

## Spécialité : Réseaux Intelligents et Cybersécurité

Présenté par : COULIBALY DAOUDA DAVID & WAHID Moncef

---

Lab. 1 — Exploration des API REST et accès RESTCONF à un IOS XE

---

Encadré par:

Pr. JIHANE JEBRANE

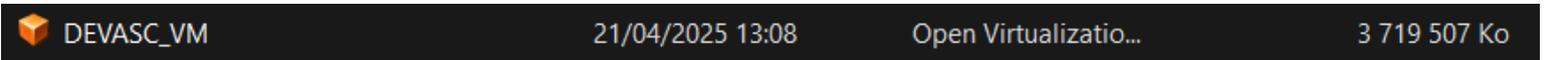
ENSA Khouribga

**Année universitaire :2024/2025**

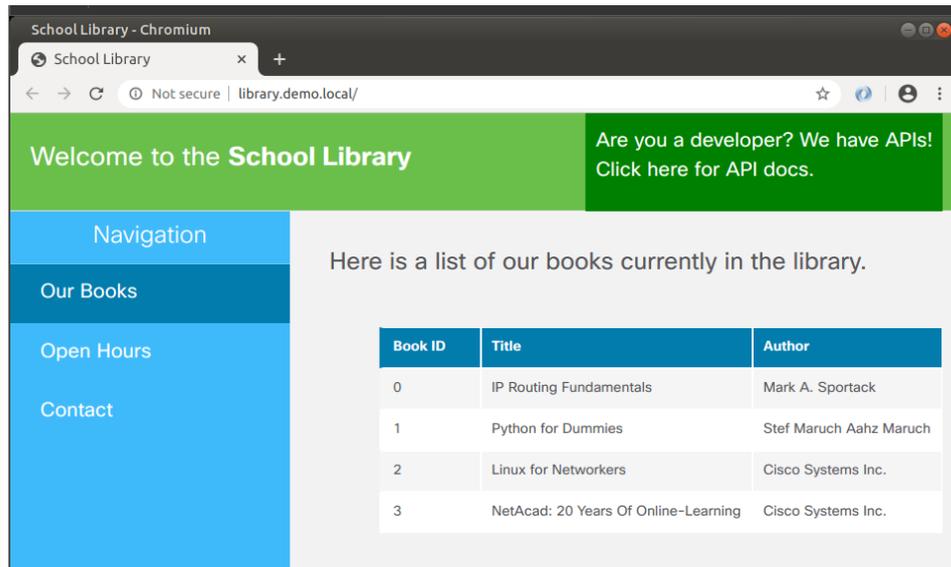
## Section 1: Interaction avec un simulateur d'API :

### Partie 1 : Préparation du Lab. Environnement :

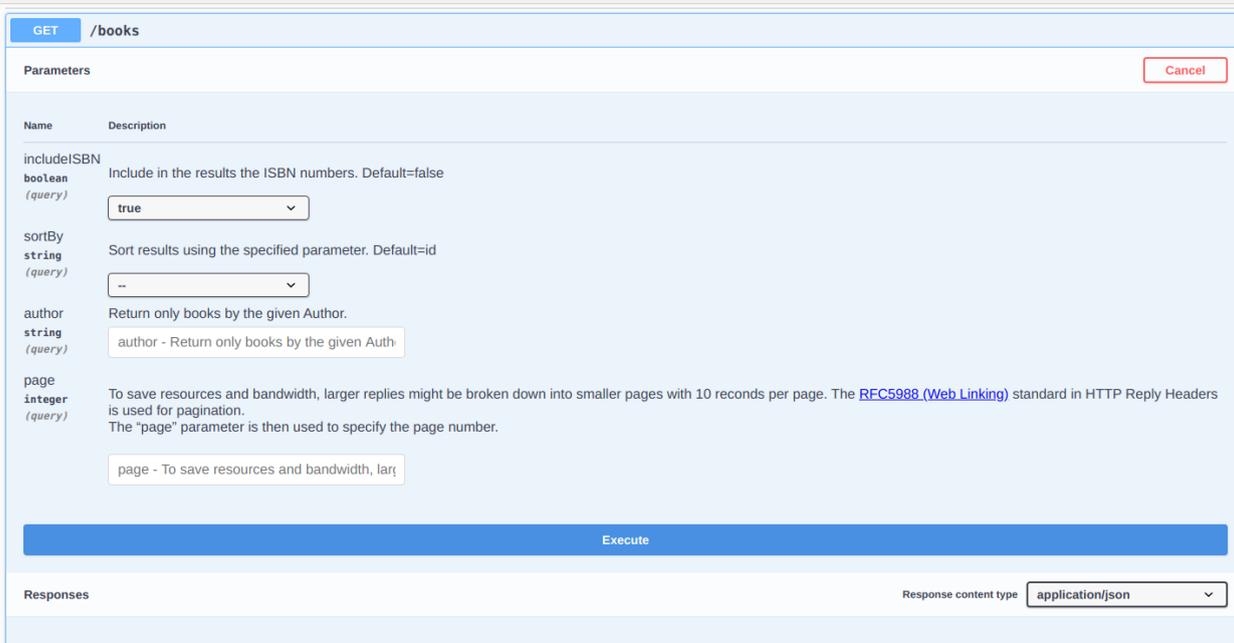
Après nous avons téléchargé l'image depuis cicso , nous la installons sur VMWARE :



### Partie 2: Interaction avec l'API School Library :



### Étape 1 : Lister les livres avec leur ISBN via l'API GET /books :



Résultats de requête avec isbn = true:

Execute Clear

Responses Response content type application/json

Curl

```
curl -X GET "http://library.demo.local/api/v1/books?includeISBN=true" -H "accept: application/json"
```

Request URL

```
http://library.demo.local/api/v1/books?includeISBN=true
```

Server response

Code Details

Étape 2 : Obtenir un jeton (token) via l'API POST /loginViaBasic:

School Library API

library.demo.local/api/v1/docs

GET /books/{id}

PUT /books/{id}

POST /loginViaBasic

Parameters

No parameters

Execute

LOADING

Responses Response content type application/json

Code	Description
200	Success

Name	Description	Type
Authorization: Basic	BASE64 encoded username:password	string

POST /loginViaJSON

Le jeton :

