NOMBRE: Isidro Lara Lopez
CARRERA: INFRAESTRUCTURA DE REDES DIGITALES
DOCENTE: BARRON RODRIGUEZ GABRIEL
NO. CONTROL: 1221100381
FECHA: 14 DE DICIEMBRE DEL 2022
GRUPO: GIR0441

www.netacad.com

Lab – CLI Automation with Python using netmiko

Objectives

Part 1: Install the netmiko Python module

Part 2: Connect to IOS XE's SSH service using netmiko

Part 3: Use netmiko to gather information from the device

Part 4: Use netmiko to alter configuration on the device

Background / Scenario

For simple network automation using a remote telnet or ssh based command line, network administrators have been using various screen scraping techniques for a long period of time. Initially the "expect" based scripts we utilized to automate entering commands when a specific expected string appeared on the command line. With the evolution of the Python language, the netmiko Python module has emerged as an open source project hosted and maintained on GitHub.com that provides a simple network automation interface using similar techniques like the "expect" based scripts.

In this lab activity, you will identify the potential but also the limitations of using netmiko to transport CLI commands for network automation.

Required Resources

- Access to a router with the IOS XE operating system version 16.6 or higher.
- Access to the Internet
- Python 3.x environment

Instructions

Part 1: Install the netmiko Python module

In this part, you will install netmiko module into your Python environment. Netmiko is a python module that simplifies ssh CLI connection to network devices. It has built in functionality to identify to execute "exec mode" commands, as well as apply new commands in the running configuration.

Explore the netmiko module on the project GitHub repository: https://github.com/ktbyers/netmiko

Step 1: Use pip to install netmiko.

a. Start a new Windows command prompt (cmd).

b. Install netmiko using pip in the Windows command prompt:

```
pip install netmiko
```

```
                                                       ilara@LPI: ~                                                         _  □  ⊗
 Archivo  Editar  Ver  Buscar  Terminal  Ayuda
ilara@LPI:~$ pip install netmiko
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: netmiko in ./.local/lib/python3.10/site-packages (4.1.2)
Requirement already satisfied: ntc-templates>=2.0.0 in ./.local/lib/python3.10/site-packages (from netmiko) (3.1.0)
Requirement already satisfied: textfsm==1.1.2 in ./.local/lib/python3.10/site-packages (from netmiko) (1.1.2)
Requirement already satisfied: paramiko>=2.7.2 in ./.local/lib/python3.10/site-packages (from netmiko) (2.11.0)
Requirement already satisfied: tenacity in ./.local/lib/python3.10/site-packages (from netmiko) (8.1.0)
Requirement already satisfied: scp>=0.13.3 in ./.local/lib/python3.10/site-packages (from netmiko) (0.14.4)
Requirement already satisfied: pyserial in ./.local/lib/python3.10/site-packages (from netmiko) (3.5)
Requirement already satisfied: setuptools>=38.4.0 in /usr/lib/python3/dist-packages (from netmiko) (59.6.0)
Requirement already satisfied: pyyaml>=5.3 in /usr/lib/python3/dist-packages (from netmiko) (5.4.1)
Requirement already satisfied: future in ./.local/lib/python3.10/site-packages (from textfsm==1.1.2->netmiko) (0.18.2)
Requirement already satisfied: six in /usr/lib/python3/dist-packages (from textfsm==1.1.2->netmiko) (1.16.0)
Requirement already satisfied: pynacl>=1.0.1 in /usr/lib/python3/dist-packages (from paramiko>=2.7.2->netmiko) (1.5.0)
Requirement already satisfied: bcrypt>=3.1.3 in ./.local/lib/python3.10/site-packages (from paramiko>=2.7.2->netmiko) (4.0.1)
Requirement already satisfied: cryptography>=2.5 in /usr/lib/python3/dist-packages (from paramiko>=2.7.2->netmiko) (3.4.8)
Requirement already satisfied: cffi>=1.4.1 in ./.local/lib/python3.10/site-packages (from pynacl>=1.0.1->paramiko>=2.7.2->netmiko) (1.15.1)
Requirement already satisfied: pycparser in ./.local/lib/python3.10/site-packages (from cffi>=1.4.1->pynacl>=1.0.1->paramiko>=2.7.2->netmiko) (2.21)
ilara@LPI:~$ ▌
```

   c.   Verify that netmiko has been successfully installed. Start Python IDLE and in the interactive shell try to import the netmiko module:

```python
import netmiko
```

# Part 2: Connect to IOS XE's SSH service using netmiko

Connect to IOS XE's SSH service using netmiko.

