## vManageでAPIを実行するためのPostmanの設定

## 内容

概要
 システム要件
 背景説明
 APIを実行するためのPostmanの設定
 ステップ1: Postmanを開き、新しいHTTP要求を作成します。
 ステップ2: vManageに対するユーザ名とパスワードのクレデンシャルで認証します。
 ステップ3: トークンを要求する
 ステップ4: vManageに対する別のAPIの実行に進みます。
 ステップ5: セッションを閉じる
 自動化された環境でのAPI呼び出しの実行
 変数にトークンを保存する方法
 新しいセッションのSESSIONID cookieをクリアする方法
 Collection Runnerの使用方法

### 概要

このドキュメントでは、Postmanを使用してApplication Programming Interface(API)を実行する方 法について説明します。

## システム要件

Postmanのインストール

•vManageおよびユーザ名とパスワードのクレデンシャルへのアクセス

**注**: Postmanがない場合は、<u>https://www.postman.com/downloads/</u>からダウンロードしてく ださい。

## 背景説明

主に使用される、または最も一般的に使用されるHTTP動詞(または適切に呼び出されるメソッド)は、POST、GET、PUT、PATCH、およびDELETEです。

これらは、それぞれ作成、読み取り、更新、および削除(またはCRUD)操作に対応します。

他にも多くの動詞がありますが、あまり使われていません。これらの頻度の低い方法のうち、 OPTIONSおよびHEADは他の方法よりも頻繁に使用されます。

## APIを実行するためのPostmanの設定

ステップ1: Postmanを開き、新しいHTTP要求を作成します。

強調表示されたオプションのいずれかをクリックすると、新しいHTTP要求を作成できます。



新しいHTTP要求を作成します。

# ステップ 2:vManageに対するユーザ名とパスワードのクレデンシャルで認証します。

別のHTTP要求を作成します。

- 1. HTTP動詞としてPOSTを選択します。
- 2. POSTの横に<u>https://<vmanage-ip>/j\_security\_checknext</u>を追加します。
- 3. Bodyをクリックし、KEYパラメータj\_usernameとj\_passwordとそれぞれの値を追加しま す。
- 4. [Send] をクリックします。
- 注:この例では、vManage ipアドレスは10.88.244.30で、ポートは5301です

注: ユーザ名とパスワードの値には、adminを使用します。

Postmanのパラメータを完全に設定します。

		Postma	n		
🥖 Hom	ne Workspaces × API Network ×	Reports Explore Q	Search Postman	🐥 Invite 🖑 🚳 🗘	😧 Upgrade 🗸
A My Wor	kspace New Import	POST https://10.88.244.30:! • + •••		No Environmen	nt v ©
Collections	+ =	https://10.88.244.30:5301/j_security_check		) Save	~ 🖉 🗉 🛷
00 APIs	<ul> <li>✓ Cisco-Reservable-SD-WAN ★</li> <li>✓ □ 1.Authentication</li> </ul>	POST 1 https://10.88.244.30:5301/j_security	_check	24	Send 🗸
Environments	OST Authentication     OET Token     C 2. SD-WAN Fabric Devices	Params Authorization Headers (9) Body • none form-data • x-www-form-urlencoded	Pre-request Script Tests Settings raw binary GraphQL	3	Cookies
	GET Fabric Devices	KEY	VALUE	DESCRIPTION	••• Bulk Edit
Mock Servers	GET Devices Status	j_username	admin		
44	GET Device Counters	j_password	admin		
Monitors	GET Interface statistics	Key	Value	Description	
010	<ul> <li>3. SD-WAN Device Template</li> </ul>				
Flows	GET Template Feature				
-	GET Template Feature Type				
4) History	GET Device Templates				
	4. SD-WAN Device Policy				
	GET vEdge Template Policy	Body Cookies (1) Headers (10) Test Results	Status: 20	0 OK Time: 1153 ms Size: 415 B	Save Response V
	GET Policy List				
	✓  ☐ 5. Tenant list	Pretty Raw Preview Visualize Text	✓ ==		r q
	GET Tenant-list	1		Hesponse	
	<b>GET</b> system statistics	L			