Request URL

```
http://library.demo.local/api/v1/loginViaBasic
```

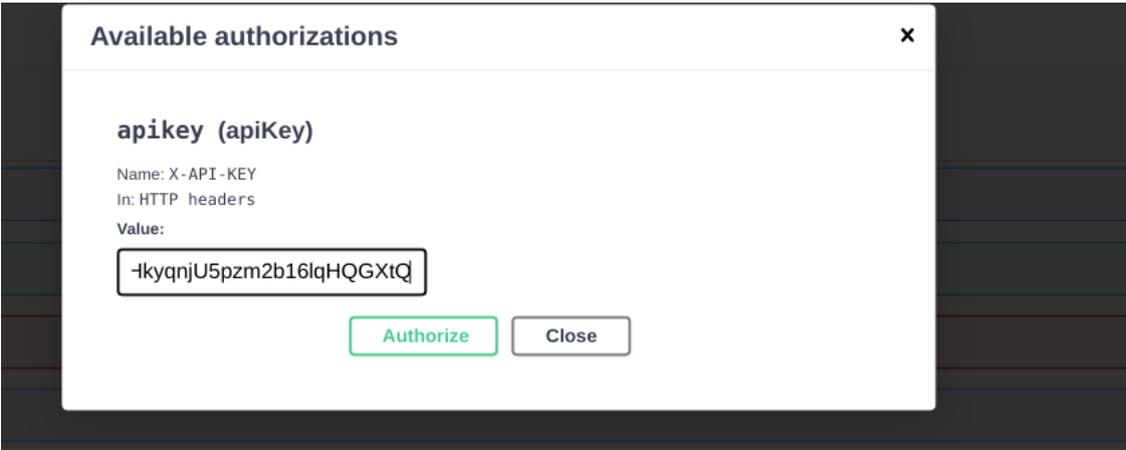
Server response

Code	Details
200	<p>Response body</p> <pre>{   "token": "cisco Efp8RgNH34jAKZLK6omLCwk5fNSKoL3SVsUM8CKL98w" }</pre> <p>Response headers</p> <pre>access-control-allow-origin: http://library.demo.local content-length: 69 content-type: application/json date: Mon, 21 Apr 2025 14:37:42 GMT server: Werkzeug/0.14.1 Python/3.8.2 vary: Origin</pre>

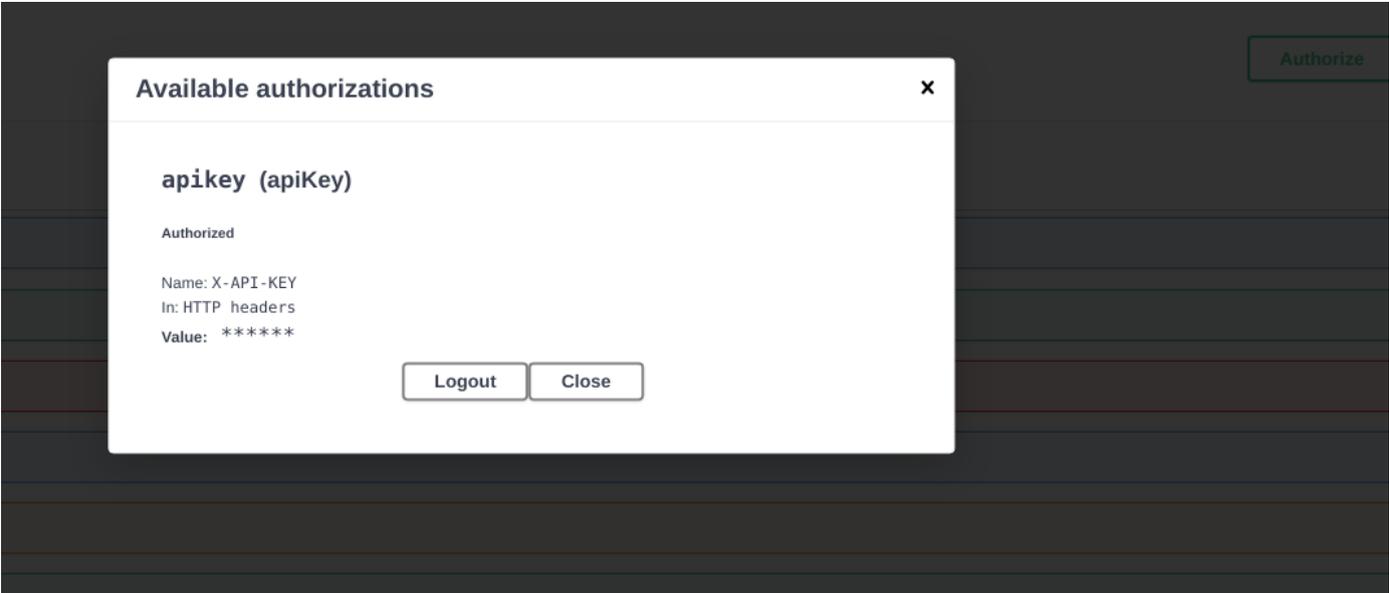
Responses

Code	Description
200	Success

Authorize pour que les API verrouillées sont devient accessibles.:

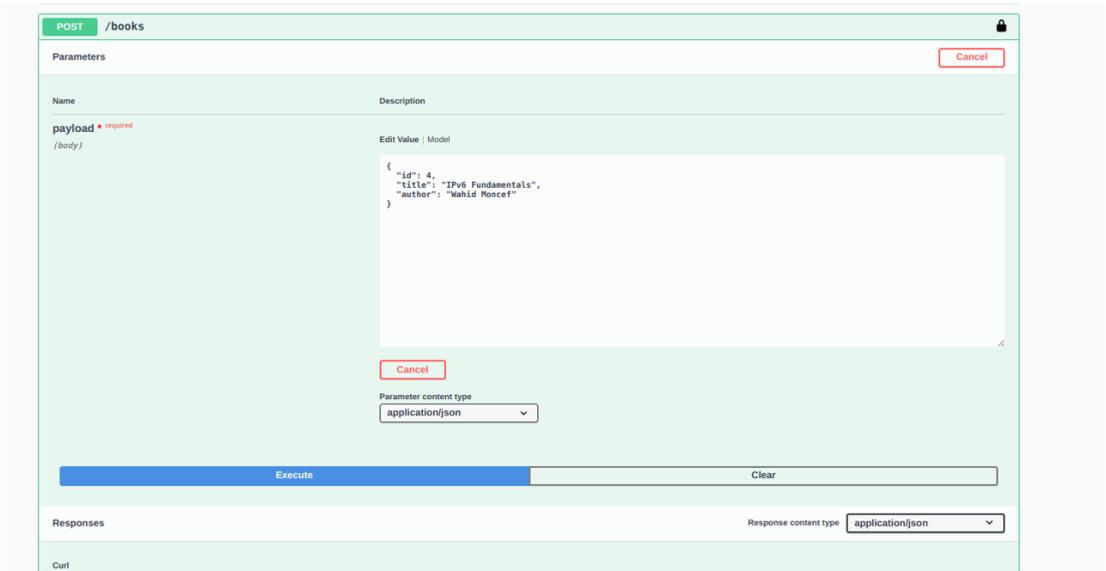


Résultats :



