

**DNA Center Authentication API - Postman**

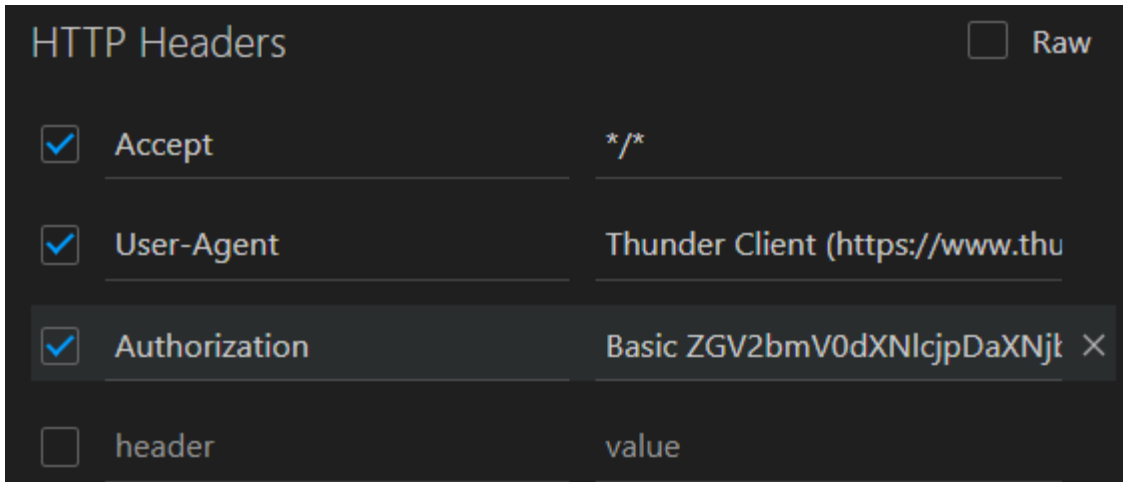
¿En qué consiste la autenticación HTTP Básica?

Consiste en que el cliente envía una petición al servidor mediante nombre de dominio

¿Cuál es el objetivo de utilizar TLS en la Autenticación Básica?

Tiene como objetivo la protección de los datos

¿Qué headers usaste para extraer el Token?

**Get started with Authentication API - Python**

¿Qué módulos se instalan para poder consumir REST API con Python?

Requests y JSON

Una vez hecha la petición con Python ¿Qué cabeceras o Headers utilizaste?

<input checked="" type="checkbox"/>	Postman-Token	<i>i</i>	<calculated when request is sent>
<input checked="" type="checkbox"/>	Host	<i>i</i>	<calculated when request is sent>
<input checked="" type="checkbox"/>	User-Agent	<i>i</i>	PostmanRuntime/7.42.0
<input checked="" type="checkbox"/>	Accept	<i>i</i>	*/*
<input checked="" type="checkbox"/>	Accept-Encoding	<i>i</i>	gzip, deflate, br
<input checked="" type="checkbox"/>	Connection	<i>i</i>	keep-alive
<input checked="" type="checkbox"/>	x-auth-token		eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI2...

En la función get\_auth\_token la URL que la compone

```
url = 'https://' + env_lab.DNA_CENTER['host'] + endpoint
```

Host

```
1
2  DNA_CENTER = {
3      "host": "sandboxdnac.cisco.com",
4      "port" : "443",
5      "username": "devnetuser",
6      "password": "Cisco123!"
7  }
```

Endpoint para buscar en la api

```
endpoint = '/dna/system/api/v1/auth/token'
```

### Retrieving a list of network devices with Postman

Una vez accedido al Cisco DNA Center ¿Lista la información que muestra para cada dispositivo de red? **Si la enlista**

Menciona el nombre del header para realizar la petición al servidor

**x-auth-token**

¿Cuál es la diferencia entre Params y Headers? ¿Cuándo usar uno y otro?

**Los parámetros se envían en la URL, en métodos GET**

**Los headers como su nombre lo indica se envía en la cabeza de la solicitud, más no en la URL**

¿Qué método se utiliza para listar los dispositivos de red? ¿Por qué es necesario?