注意:このAPIコールの応答は空である必要があります

### ステップ3:トークンの要求

- 1. HTTP動詞として[GET]を選択します。
- 2. GET https://<vmanage-ip>/datservice/client/tokenの横にAPIコールの詳細を追加します。
- 3. [Send] をクリックします

**注**:vManageバージョン19.2.1以降では、正常にログインしたユーザがAPIコールを介して 各POST/PUT/DELETE操作に対してX-XSRG-TOKENまたはCSRFトークンを送信する必要 があることが必須になっています。

API呼び出しが実行されると、応答ストリングが本文に表示されます。その文字列を保存します 。次の図は、In Postmanの出力を示しています。

vManage認証。

•••		P	ostman		
🥖 Hom	ne Workspaces ~ API Network ~	Reports Explore	Q Search Postman	💊 🐥 Invite 🗞 🕸 🗘	Opgrade Upgrade
A My Wor	kspace New Import	POST https://10.88.244.30:1 GET https://10.88.244.30	0:5:● + ***	No Environr	nent 🗸 🤇
Collections	+ = · · · · · · · · · · · · · · · · · ·	https://10.88.244.30:5301/dataservice/client/to/	sen .	🖺 Save	
APIs	<ul> <li>LAuthentication</li> <li>POST Authentication</li> <li>OET Token</li> </ul>	GET https://10.88.244.30:5301/dat Params Authorization Headers (7) Bod Query Params	aservice/client/token y Pre-request Script Tests Settings	2	3 Send V
	<ul> <li>E 2. SD-WAN Fabric Devices</li> <li>GET Fabric Devices</li> </ul>	KEY	VALUE	DESCRIPTION	••• Bulk Edit
Monitors eta	CET Device Status     OFT Device Counters     OFT Interface statistics     CET Interface Statistics     CET Template Feature			and the second se	
4) History	CET Template Feature Type     CET Device Templates     C				
	<ul> <li>vEdge Template Policy</li> <li>Policy List</li> <li>5. Tenant list</li> </ul>	Body         Cookies (1)         Headers (13)         Test Results           Pretty         Raw         Preview         Visualize	£€ ק= ∨ nozu	Status: 200 OK Time: 837 ms Size: 517 B Response	Save Response V
	GET Tenant-list GET system statistics GET Packet capture	1 9DE4EB3A230ADF4A6A802885BFD206A2A	62038E43B93D498C1F67798B915CDD6701FBF8CF	98C419F9FA1C7007F9110DE2328	

vManageのトークンを要求する

警告:図に示すようにトークンを取得しなかった場合は、この手順を繰り返してください。

#### ステップ4:vManageに対する別のAPIの実行に進みます。

この例では、POST要求が必要です

1. 実行するAPIコールを選択します。この例では、 <u>https:///dataservice/statistics/dpi/aggregation</u>です。

**ヒント**:他のAPIコールを使用する場合は、vManage url <u>https://vmanage-ip:port/apidocs</u>に アクセスしてください。

2. APIコール本文を収集します。

注意:このAPIコールにはJSON形式の本文が含まれています

		Postman	
🥖 Hom	e Workspaces v API Network v I	Reports         Explore         Q         Search Postman         Q         A Invite         R <sup>®</sup> Q         Q         O	Upgrade ~
A My Wor	kspace New Import	POST https://10.88.244.30:1      GET https://10.88.244.30:5:     POST https://10.88.244.30:1      H      No Environment	~ ©
Collections	+ = •••	https://10.88.244.30:5301/dataservice/statistics/dpi/aggregation	
oo APis	✓ ☐ 1.Authentication	POST v https://10.88.244.30:5301/dataservice/statistics/dpl/aggregation	Send 🗸 🖗
Environments	CET Token     CET 2. SD-WAN Fabric Devices	Params     Authorization     Headers (9)     Body •     Pre-request Script     Tests     Settings       • none     • form-data     • x-www-form-urlencoded     • raw     • binary     • GraphQL     JSON     ~	Cookies Beautify
Mock Servers	GET Fabric Devices GET Devices Status	<pre>1 { "query":{ "condition":"AND", "rules":[ { "value":[."2022-05-06T14:45:00","2022-05-06T14:49:59" ], "field":"entr "type":"date", "operator":"between".} ] }, "aggregation":{ "field":[ { "property":"host_name", "name":"host-na "sequence":1 }, { "property":"vdevice_name", "sequence":2 }, { "property":"local_color", "sequence":3 },</pre>	ry_time", 📲 ame",
Monitors	GET Device Counters	<pre>{'property':"remote_color", 'sequence":4 }, {'property":"application", "sequence":5 }, {'property':"local_sys "sequence":6 }, {'property":"remote_system_ip", "sequence":7 } ], "metrics":[ { "property":"octats", "type": "order":"desc".} { "property":"packets", "type:"sum", "order":"desc" } ], 'histogram": { "property": "entry</pre>	stem_ip", "sum", y_time",
o[0 Flows	GET Template Feature	"type": "minute", "interval": 5, "order": "asc" } }	T
4) History	GET Template Feature Type		
	<ul> <li>4. SD-WAN Device Policy</li> </ul>		