Étape 3 : Ajouter des livres via l'API POST /books:

Ajoutons un livre avec id 4 :



# Résultats :

Curl

```
curl -X POST "http://library.demo.local/api/v1/books" -H "accept: application/json" -H "X-API-KEY: cisco|fm_cM9Clef9VwneymqEEb73EwcWtV6fihXqmX_19wA0" -H "Content-Type: application/json" -d "{ \"id\": 4, \"title\": \"IPv6 Fundamentals\", \"author\": \"Wahid Moncef\"}"
```

Request URL

http://library.demo.local/api/v1/books

Server response

Code	Details
200	<p>Response body</p> <pre>{   "id": null,   "title": null,   "author": null }</pre> <p>Response headers</p> <pre>access-control-allow-origin: http://library.demo.local content-length: 59 content-type: application/json date: Mon, 21 Apr 2025 14:47:17 GMT server: Werkzeug/0.14.1 Python/3.8.2 vary: Origin</pre>

Responses

Code	Description
200	Success

Ajoutons un autre livre avec id 5 :

POST /books

Parameters

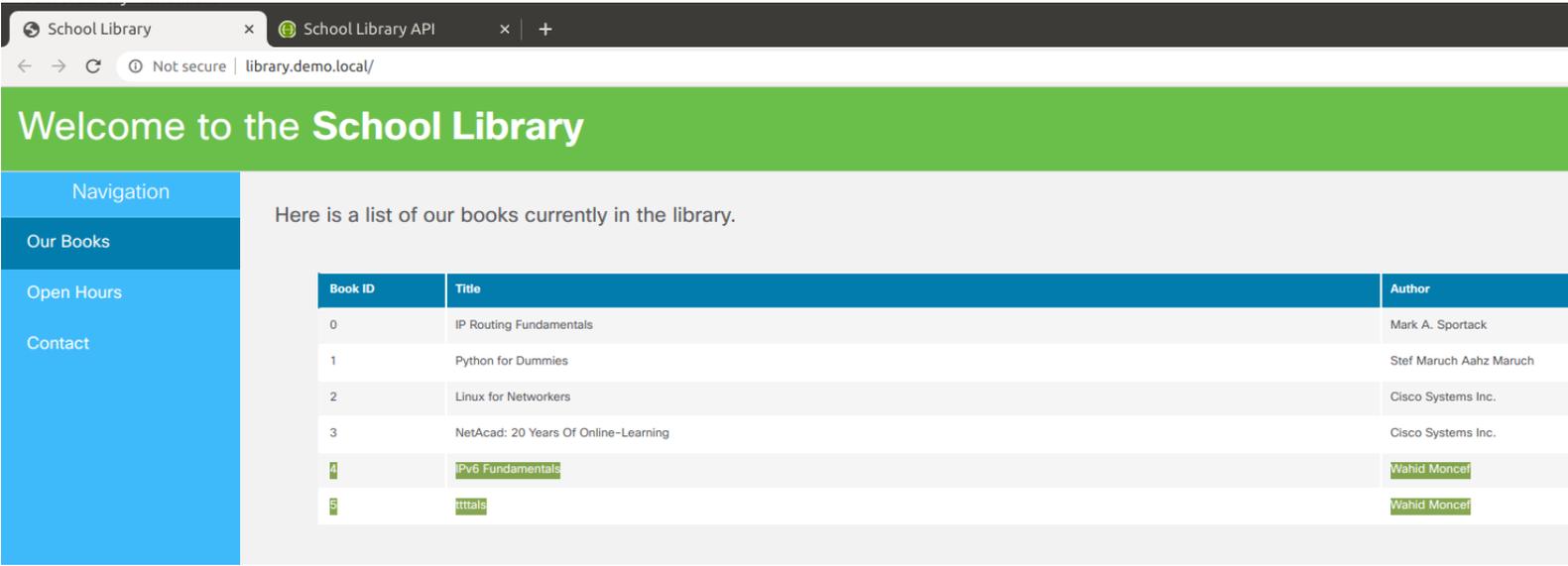
Name	Description
<b>payload</b> <span style="color: red;">* required</span> <i>(body)</i>	<p>Edit Value   Model</p> <pre>{   "id": 5,   "title": "ttttals",   "author": "Wahid Moncef" }</pre> <p>Cancel</p> <p>Parameter content type application/json</p> <p>Execute</p>

