リリース 20.12.1 以降、次のコンポーネントの変更が適用されます。**Cisco vManage** から **Cisco Catalyst SD-WAN Manager** への変更、**Cisco vAnalytics** から **Cisco Catalyst SD-WAN Analytics** への変更、**Cisco vBond** から **Cisco Catalyst SD-WAN Validator** への変更、**Cisco vSmart** から **Cisco Catalyst SD-WAN コントローラ**への変更、および **Cisco コントローラ**から **Cisco Catalyst SD-WAN 制御コンポーネント**への変更。すべてのコンポーネントブランド名変更の包括的なリストについては、最新のリリースノートを参照してください。新しい名前への移行時は、ソフトウェア製品のユーザーインターフェイス更新への段階的なアプローチにより、一連のドキュメントにある程度の不一致が含まれる可能性があります。

表 *1* : 機能の履歴

機能名	リリース情報	説明
API クロスサイト リクエスト フォージェリの防止	Cisco IOS XE Catalyst SD-WAN リリース 16.12.1b Cisco Catalyst SD-WAN リリース 19.2.1	この機能により、Cisco Catalyst SD-WAN REST API の使用時に発生するクロスサイト リクエスト フォージェリ（CSRF）に対する保護が追加されます。この保護は、API リクエストに CSRF トークンを含めることによって提供されます。リクエストを許可リストに含めて、必要に応じて保護を不要にできます。

Cisco Catalyst SD-WAN REST API トークンベース認証

Cisco Catalyst SD-WAN リリース 19.2 では、Cisco Catalyst SD-WAN REST API を使用する場合にトークンベースの認証が提供されます。この保護は、トークンを API リクエストに含めるよう要求することによって提供されます。各 API セッションは、セッション全体で有効な一意のトークンを使用します。API リクエストにこのトークンが含まれていない場合、エンドポイントが許可リストに含まれていない限り、Cisco SD-WAN Manager はリクエストを拒否します（エンドポイントを許可リストに追加する方法に関するお問い合わせは、Cisco TAC またはエスカレーション サポート チームでケースを開いてください）。

（注） ただし、許可リストに含まれていない Cisco SD-WAN Manager の一部の GET API およびすべての POST API では、クロスサイト リクエスト フォージェリ（CSRF）トークン認証が必要です。

トークンの使用

次のセクションでは、API ドキュメントまたはサードパーティ アプリケーションを使用するときに、トークンが API でどのように使用されるかについて説明します。

API ドキュメント

Cisco SD-WAN Manager はトークンを自動的に生成し、[Cisco SD-WAN Manager API Docs] ページから送信するすべてのリクエストにトークンを追加します。このプロセスではユーザーのアクションは不要です。また、[API Docs] ページの操作方法は、以前のリリースと同じです。

このトークンベースの認証から除外する API リクエストがある場合は、Cisco TAC またはエスカレーション サポート チームにケースをオープンして、それらの API エンドポイントを許可リストに含めるように要求できます。

サードパーティ製アプリケーションのユーザー

Cisco SD-WAN Manager API リクエストにスクリプトまたはサードパーティ アプリケーション（Postman、LiveAction、SolarWinds、SevOne など）を使用する場合は、API が許可リストに含まれていないかぎり、各リクエストにトークンを含める必要があります。API リクエストにトークンが含まれておらず、許可リストにも含まれていない場合、Cisco SD-WAN Manager は、リクエストを拒否し、応答コード 403（禁止）と「SessionTokenFilter: Token provided via HTTP Header does not match the token generated by the server.」というメッセージを返します。

特定の API エンドポイントを許可リストに含めるように要求するには、Cisco TAC またはエスカレーション サポート チームとのケースをオープンします。

サードパーティ API リクエストにトークンを含めるには、次の手順を実行します。

方法 1

最初の方法では、作成するセッションが cookies.txt ファイルに保存されます。ファイルに含まれる jsessionid を使用して、以降のすべてのリクエストに同じセッションを使用できます。これは推奨される方法です。

1. Cisco SD-WAN Manager にログインするには、次のコマンド例を使用し、目的の IP アドレスに従って URL を変更します。

   ```
   sampleuser$ TOKEN=$(curl "https://209.165.200.254/dataservice/client/token" -X GET
   -b cookies.txt -s  -insecure)
   ```