- 3. Headerをクリックし、文字列X-XSRF-TOKENを値としてKeyとして追加します。
- 4. 「送信」をクリッ**クします。**

次の図は、APIコールがどのように表示されるかを示しています。

•••			Postman		
🥖 Hom	ne Workspaces × API Network × I	Reports Explore	Q Search Postman	🌀 🐥 Invite 😪	ै 🕸 🗘 🥥 Upgrade 🗸
	kspace New Import	POST https://10.88.244.30:! • GET https://10	0.88.244.30:5: • POST https://10.88.244.30:! •	+ •••	No Environment v 💿
	+ = 000	https://10.88.244.30:5301/dataservice,	/statistics/dpi/aggregation		🖺 Save 🗸 🥖 🗐
00 APIs	✓ Cisco-Reservable-SD-WAN     ★     ✓	POST ~ https://10.88.244.30	0:5301/dataservice/statistics/dpi/aggregation		3 Send ~
Environments	CET Token	Params Authorization Headers (10 Headers $\odot$ 9 hidden	<ul> <li>Body          <ul> <li>Pre-request Script Tests</li> </ul> </li> </ul>	Settings	Cookies
e	GET Fabric Devices	KEY	VALUE	DESCRIPTION	∞∞ Bulk Edit Presets ∨
Mock Servers	GET Devices Status	X-XSRF-TOKEN	9DE4EB3A230ADF4A6A80	J2885BFD206A2A6203	
<b>A</b> A	GET Device Counters	Кеу	Value	Description	4
Monitors	GET Interface statistics				
010	<ul> <li>SD-WAN Device Template</li> </ul>				
Flows	GET Template Feature				
	OET Template Feature Type				
4) History	GET Device Templates				
ristory	✓  ☐ 4. SD-WAN Device Policy				

DPI集約API呼び出し。

## ステップ5:セッションを閉じる

vManageやデバイスから必要なすべての情報を取得したら、vManageのリソースを解放し、悪意 のあるユーザがセッションを使用する可能性を排除します。

## 自動化された環境でのAPI呼び出しの実行

後続のAPI呼び出しで使用するcookieと変数を保存する

#### 変数にトークンを保存する方法

後で再利用できるように、トークンを変数に保存します。

▲ OST C	GET G.X	GET G.	GET G.	GET D.	GET h	GET G.	GET G.	Þ	+	 SDWAN		•	\$
► Get n	ew token i	n 19.2 on m	y vManage							Examples 0 🔻	BUIL	LD !	<b>P</b>
GET	Ŧ	https://10.	48.21.230/da	taservice/cli	ent/token?j	son=true				Ser	id 🝷	Save	Ŧ
Params	<ul> <li>Auth</li> </ul>	orization	Headers (9)	Body	Pre-rec	quest Script	Tests 🔵	Sett	ings			Cookies	Code
1 2 3 4 5	<pre>var jsonl postman.s pm.test('     pm.expe });</pre>	Data = JSON SetEnvironm 'Status cod ect(pm.resp	<pre>I.parse(res mentVariab de is 200", conse.code)</pre>	sponseBody le("token" , () => { ).to.eql(2	); , jsonDat 00);	ta.token);				Test scripts are w run after the resp Learn more about SNIPPETS Get an environm Get a global varia	ritten in Jav ionse is reci t tests scrip ent variable ible	aScript, and eived. Its	are 🕨
										 Get a variable			