Responses

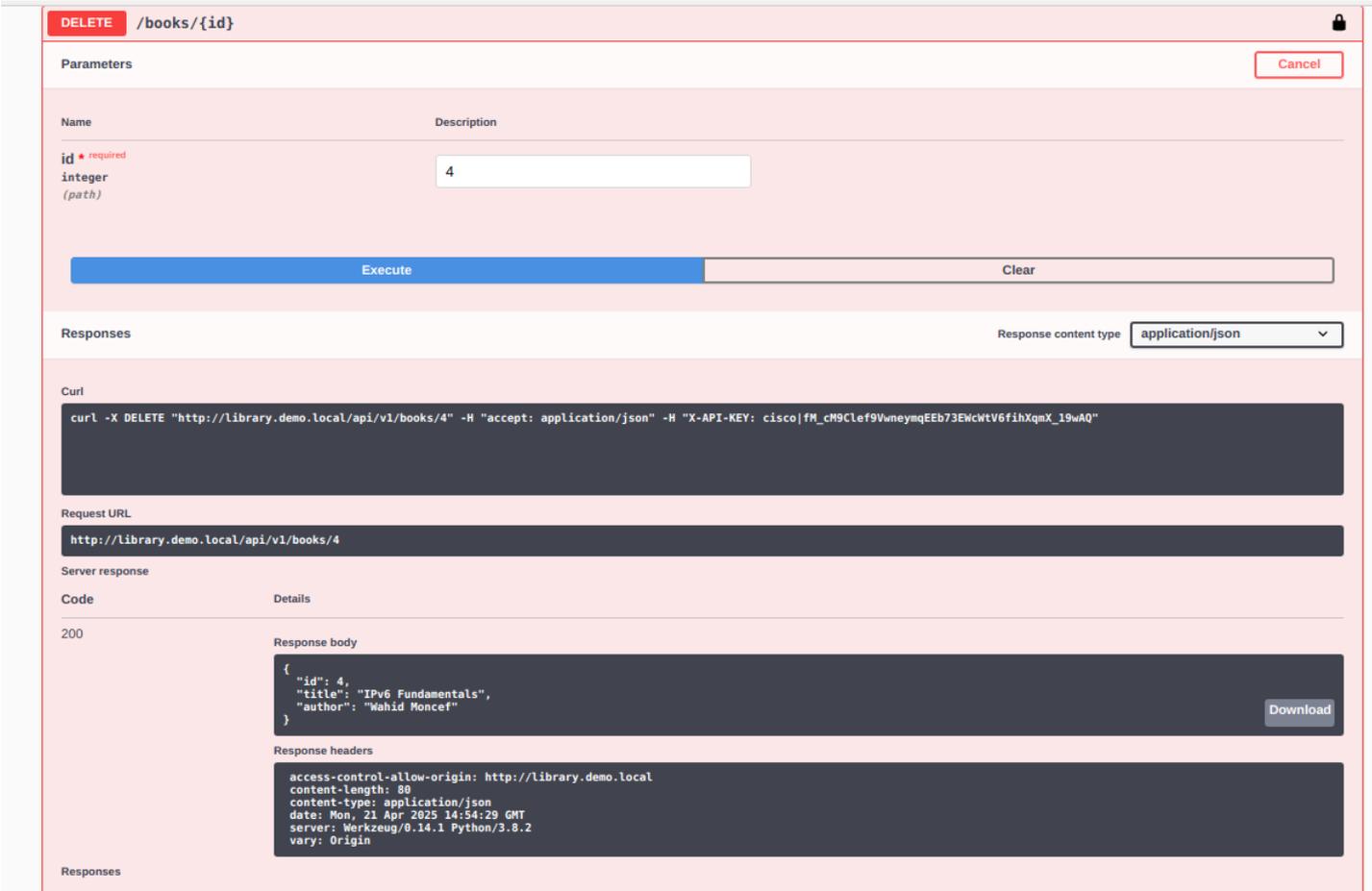
Curl

```
curl -X POST "http://library.demo.local/api/v1/books" -H "accept: application/json" -H "X-API-KEY: cisco|fm_cM9Clef9VwneymqEEb73EwcWtV6fihXqmX_19wA0" -d "{ \"id\": 5, \"title\": \"ttttals\", \"author\": \"Wahid Moncef\"}"
```

Retournons à l'onglet School Library (<http://library.demo.local>) et actualisez la page pour vérifier l'ajout :



Étape 4: Supprimer un livre avec l'API DELETE /books{id} :



Retournons à l'onglet School Library (<http://library.demo.local>) et actualisez la page pour vérifier suppression de livre id 4 :

School Library x School Library API x +

Not secure | library.demo.local/

# Welcome to the School Library

Navigation

- Our Books
- Open Hours
- Contact

Here is a list of our books currently in the library.

Book ID	Title	Author
0	IP Routing Fundamentals	Mark A. Sportack
1	Python for Dummies	Stef Maruch Aahz Maruch
2	Linux for Networkers	Cisco Systems Inc.
3	NetAcad: 20 Years Of Online-Learning	Cisco Systems Inc.
5	ttttals	Wahid Moncef

### Partie 3 : Utilisation de Postman pour effectuer des appels API:

Étape 1 : Ouvrir Postman & Étape 2 : Lister les livres avec l'API GET /books:

Postman

File Edit View Help

New Import Runner

My Workspace

Filter

History Collections APIs

Save Responses Clear all

Today

GET http://library.demo.local/api/v1/books

Launchpad GET http://library.demo.local/api/v1...