**Se utiliza el método request.get() y se utiliza para obtener los datos desde la API**

### Creating a network device list with a Python function

Talk

Documenta las funciones get\_device\_list, print\_device\_list, get\_auth\_token

## PROGRAMACIÓN DE REDES

```
5 def get_device_list():
6     #Función para recuperar la lista de dispositivos de red desde Cisco DNA Center.
7     token = get_auth_token() # Obtiene el token de autenticación
8     url = "https://sandboxnac.cisco.com/api/v1/network-device" # URL del endpoint de dispositivos de red
9     hdr = {'x-auth-token': token, 'content-type': 'application/json'} # Headers de la solicitud
10
11     resp = requests.get(url, headers=hdr, verify=False) # Realiza la solicitud GET
12
13     device_list = resp.json() # Convierte la respuesta a JSON
14     print_device_list(device_list) # Imprime la lista de dispositivos
```

```
15 def print_device_list(device_json):
16     # Imprime el encabezado de la tabla con nombres de columnas, ajustado para alinearse.
17     print("{0:42}{1:17}{2:12}{3:18}{4:12}{5:16}{6:15}".
18           format("hostname", "mgmt IP", "serial", "platformId", "SW Version", "role", "Uptime"))
19
20     # Itera sobre cada dispositivo en la respuesta JSON.
21     for device in device_json['response']:
22         # Asigna "N/A" al tiempo de actividad si el valor es None; de lo contrario, usa el valor de 'upTime'.
23         uptime = "N/A" if device['upTime'] is None else device['upTime']
24
25         # Verifica si el número de serie contiene comas (indicando varios números de serie).
26         if device['serialNumber'] is not None and "," in device['serialNumber']:
27             # Empareja cada número de serie con su correspondiente platformId en una lista de tuplas.
28             serialPlatformList = zip(device['serialNumber'].split(","), device['platformId'].split(","))
29         else:
30             # Si solo hay un número de serie y un platformId, los agrupa en una tupla única.
31             serialPlatformList = [(device['serialNumber'], device['platformId'])]
32
33         # Itera sobre cada par (serialNumber, platformId) en la lista.
34         for (serialNumber, platformId) in serialPlatformList:
35             # Imprime los detalles de cada dispositivo en formato tabular.
36             print("{0:42}{1:17}{2:12}{3:18}{4:12}{5:16}{6:15}".
37                   format(device['hostname'],
38                           device['managementIpAddress'],
39                           serialNumber,
40                           platformId,
41                           device['softwareVersion'],
42                           device['role'], uptime))
```

## PROGRAMACIÓN DE REDES

```
43 def get_auth_token():
44     """
45     Construye y envía una solicitud de autenticación para obtener un token de Cisco DNA Center.
46     """
47     # Define la URL del endpoint de autenticación de Cisco DNA Center.
48     url = 'https://sandboxdnac.cisco.com/dna/system/api/v1/auth/token'
49
50     resp = requests.post(url, auth=HTTPBasicAuth(DNAC_USER, DNAC_PASSWORD), verify=False)
51
52     # Extrae el token de autenticación desde la respuesta JSON.
53     token = resp.json()['Token']
54
55     # Devuelve el token de autenticación para su uso en futuras solicitudes.
56     return token
57 if __name__ == "__main__":
58     get_device_list()
```

## Creating a list of network device interfaces with a Python function

¿Cuál es la URL a emplear?

