

Programación de Redes

Unidad III. Programación de Redes

2.9 Lab – NETCONF wPython Get Operational Data

Reyna Yazmin Ríos Martínez

N° Control. 1222100500

Fecha: 04 / Dic / 2023

The screenshot displays a desktop environment with three main windows:

- PDF Document (Left):** A document titled "2.9 Lab - NETCONF wPython" with instructions for creating a Python script to connect to a Cisco IOS XE device via NETCONF. The instructions include:
 - a. In Python IDLE, create a new Python script file.
 - b. In the new Python script file editor, import the "manager" class from the `ncclient` module and the `xml.dom.minidom` module:

```
from ncclient import manager
import xml.dom.minidom
```
 - c. Set up an `m` connection object using the `manager.connect()` function to the IOS XE device:

```
m = manager.connect(
    host='10.10.20.48',
    port=830,
    username='cisco',
    password='cisco12345',
    hostkey_verify=False
)
```
 - d. After a successful NETCONF connection, using the `get()` function of the "m" NETCONF object to retrieve and print the device's operational data. The `get()` function expects a "filter" that defines the NETCONF filter.The document also includes a note: "Note: Executing the `get()` function without a filter is similar to execute 'debug all'." and a filter definition:

```
netconf_filter = """
<filter>
```
- Python Script (Middle):** A file named `Lab29_ryazmin.py` containing the following code:

```
1 ...
2 Nombre: Reyna Yazmin Rios
3 Fecha: 04 / Dic / 2023
4 Descripción: Lab 2.9 NETCONF wPython Get Operational Data
5 ...
6 from ncclient import manager
7 import xml.dom.minidom
8
9 m = manager.connect(
10     host='10.10.20.48',
11     port=830,
12     username='cisco',
13     password='cisco12345',
14     hostkey_verify=False
15 )
16
```
- Cisco AnyConnect Client (Bottom Right):** A window showing a successful VPN connection to `devnetsandbox-usw1-reserva01.cisco.com:20293`. The status is "Connected" and the IP address is `IPv4`.

El filtro no se encuentra:

```
netconf_filter = """
<filter>

  <interfaces-state xmlns="urn:ietf:params:xml:ns:yang:ietf-
interfaces"/>
</filter>
"""
```



```
File Edit Selection View Go Run ... unidad3
Lab29_ryazmin.py 2 X
Lab29_ryazmin.py > ...
1 """
2     Nombre: Reyna Yazmin Rios
3     Fecha: 04 / Dic / 2023
4     Descripción: Lab 2.9 NETCONF wPython Get Operational Data
5 """
6 from ncclient import manager
7 import xml.dom.minidom
8
9 m = manager.connect(
10     host='10.10.20.48',
11     port=830,
12     username='developer',
13     password='Cisco12345',
14     hostkey_verify=False
15 )
16
17 netconf_filter = """
18 <filter>
19   <interfaces-state xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces"/>
20 </filter>
21 """
22 netconf_reply = m.get(filter = netconf_filter)
23 print(xml.dom.minidom.parseString(netconf_reply.xml).toprettyxml())
24
25 import xmltodict
26 netconf_reply_dict = xmltodict.parse(netconf_reply.xml)
27 for interface in netconf_reply_dict["rpc-reply"]["data"]["interfaces-state"]["interface"]:
28     print("Name: {} MAC: {} Input: {} Output {}".format(
29         interface["name"],
30         interface["phys-address"],
31         interface["statistics"]["in-octets"],
32         interface["statistics"]["out-octets"]
33     ))
34
Ln 23, Col 68 Spaces: 4 UTF-8 CRLF Python 3.11.5 64-bit
13°C Nublado 04-25 p. m. 04/12/2023
```

```
File Edit Selection View Go Run ... unidad3
Lab29_ryazmin.py 2 •
Lab29_ryazmin.py > ...
1 """
2     Nombre: Reyna Yazmin Rios
3     Fecha: 04 / Dic / 2023
4     Descripción: Lab 2.9 NETCONF wPython Get Operational Data
5 """
6 from ncclient import manager
7 import xml.dom.minidom
8
9 m = manager.connect(
10     host='10.10.20.48',
11     port=830,
12     username='developer',
13     password='Cisco12345',
14     hostkey_verify=False
15 )
16
17 netconf_filter = """
18 <filter>
19   <interfaces-state xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces"/>
20 </filter>
21 """
22 netconf_reply = m.get(filter = netconf_filter)
23 print(xml.dom.minidom.parseString(netconf_reply.xml).toprettyxml())
24
25 import xmltodict
26 netconf_reply_dict = xmltodict.parse(netconf_reply.xml)
27 for interface in netconf_reply_dict["rpc-reply"]["data"]["interfaces-state"]["interface"]:
28     print("Name: {} MAC: {} Input: {} Output {}".format(
29         interface["name"],
30         interface["phys-address"],
31         interface["statistics"]["in-octets"],
32         interface["statistics"]["out-octets"]
33     ))
34
Ln 34, Col 6 Spaces: 4 UTF-8 CRLF Python 3.11.5 64-bit
13°C Nublado 04-26 p. m. 04/12/2023
```

```
C:\Windows\system32\cmd.exe
encoding="UTF-8"?><error-info xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"><bad-element>filter
</bad-element>\n</error-info>\n', 'path': '\n /rpc/get\n ', 'message': None}

C:\Users\yazmi\Documents\unidad3>python Lab29_ryazmin.py
Traceback (most recent call last):
  File "C:\Users\yazmi\Documents\unidad3\Lab29_ryazmin.py", line 8, in <module>
    netconf_reply_dict = xmldict.parse(netconf_reply.xml)
    ^^^^^^^^^^^^^^^^^
NameError: name 'netconf_reply' is not defined

C:\Users\yazmi\Documents\unidad3>python Lab29_ryazmin.py
Traceback (most recent call last):
  File "C:\Users\yazmi\Documents\unidad3\Lab29_ryazmin.py", line 8, in <module>
    netconf_reply_dict = xmldict.parse(netconf_reply.xml)
    ^^^^^^^^^^^^^^^^^
NameError: name 'netconf_reply' is not defined

C:\Users\yazmi\Documents\unidad3>python Lab29_ryazmin.py
Traceback (most recent call last):
  File "C:\Users\yazmi\Documents\unidad3\Lab29_ryazmin.py", line 22, in <module>
    netconf_reply = m.get(filter = netconf_filter)
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\yazmi\AppData\Local\Programs\Python\Python312\Lib\site-packages\ncclient\manager.py", line 257, in execute
    return cls(self._session,
    ^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\yazmi\AppData\Local\Programs\Python\Python312\Lib\site-packages\ncclient\operations\retrieve.py", line 91, in request
    return self._request(node)
    ^^^^^^^^^^^^^^^^^
  File "C:\Users\yazmi\AppData\Local\Programs\Python\Python312\Lib\site-packages\ncclient\operations\rpc.py", line 375, in _request
    raise self._reply.error
ncclient.operations.rpc.RPCError: {'type': 'protocol', 'tag': 'unknown-element', 'app_tag': None, 'severity': 'error', 'info': '<?xml version="1.0"
encoding="UTF-8"?><error-info xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"><bad-element>filter
</bad-element>\n</error-info>\n', 'path': '\n /rpc/get\n ', 'message': None}

C:\Users\yazmi\Documents\unidad3>
```