The netmiko module provides a "`ConnectHandler()`" function to setup the remote ssh connection. After a successful connection, the returned object represents the ssh cli connection to the remote device.

   a.   In Python IDLE, create a new Python script file:

   b.   In the new Python script file editor, import the "`ConnectHandler()`" function from the netmiko module:

```python
from netmiko import ConnectHandler
```

   c.   Setup a sshCli connection object using the `ConnectHandler()` function to the IOS XE device.

         www.netacad.com
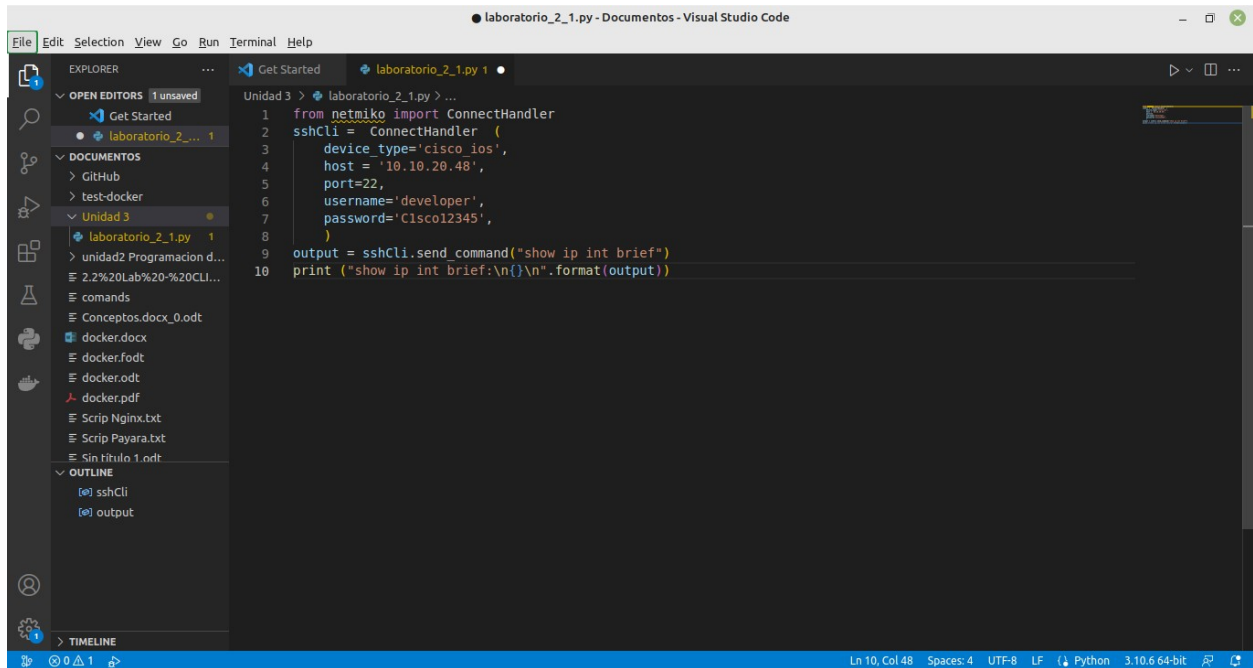
The parameters of the `ConnectHandler()` function are:

- `device_type` – identifies the remote device type
- `host` – the address (host or IP) of the remote device (adjust the IP address "192.168.56.101" to match your router's current address)
- `port` – the remote port of the ssh service
- `username` – remote ssh username (in this lab "cisco" for that was setup in the IOS XE VM)
- `password` – remote ssh password (in this lab "cisco123!" for that was setup in the IOS XE VM)