Untitled Request

GET http://library.demo.local/api/v1/books

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE
Key	Value

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

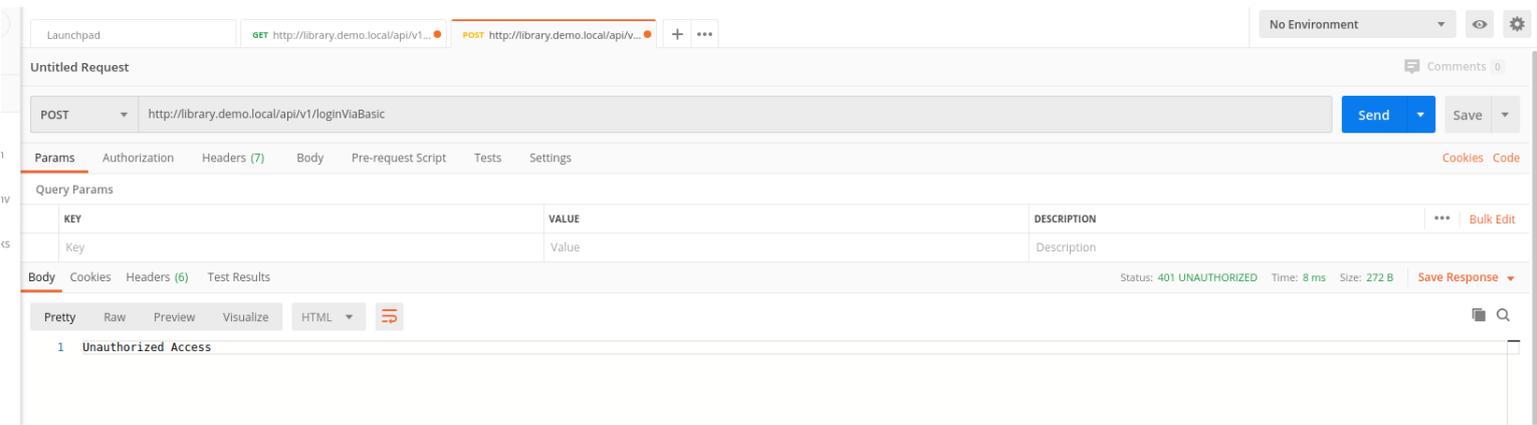
```

1  [
2  {
3    "id": 0,
4    "title": "IP Routing Fundamentals",
5    "author": "Mark A. Sportack"
6  },
7  {
8    "id": 1,
9    "title": "Python for Dummies",
10   "author": "Stef Maruch Aahz Maruch"
11 },
12 {
13   "id": 2,
14   "title": "Linux for Networkers",
15   "author": "Cisco Systems Inc."
16 },
17 {
18   "id": 3,
19   "title": "NetAcad: 20 Years Of Online-Learning",
20   "author": "Cisco Systems Inc."
21 },
22 {
23   "id": 5,
24   "title": "ttttals",
25   "author": "Wahid Moncef"
26 }
-- ]

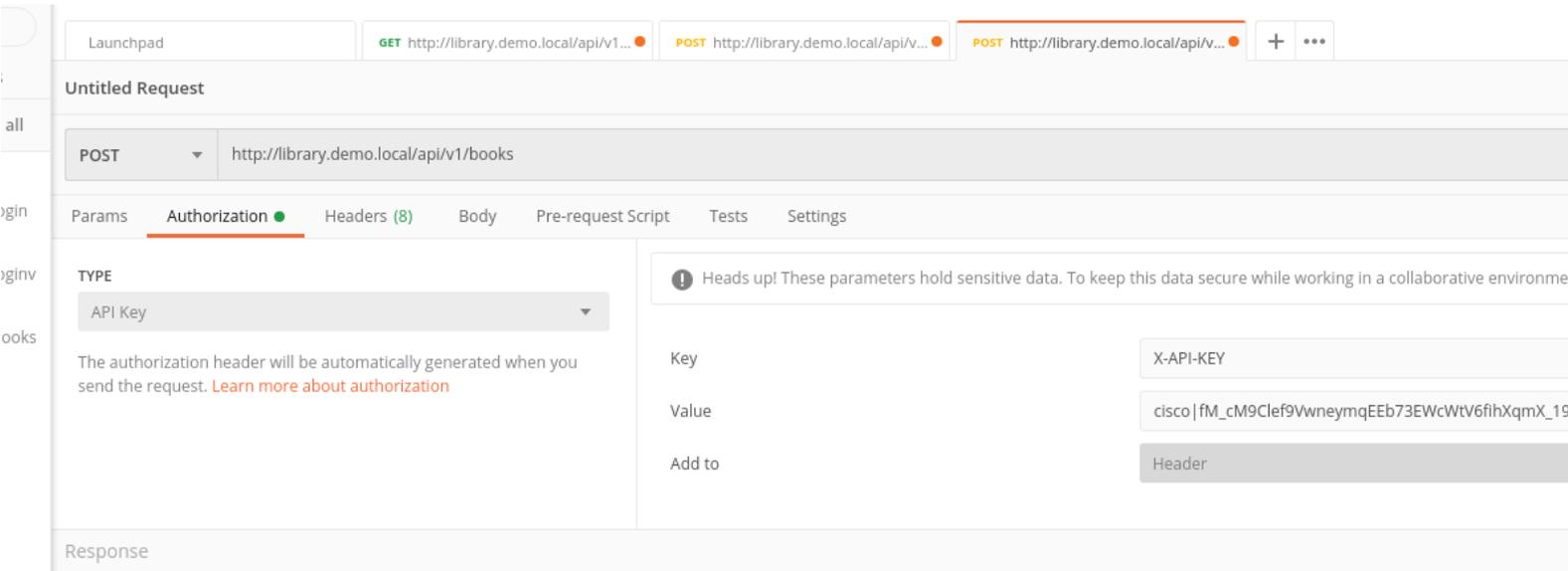
```

### Étape 3 : Obtenir un jeton (token) avec l'API POST /loginViaBasic:

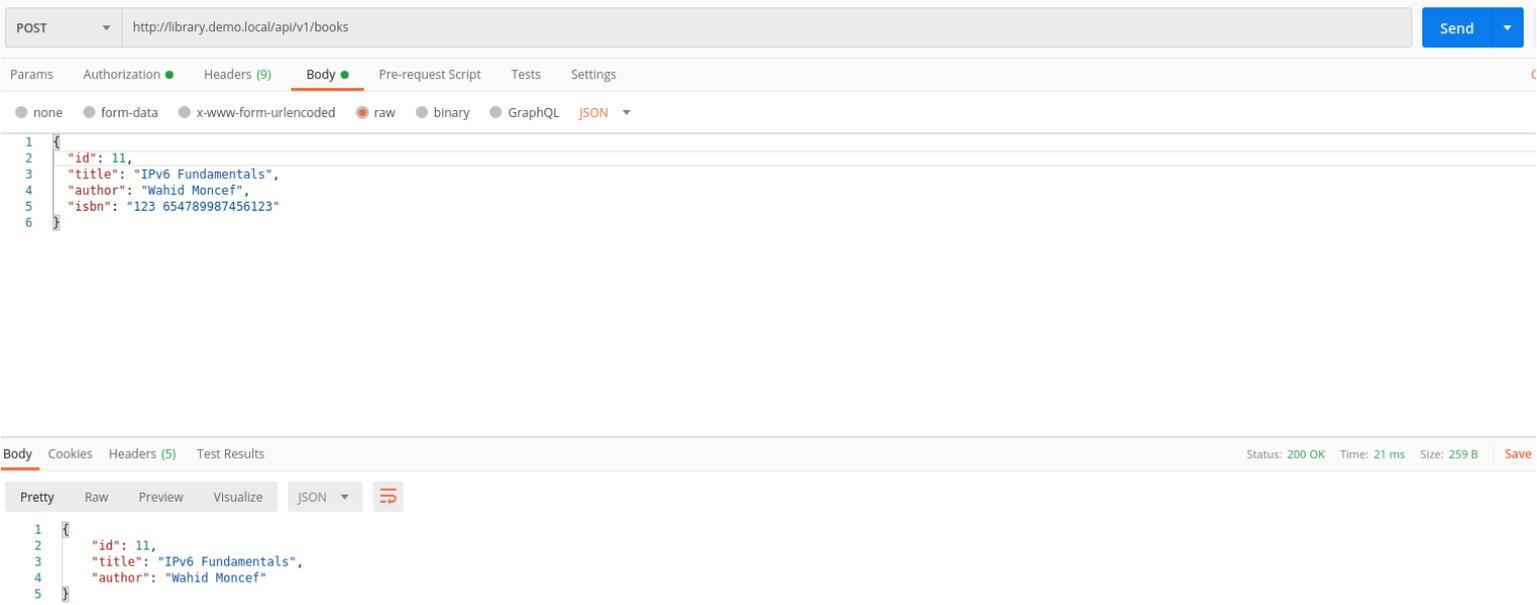
Nous n'avons pas l'accès , car on a pas load nos coordonnées



### Étape 4 : Ajouter un livre via POST /books



Remplissons le body par un livre avec titre (IPV6 FUNDAMENTALS) :