   ログインを確認するには、cookies.txt ファイルを参照してください。

2. Cisco SD-WAN Manager にログインした後、リクエストを送信してトークンを取得します。ここで、vManage_IP は、Cisco SD-WAN Manager サーバーの IP アドレスです。トークンは、文字列形式または JSON 形式で取得できます。

   文字列形式でトークンを取得するには、次の URL を使用します。

   https://*vManage_IP*/dataservice/client/token

   JSON 形式でトークンを取得するには（Cisco IOS XE SD-WAN リリース 16.12 および Cisco SD-WAN リリース 19.2 以降）、次の URL を使用します。

   https://*vManage_IP*/dataservice/client/token?json=true

   これらのコールが返すトークンは、現在のセッションの残りの期間有効です。次の例は、トークンを取得するためのリクエストを示しています。

   文字列形式でトークンを取得するコマンド：

   ```
   sampleuser$ TOKEN=$(curl "https://vManage_IP/dataservice/client/token" -X GET -b
   cookies.txt -s  -insecure)
   ```

   文字列形式の出力：

   ```
   FC5B19BB3521EE20CFBDCD3CEDCC48F50CB1095C9654407936029E9C0EF99FEAE50440B60E49F7CD4A0BAB5307C2855F2E0C
   ```

   JSON 形式でトークンを取得するためのコマンド：

   ```
   TOKEN=$(curl "https://vManage_IP/dataservice/client/token?json=true" -X GET -b
   cookies.txt -s  -insecure)
   ```

   JSON 形式の出力：

   ```
   sampleuser$ echo $TOKEN

   {"token":"56CF324A8F67993B6FCCF57302068B0756DA8703BE712EFA18D4D9055B11312843F9D30B48A3902320FFAA8659AD01202A63"}
   ```

（注） curl コマンドでは JSON 形式はサポートされていません。

3. 現在のセッションにおける後続の各 API リクエストのヘッダーに、生成したトークンで構成される値を使用した X-XSRF-TOKEN キーを含めます。

   次の例は、生成されたトークンがヘッダーに含まれている GET リクエストと POST リクエストを示しています。

   コマンド：

```
sampleuser$ curl "https://vManage_IP/dataservice/server/info" -b cookies.txt -silent
 -insecure -H "X-XSRF-TOKEN: $TOKEN"
```

   出力：

```
{"Achitecture":"amd64","Available processors":2}
```

   コマンド

```
sampleuser$ curl
"https://vManage_IP/dataservice/settings/configuration/emailNotificationSettings" -X
 POST -b cookies.txt -silent  -insecure -H "X-XSRF-TOKEN: $TOKEN" -d
'{"enabled":true,"from_address":"test@mydomain.com","protocol":"smtp","smtp_server":"a.com",

"smtp_port":25,"reply_to_address":"test@test.com","notification_use_smtp_authentication":false}='
```

   出力：

```
{"data":[{"enabled":true,"notification_use_email_setting_authentication":false,"notification_use_smtp_authentication":false}]}
```

4. Cisco SD-WAN リリース 19.2.1 以降では、メモリリークを防ぐために、トークンを含む各 API コールの後にログアウトする必要があります。

   次の例は、ログアウトする方法を示しています。

   コマンド：

```
sampleuser$ curl "https://vManage_IP/logout" -b cookies.txt  -insecure -H
"X-XSRF-TOKEN:$TOKEN"
```

   出力：

```
Replaced cookie JSESSIONID="DcOke5mqix_15qCpWA1blIJVAMnVg3lDMU4ABRgVinvalid" for
domain 209.165.200.254, path /, expire 0
      < set-cookie: JSESSIONID=DcOke5mqix_15qCpWA1blIJVAMnVg3lDMU4ABRgVinvalid
```

（注） セッションからログアウトしたことを確認するには、jsessionid をチェックし、それが「invalid」で終わっていることを確認します。

方法 2

2 つ目の方法では、作成するセッションは保存されず、リクエストごとに新しいセッションを作成する必要があります。

1. Cisco SD-WAN Manager にログインした後、リクエストを送信してトークンを取得します。ここで、vManage_IP は、Cisco SD-WAN Manager サーバーの IP アドレスです。トークンは、文字列形式または JSON 形式で取得できます。