Part 3: Use netmiko to gather information from the device

Send show commands and display the output

a. Using the `sshCli` object, returned by the `ConnectHandler()` function that represents the ssh cli remote session, send some "show" command and print the output. Use the `send_command()` function of the sshCli object with a string parameter that represents the command you wish to execute in the exec mode:
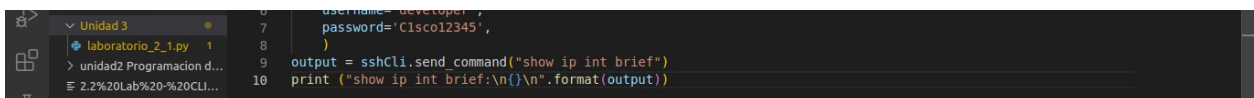
comando que nos dara la salida de las interfaces que tiene el router .

b. Execute the Python script file to see the results.

If you have not saved the script file yet, you will be prompted to save it before it is executed.

c. Verify the results:

observamos la salida del escrip creado para mostar las interfaces del router .

d. Verify the data type of the "output" variable. How would you extract the IP address and the Interface Name into variables? What if there were multiple interfaces?

Part 4: Use netmiko to alter configuration on the device

In the following steps, you will alter the configuration of the device by creating new loopback interfaces.

Create a new loopback interface

Using the `sshCli` object, returned by the `ConnectHandler()` function that represents the ssh cli remote session, send some configuration command and print the output. Use the `send_config_set()` function of the sshCli object with a list parameter including the configuration commands as strings you wish to execute in the exec mode:

```
config_commands = [
    'int loopback 1',
    'ip address 2.2.2.2 255.255.255.0',
    'description WHATEVER'
                    ]

output = sshCli.send_config_set(config_commands)
```
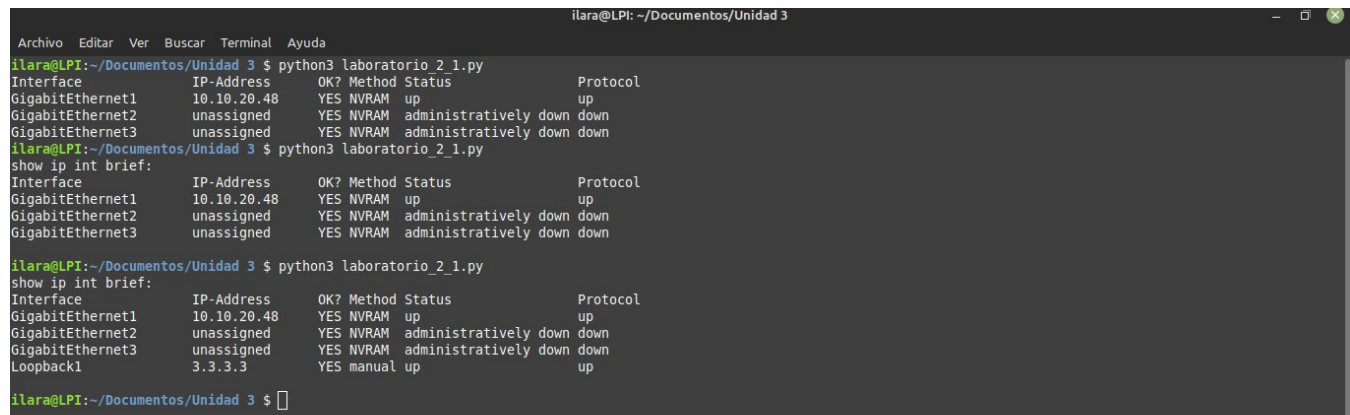
Lab – CLI Automation with Python using netmiko
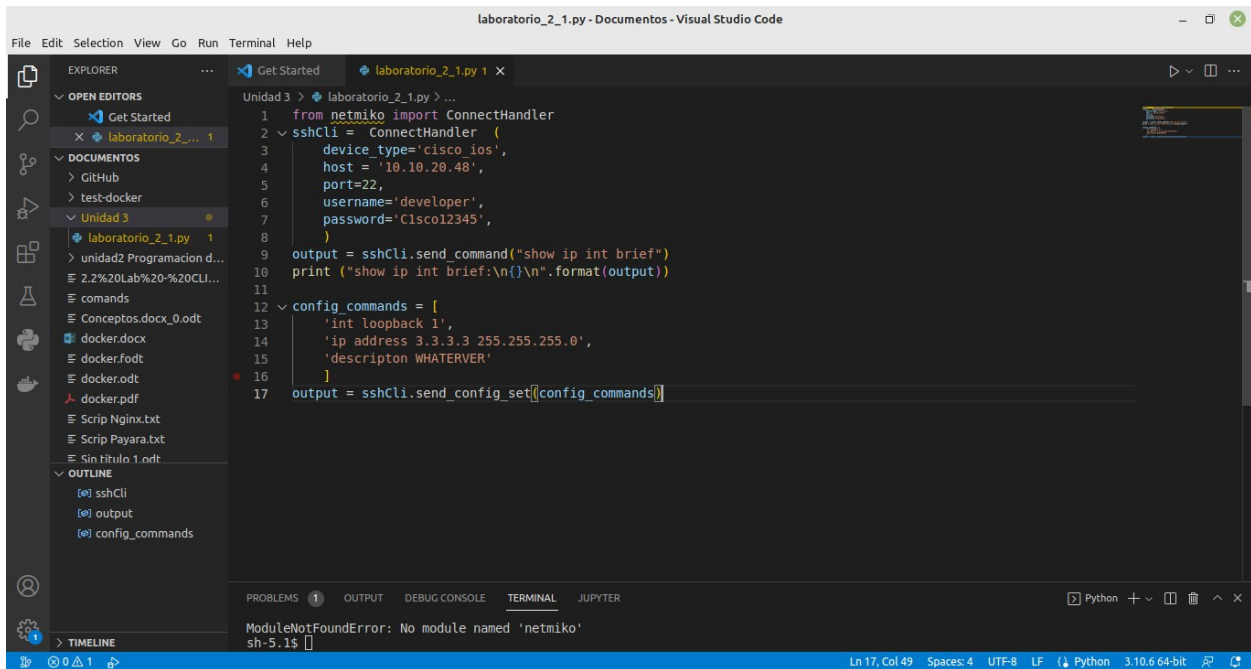
ejecutamos el comafo paa crear la interfas loopbak en el router que estamos conectados.