### Étape 5 : Vérifier les livres ajoutés via GET :

The screenshot shows a REST client interface with a GET request to `http://library.demo.local/api/v1/books?includeISBN=true&sortBy=author`. The response is a JSON array of book objects, displayed in a 'Pretty' format. The response contains 6 book entries with their respective IDs, titles, authors, and ISBNs.

```
1 [
2   {
3     "id": 2,
4     "title": "Linux for Networkers",
5     "author": "Cisco Systems Inc.",
6     "isbn": "000-0000000123"
7   },
8   {
9     "id": 3,
10    "title": "NetAcad: 20 Years Of Online-Learning",
11    "author": "Cisco Systems Inc.",
12    "isbn": "000-0000001123"
13  },
14  {
15    "id": 0,
16    "title": "IP Routing Fundamentals",
17    "author": "Mark A. Sportack",
18    "isbn": "978-1578700714"
19  },
20  {
21    "id": 1,
22    "title": "Python for Dummies",
23    "author": "Stef Maruch Aahz Maruch",
24    "isbn": "978-0471778646"
25  },
26  {
27    "id": 5,
28    "title": "ttttals",
29    "author": "Wahid Moncef"
30  },
31  {
32    "id": 10,
33    "title": "IPv6 Fundamentals",
34    "author": "Wahid Moncef",
35    "isbn": "123 654789987456123"
36  },
37  {
```

### Étape 6 : Utiliser des paramètres supplémentaires avec GET /books :

ISBN :

The screenshot shows the parameter configuration interface for the `GET /books` endpoint. It lists several query parameters with their descriptions and current values:

- includeISBN**: boolean (query), Include in the results the ISBN numbers. Default=false. Value: `true`.
- sortBy**: string (query), Sort results using the specified parameter. Default=id. Value: `author`.
- author**: string (query), Return only books by the given Author. Value: `author - Return only books by the given Auth`.
- page**: integer (query), To save resources and bandwidth, larger replies might be broken down into smaller pages with 10 records per page. The RFC5988 (Web Linking) standard in HTTP Reply Headers is used for pagination. The "page" parameter is then used to specify the page number. Value: `page - To save resources and bandwidth, lar`.

Buttons for 'Execute' and 'Clear' are visible at the bottom.

Untitled Request

GET <http://library.demo.local/api/v1/books?includeISBN=true&sortBy=author>

Params ● Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	includeISBN	true
<input checked="" type="checkbox"/>	sortBy	author
	Key	Value

Response

Résultats :

Responses

Curl

```
curl -X GET "http://library.demo.local/api/v1/books?includeISBN=true&sortBy=author" -H "accept: application/json"
```

Request URL

```
http://library.demo.local/api/v1/books?includeISBN=true&sortBy=author
```

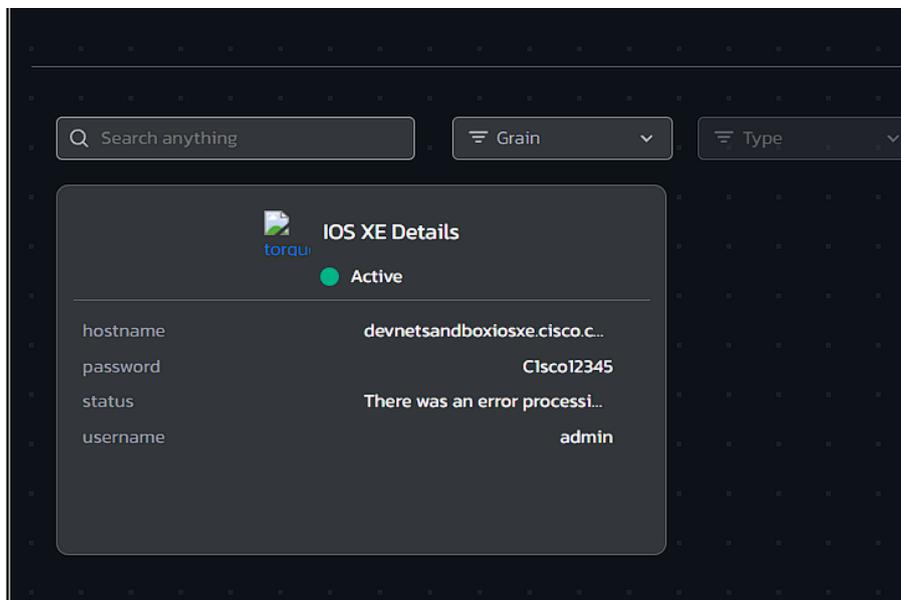
Server response

Code	Details
200	<p>Response body</p> <pre>{   {     "id": 2,     "title": "Linux for Networkers",     "author": "Cisco Systems Inc.",     "isbn": "000-0000000123"   },   {     "id": 3</pre>

```
Launchpad | GET http://library.demo.local/api/v1... | POST http://library.demo.local/api/v... | POST http://library.der
GET http://library.demo.local/api/v1/books?includeISBN=true&sortBy=author
Pretty Raw Preview Visualize JSON
1 {
2   {
3     "id": 2,
4     "title": "Linux for Networkers",
5     "author": "Cisco Systems Inc.",
6     "isbn": "000-000000123"
7   },
8   {
9     "id": 3,
10    "title": "NetAcad: 20 Years Of Online-Learning",
11    "author": "Cisco Systems Inc.",
12    "isbn": "000-0000001123"
13  },
14  {
15    "id": 0,
16    "title": "IP Routing Fundamentals",
17    "author": "Mark A. Sportack",
18    "isbn": "978-1578700714"
19  },
20  {
21    "id": 1,
22    "title": "Python for Dummies",
23    "author": "Stef Maruch Aahz Maruch",
24    "isbn": "978-0471778646"
25  },
26  {
27    "id": 5,
28    "title": "ttttals",
29    "author": "Wahid Moncef"
30  },
31  {
32    "id": 10,
33    "title": "IPv6 Fundamentals",
34    "author": "Wahid Moncef",
35    "isbn": "123 654789987456123"
36  },
37  }
```

## Section 2 Découverte et utilisation du protocole RESTCONF :

los xe est dans un état UP :