   文字列形式でトークンを取得するには、次の URL を使用します。

   https://*vManage_IP*/dataservice/client/token

   JSON 形式でトークンを取得するには（Cisco IOS XE SD-WAN リリース 16.12 および Cisco SD-WAN リリース 19.2 以降）、次の URL を使用します。

   https://*vManage_IP*/dataservice/client/token?json=true

   これらのコールが返すトークンは、現在のセッションの残りの期間有効です。次の例は、トークンを取得するためのリクエストを示しています。

   文字列形式でトークンを取得するコマンド：

   ```
   sampleuser$ curl --user admin:admin https://vManage_IP/dataservice/client/token
   --insecure
   ```

   文字列形式の出力：

   ```
   FC5B19BB3521EE20CFBDCD3CEDCC48F50CB1095C9654407936029E9C0EF99FEAE50440B60E49F7CD4A0BAB5307C2855F2E0C
   ```

   JSON 形式でトークンを取得するためのコマンド：

   ```
   sampleuser$ curl --user admin:admin
   https://vManage_IP/dataservice/client/token?json=true --insecure
   {"token":"F1E047E444DB2CA4237B0246DFE133345584B788C6E8776F04749A371B73F3C0C683043F1CDBB5E01BBBDA7D6C35F58EA37A"}
   ```

   JSON 形式の出力：

   ```
   FC5B19BB3521EE20CFBDCD3CEDCC48F50CB1095C9654407936029E9C0EF99FEAE50440B60E49F7CD4A0BAB5307C2855F2E0C
   ```

2. 現在のセッションにおける後続の各 API リクエストのヘッダーに、生成したトークンで構成される値を使用した X-XSRF-TOKEN キーを含めます。

   次の例は、生成されたトークンがヘッダーに含まれている GET リクエストと POST リクエストを示しています。

   コマンド：

   ```
   sampleuser$ curl "https://vManage_IP/dataservice/server/info" -H "Cookie:
   JSESSIONID=pSwrx3AEWokiDO1TkFiOjgSehp-ITNdFn7Xj9PsL.c331d01e-91d7-41cc-ab90-b629c2ae6d97"
   --insecure -H "X-XSRF-TOKEN=
   FC5B19BB3521EE20CFBDCD3CEDCC48F50CB1095C9654407936029E9C0EF99FEAE50440B60E49F7CD4A0BAB5307C2855F2E0C"
   ```

   出力：

   ```
   {"Achitecture":"amd64","Available processors":2}
   ```

   コマンド

   ```
   sampleuser$
   "https://vManage_IP/dataservice/settings/configuration/emailNotificationSettings" -H
    "Cookie:
   JSESSIONID=pSwrx3AEWokiDO1TkFiOjgSehp-ITNdFn7Xj9PsL.c331d01e-91d7-41cc-ab90-b629c2ae6d97"
   --insecure -H "X-XSRF-TOKEN=
   FC5B19BB3521EE20CFBDCD3CEDCC48F50CB1095C9654407936029E9C0EF99FEAE50440B60E49F7CD4A0BAB5307C2855F2E0C"
   -X POST --insecure -d
   '{"enabled":true,"from_address":"test@mydomain.com","protocol":"smtp","smtp_server":"a.com",
   "smtp_port":25,"reply_to_address":"test@test.com","notification_use_smtp_authentication":false}='
   ```

出力：

```
{"data":[{"enabled":true,"protocol":"smtp","smtp_server":"a.com","from_address":"test@mydomain.com",
"smtp_port":25,"notification_use_smtp_authentication":false,"reply_to_address":"test@test.com"}]}
```

3. Cisco SD-WAN リリース 19.2.1 以降では、メモリリークを防ぐために、トークンを含む各 API コールの後にログアウトする必要があります。

   次の例は、ログアウトする方法を示しています。

   コマンド：

```
sampleuser$ curl "https://vManage_IP/logout" -b cookies.txt  --insecure -H
"X-XSRF-TOKEN:$TOKEN"
```

   出力：

```
Replaced cookie JSESSIONID="DcOke5mqix_15qCpWA1blIJVAMnVg3lDMU4ABRgVinvalid" for
domain 209.165.200.254, path /, expire 0
< set-cookie: JSESSIONID=DcOke5mqix_15qCpWA1blIJVAMnVg3lDMU4ABRgVinvalid
```

（注） セッションからログアウトしたことを確認するには、jsessionid をチェックし、それが「invalid」で終わっていることを確認します。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。