Why does the output from "show ip int brief" not include the "loopback1" interface?

Aqui obsrvamos como se creo correctamente la interfaz loopbak

How to execute and display the output from the "show ip int brief" command after the loopback interfaces was created?

Add code to create a new loopback interface (loopback2) with the same IP address as on the existing loopback interface, only with a different description.

Execute the Python script file and verify the results.

Was the new loopback2 interface successfully created?

Was the new configuration change accepted, partially accepted or rejected?

¿ Que es netmiko?

Netmiko es una libreria de redes multivendedores basada en Paramiko, que es una libreria estandar para las conexiones ssh Python. Con Netmiko como base, se pueden realizar programas y scripts que faciliten y mejoren la administracion de los equipos de redes. Netmiko es una biblioteca SSH Python de varios proveedores que simplifica el proceso de conexión a dispositivos de red a través de SSH. Esta biblioteca agrega lógica específica del proveedor a paramiko, que es la biblioteca SSH de facto en Python.

En conclusion el modulo netmiko sirve para administrar la red por mrdio de conexiones ssh a los dispositivos de la red. Ess un modulo muy importante ymuy utilizado para la administracion de la red.

Algunos Casos de uso de netmiko son los siguientes;

- **Copias de seguridad de la configuración** : automatice la recuperación de la salida de la configuración en ejecución de forma programada.
- **Auditorías de seguridad** : ejecute un comando para comprender si el dispositivo está ejecutando una versión de software vulnerable.
- **Automatice la solución de problemas** : automatice el proceso de ejecución de varios comandos para solucionar un problema.

Las caracteristicas Características:

- **Análisis estructurado** : admite el análisis a través de las bibliotecas de análisis TTP, TextFSM y Genie.
- **Multiproveedor** : admite un gran conjunto de dispositivos de varios proveedores.
- **Configuración de dispositivos** : proporciona métodos para aplicar la configuración desde una lista de comandos o un archivo de comandos.
- **Configuración del dispositivo** : admite varios métodos para leer la configuración de los dispositivos.
- **Ajuste de estabilidad** : admite varias opciones para garantizar la estabilidad de dispositivos lentos o transportes de red