トークンを変数に保存する

#### トークンをJSON形式で要求する場合は、トークンを保存します。[Tests] タブを使用して、表示 されている行を貼り付けます。

var jsonData = JSON.parse(responseBody);
postman.setEnvironmentVariable("token", jsonData.token);

#### その後、API呼び出しはすべてトークン変数を使用できます。

<b>∢</b> GE	TG.	GET G.	GET G.	GET D.	GET h	GET G.	GET G.	GET G.	< ▶	+	••••	SDWAN	Ŧ	o	\$
⊧ Ge	et server	informa	tion 19.2 la	b vManage	with corr	ect token						Examples 1 🔻	BUILD	1	<b>P</b>
GET		₹ h	ttps://10.48	.21.230/data	aservice/ser	ver/info						Sen	d 💌	Save	Ŧ
Parar	ms Au	uthorizati	on Hea	ders (10)	Body	Pre-reques	st Script	Tests	Settir	ngs				Cookies	Code
~	Host 🗊					<calculated< th=""><th>when reque</th><th>st is sent&gt;</th><td></td><td></td><td></td><td></td><td></td><td></td><td></td></calculated<>	when reque	st is sent>							
~	User-Age	nt				PostmanRur	ntime/7.26.3	3							
~	Accept (					*/*	not 2020-08-	-24 at 10 03	36 ppg						
~	Accept-Er	ncoding				gzip, deflate	, br	24 01 10.00	.oo.prig						
~	Connecti	on 🔅				keep-alive									
~	X-XSRF-T	OKEN				{{token}}									
~	Content-	Туре				application/j	json								
	Key					Value					D	escription			

トークン変数を使用する

#### 新しいセッションのSESSIONID cookieをクリアする方法

API呼び出しを実行して外に出るときは常に、JSESSIONIDを使用します。

以前のリリースで使用したような基本認証は使用できません。代わりに、クレデンシャルを提供し、IDをCookieに保存するだけです。この前に、すべてのクッキーまたは特定のクッキーをクリ アするためにプレテストを使用できます。

< La	au	POST CX	GET G.	GET G.	GET G.	GET D.	GET h	GET G.	► + •	SDW/	N	Ŧ	•	۵
▶ Get J	SESSION	llD in new	ver release	(s)							Examples 0 🔻	BUILD		<b>P</b>
POST		+ http	s://10.48.21	.230/j_secu	rity_check						Send	•	Save	Ŧ
Params	Auth	horization	Header	rs (10)	Body 鱼	Pre-reques	st Script ●	Tests 🔵	Settings				Cookies	Code
1 2 3 4 5	<pre>const jar.cl // e });</pre>	jar = pm ear(pm.r error – <	n.cookies. request.ur Error>	jar(); l, functi	ion (error	) {					Pre-request scripts a and are run before th Learn more about pr SNIPPETS Get an environment of Get a global variable Get a variable Set an environment of Set a global variable	re written i e request e-request : variable	in JavaScrip Is sent. scripts	<sup>,,</sup> ⊧