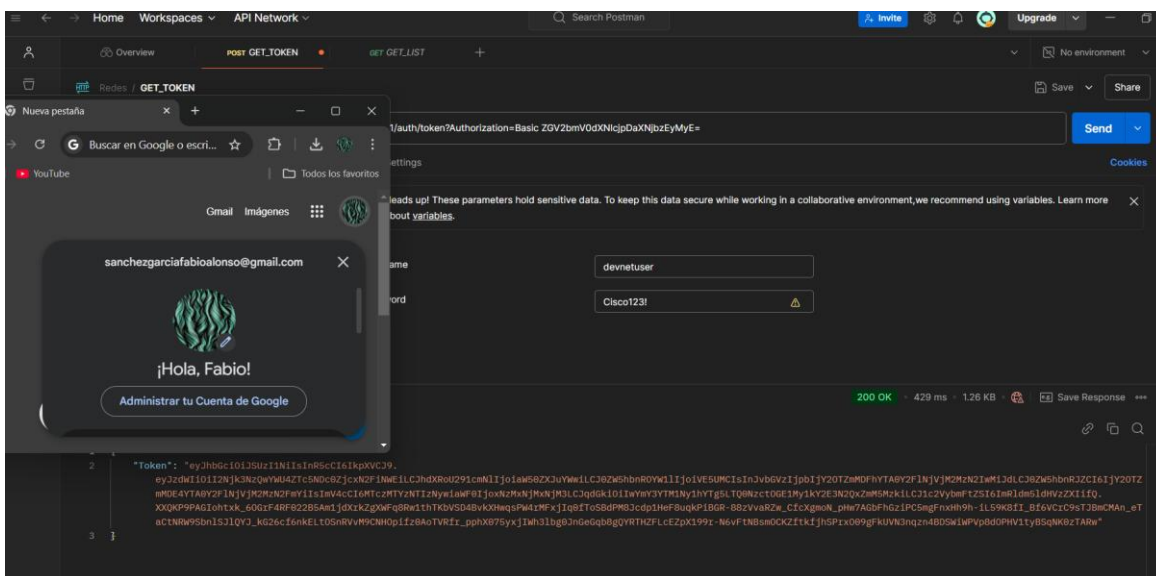
<https://sandboxdnac.cisco.com/api/v1/network-device>

## Execute Commands Across the Network with Cisco DNA Center Platform and Command Runner

¿Cuál es la URL?

<https://sandboxdnac.cisco.com/>  
Username: `devnetuser`  
Password: `Cisco123!`

Crea un petición Postman y pega tu salida (no se te olvide la autoría)



**Lab Remote Command Execution and Collecting Results**

¿Qué entiendes por Universal Unique Identifier (UUID)?

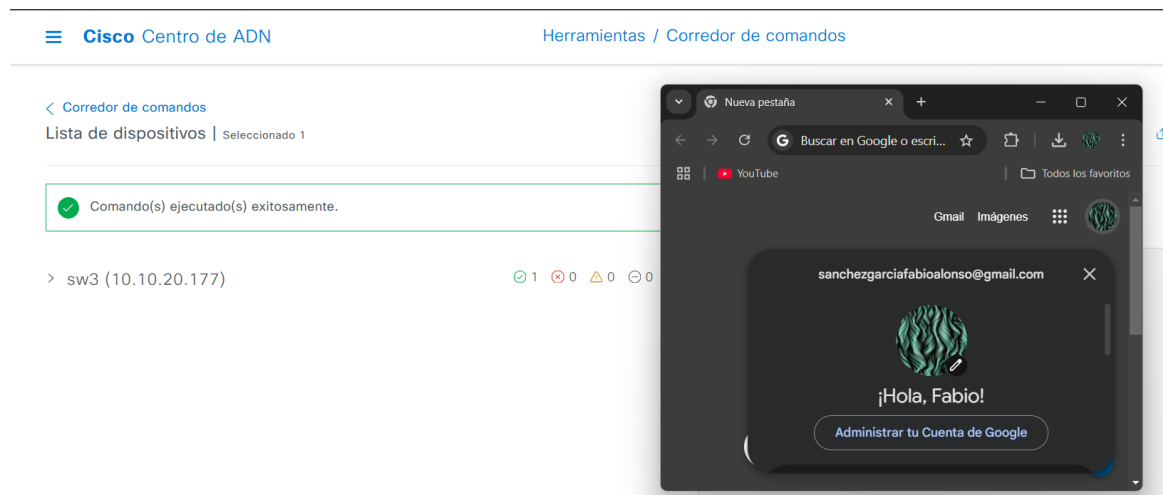
Algo como Identificador Único Universal, como obtener algunos datos específicos

¿Qué comandos ejecutan para dar la salida? Explica el comando

Usé el cd porque me pareció el más sencillo en utilizar, puesto que el comando es para moverse entre subdirectorios, por eso lo utilicé

Haz la prueba de los comandos en <https://sandboxdnac.cisco.com/dna/tools/command-runner>

Checa Tools topología

**Understanding Template Programmer and API**

Navega hasta la opción y verifica las opciones

NO EXISTE LA OPCION EN DEVNET-SANDBOX