Étape 1 : Vérification de l'adresse IP du routeur CSR1kv :

```
Cat8000V#show ip int
Cat8000V#show ip interface br
Cat8000V#show ip interface brief
Interface                IP-Address      OK? Method Status      Protocol
GigabitEthernet1        10.10.20.48    YES NVRAM    up          up
GigabitEthernet2        unassigned     YES NVRAM    administratively down down
GigabitEthernet3        unassigned     YES NVRAM    administratively down down
Loopback0                10.0.0.1      YES NVRAM    up          up
Loopback10              unassigned     YES unset   up          up
Loopback21              172.16.2.254  YES manual up          up
Loopback101            unassigned     YES unset   up          up
VirtualPortGroup0       192.168.1.1   YES NVRAM    up          up
Cat8000V#
```

Étape 2 : Vérification de la connectivité réseau depuis la VM DEVASC :

Nous avons passé l'étape de Ping , mais l'étape de ssh fonctionne :

Étape 3 : Vérification de la connectivité SSH vers le routeur CSR1kv :

```
devasc@labvm: ~
File Edit View Search Terminal Help
RSA key fingerprint is SHA256:c5UmWfeS/jSAWRm/9YJtmQ8/PcLJ0m3xjSJDs4AF7IE.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'devnetsandboxiosxe.cisco.com,131.226.217.181' (RSA)
to the list of known hosts.

WELCOME TO MY ROUTER
HAVE NICE DAY
GOOD LUCK !
Password:
Password:
Password:

Welcome to the DevNet Sandbox for Cat8000V and IOS XE

The following programmability features are already enabled:

- NETCONF
- RESTCONF

Thanks for stopping by.

Cat8000V#
```

Partie 2 : Configuration du routeur IOS XE pour l'accès RESTCONF :

Étape 1 : Vérification du fonctionnement des démons RESTCONF :

```
Loopback21                172.16.2.254    YES manual up
Loopback101              unassigned     YES unset   up
VirtualPortGroup0       192.168.1.1   YES NVRAM    up
Cat8000V#show platform software yang-management process
confd                    : Running
nesd                     : Running
syncfd                   : Running
ncsshd                   : Running
dmiauthd                 : Running
nginx                    : Running
ndbmand                  : Running
pubd                     : Running

Cat8000V#
```

## Étape 2 : Activation du service RESTCONF & Étape 3 : Activation du service HTTPS :

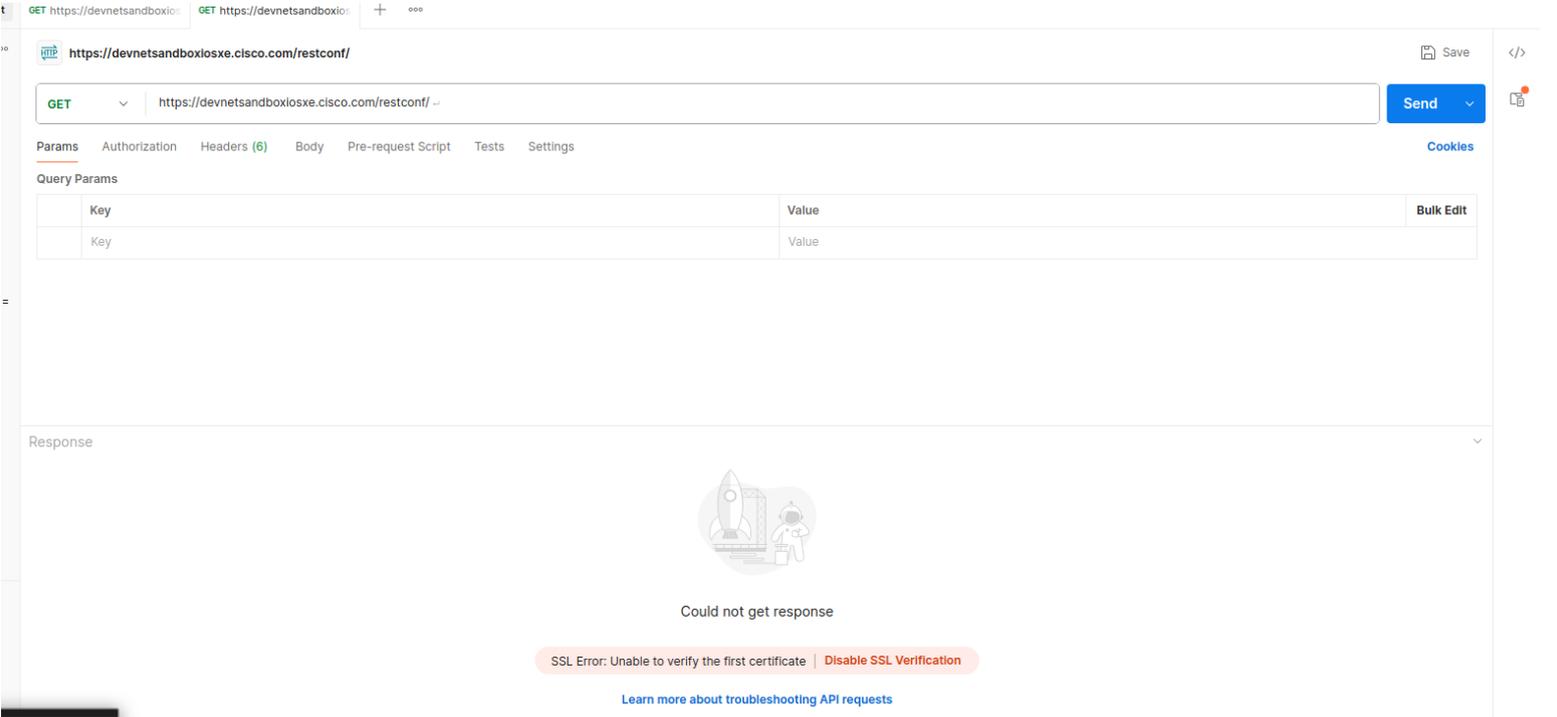
```
Cat8000V#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Cat8000V(config)#
Cat8000V(config)#restconf
Cat8000V(config)#configure terminal
      ^
% Invalid input detected at '^' marker.

Cat8000V(config)#ip http secure-server
Cat8000V(config)#ip http authentication local
Cat8000V(config)#exit
Cat8000V#
```