Cookieのクリア

これは、Pre-requestスクリプトに記述されたコードを介して行われます。

#### Collection Runnerの使用方法

セッションを実行し、各セッションに固有のデータを保存できる環境が整ったので、Collection Runnerを使用して一連のコールを実行できます。

繰り返すイベントの順序を選択し、PostmanがAPI呼び出しを実行できるように繰り返し回数を選 択します。

Choose a collection or folder	RUN ORDER.	Deselect All	Select All	Reset
Q Search for a collection or folder	POST Get JSESSIONID in newer release(s)			
< Viptela	GET Get new token in 19.2 on my vManage			
Post https://10.48.21.230/apidocs Post https://10.48.21.230/dataservice/device Post https://10.48.21.230:443/dataservice/statistics/approute Post https://10.48.21.230:443/dataservice/sta	Get Get server info with in-correct token  Get Get server information 19.2 lab vManage with correct token  Post https://10.48.21.230/apidocs  Post https://10.48.21.230/dataservice/device  Post https://10.48.21.230.443/dataservice/statistics/approute			
Environment SDWAN +	Post https://10.48.21.230:443/dataservice/statistics/approute     Post https://10.48.21.230:443/dataservice/statistics/approute     est https://10.48.21.230:443/dataservice/statistics/approute			
Delay 0 ms	POST         https://10.48.21.230/dataservice/system/device           POST         https://10.48.21.230/dataservice/template/device/config/config			
Data     Select File       Save responses     Image: Comparison of the select	Post         https://10.48.21.230/dataservice/system/device/fileupload           Put         https://mtv5-sdwan-vman-1           Post         https://10.48.21.230/dataservice/system/device			
Run collection without using stored cookies Save cookies after collection run	POST       https://10.48.21.230/dataservice/system/device         GET       Get new token in 19.2         GET       inventory call			
Run Viptela	GET https://10.48.21.230/dataservice/alarms			

回収ランナー

コールの「ライブラリ」から、特定のフロー/順序を実行するために特定の順序でコールを配置し ます。

# 200 OKまたはその他の値を応答として取得するかどうかを結果チェックに入れ、合格または不合格として扱います。

► Get new token in 19.2 on my vManage Examples 0 - BUIL		<b>P</b>
GET • https://10.48.21.230/dataservice/client/token?json=true Send •	Save	*
Params       Authorization       Headers (9)       Body       Pre-request Script       Tests       Settings         1       var       jsonData       JSON.parse(responseBody);       Test scripts are written in Java         2       postman.setEnvironmentVariable("token", jsonData.token);       Test scripts are written in Java         3       pm.test("Status code is 200", () => {       Learn more about tests script         4       pm.expect(pm.response.code).to.eql(200);       SNIPPETS         5       });       Get an environment variable         Get a global variable       Set a global variable         Set a global variable       Set a global variable	Cookies Script, and a ved.	Code
Body       Cookies (1)       Headers (13)       Test Results       Status: 200 OK       Time: 67 ms       Size: 550 B       Sa         Pretty       Raw       Preview       Visualize       JSON       Time:       57 ms       Size: 550 B       Sa	ve Respor	Q
1 2 3 3 <i>応答コードの確認</i>	059"	

pm.test("Status code is 200", () => {
 pm.expect(pm.response.code).to.eql(200);
});

#### その後、私たちは私たちの実行で合格または失敗を見ることができます。

ollection Runne	er Run Results		My Workspace 🔻		Run In Command Line	Doc
20 PASSED	Viptela SDWAN			Run Summary   Export Results	Retry	New
Iteration 1						
POST G	Get JSESSIONID in newer release(s)	https://10.48.21.230/j_se	Viptela / Get JSESSIONID in newer			
St.	atus code is 200					
GET GE	et new token in 19.2 on my vManag	e https://10.48.21.230/dat	Viptela / Get new token in 19.2 on	200 OK 53 ms 550 B		
St.	atus code is 200					
GET Ge	et server info with in-correct token	https://10.48.21.230/dat	Viptela / Get server info with in-co	🔴 403 Forbidden 🔵 56 ms 🌑 583 B		
St.	atus code is 403					
GET GE	et server information 19.2 lab vMar	hage with correct token https://	/10.48.21.230/dat Viptela / Get server ir	nformation 1 200 OK • 45	9 ms 🛑 486 B 🔺	
St.	atus code is 200					
Iteration 2						
POST C	Get JSESSIONID in newer release(s)	https://10.48.21.230/j_se	Viptela / Get JSESSIONID in newer			
Sta	atus code is 200					
GET GE	et new token in 19.2 on my vManag	e https://10.48.21.230/dat	Viptela / Get new token in 19.2 on	200 OK 48 ms 550 B		
Sta	atus code is 200					
GET Ge	et server info with in-correct token	https://10.48.21.230/dat	Viptela / Get server info with in-co	403 Forbidden 49 ms 583 B		
	atus sada is 402					

自動実行

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人に よる翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっ ても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性につ いて法的責任を負いません。原典である英語版(リンクからアクセス可能)もあわせて参照する ことを推奨します。