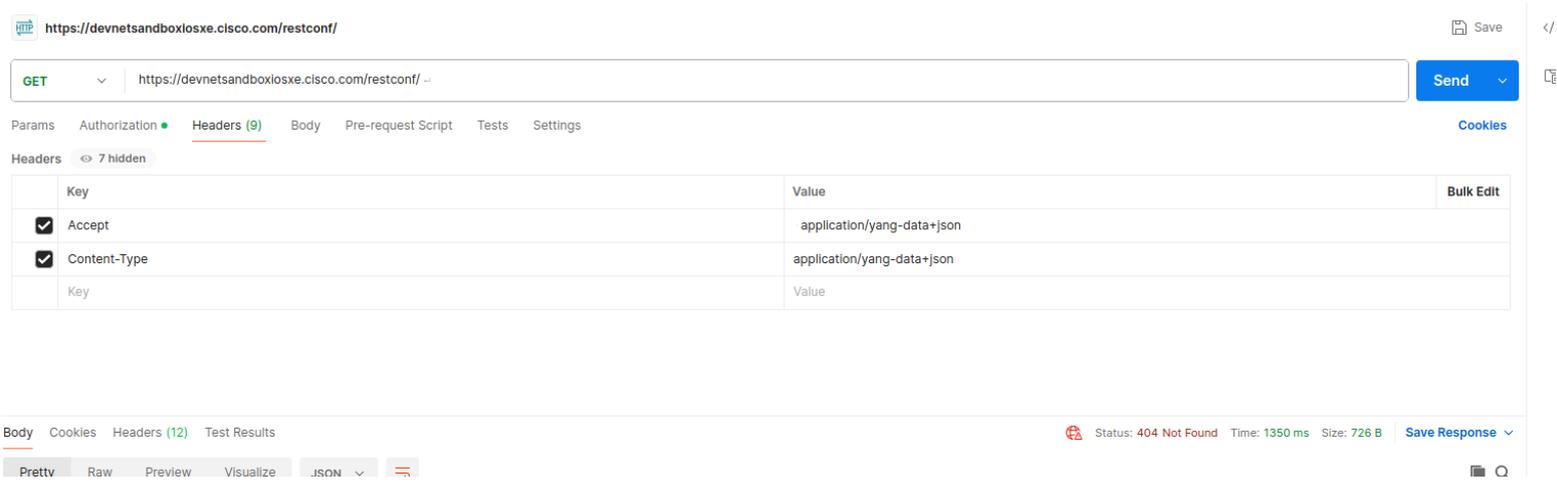
School Library - Chromium

## Partie 3 : Configuration de Postman pour interagir avec RESTCONF:

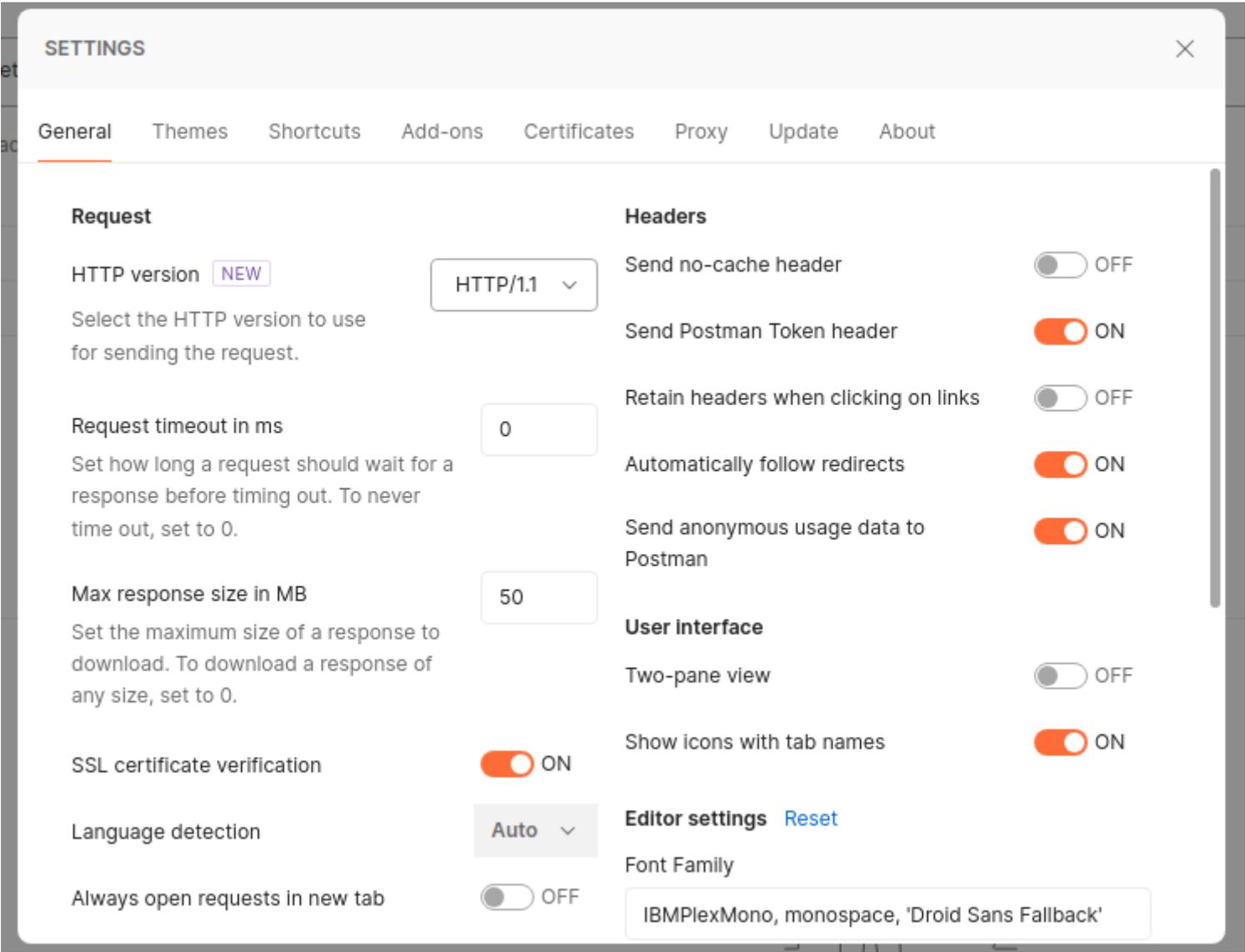
### Étape 1 : Lancer Postman sur la machine virtuelle DEVASC :



### Étape 2 : Créer une nouvelle requête GET RESTCONF :



Désactivons SSL :



Nous avons trouvé des problèmes de connectivité en couche OSI niveau 3 , (par contre au niveau 7 ‘HTTPS’ fonctionne bien !)

```
devasc@labvm:~$ ping -c 5 devnetsandboxiosxe.cisco.com
PING devnetsandboxiosxe.cisco.com (131.226.217.181) 56(84) bytes of data:
--- devnetsandboxiosxe.cisco.com ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4077ms

devasc@labvm:~$
```