CNIT-381

FALL 2020

NETMIKO

Low Level Interactions

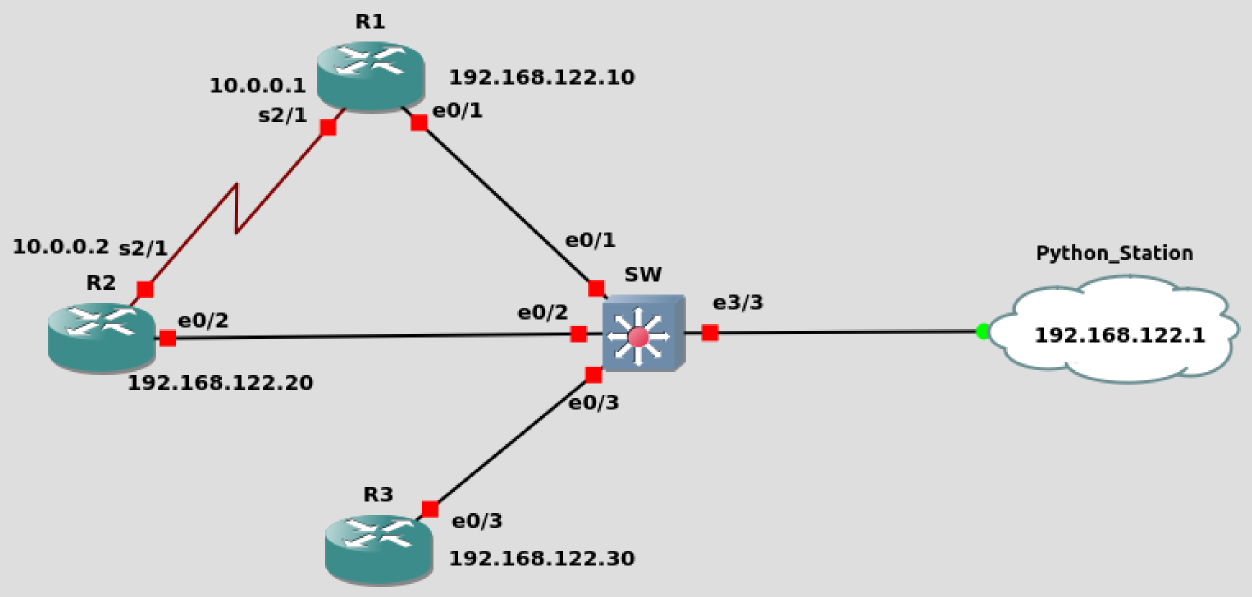
Introduction

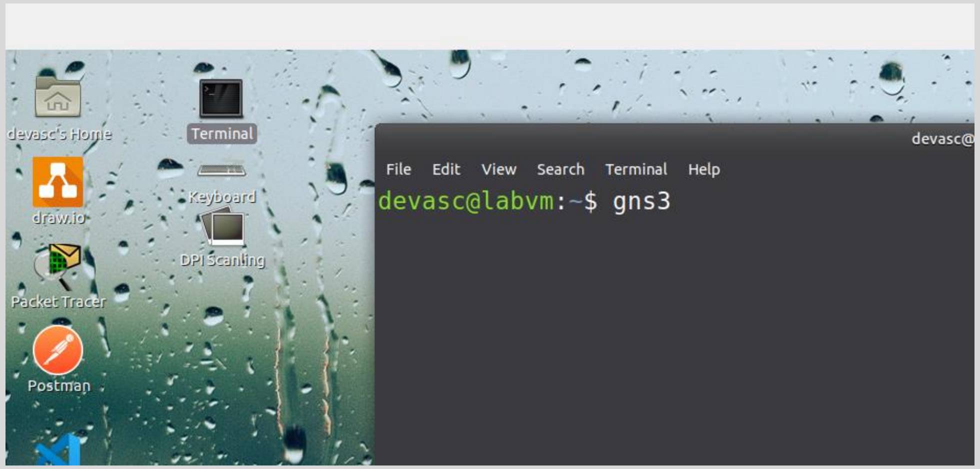
- WE SAW THAT PARAMIKO LIBRARY USED TO AUTOMATE CONFIGURATION TASKS OF NETWORKING DEVICES USING SSH.
- NETMIKO IS A NICER LIBRARY SUPPORT MULTI VENDOR NETWORK LIBRARY BASED ON PARAMIKO.
- IT RUNS ON TOP OF PARAMIKO AND IS USED TO REDUCE ITS COMPLEXITY.
- BOTH PARAMIKO AND NETMIKO ARE ALTERNATIVES TO CONFIGURE DEVICES THAT DO NOT SUPPORT APIS.
- AN API IS A STRUCTURED MODE OF SENDING AND RECEIVING STRUCTURED DATA FROM NETWORK DEVICES.

Paramiko vs. Netmiko.

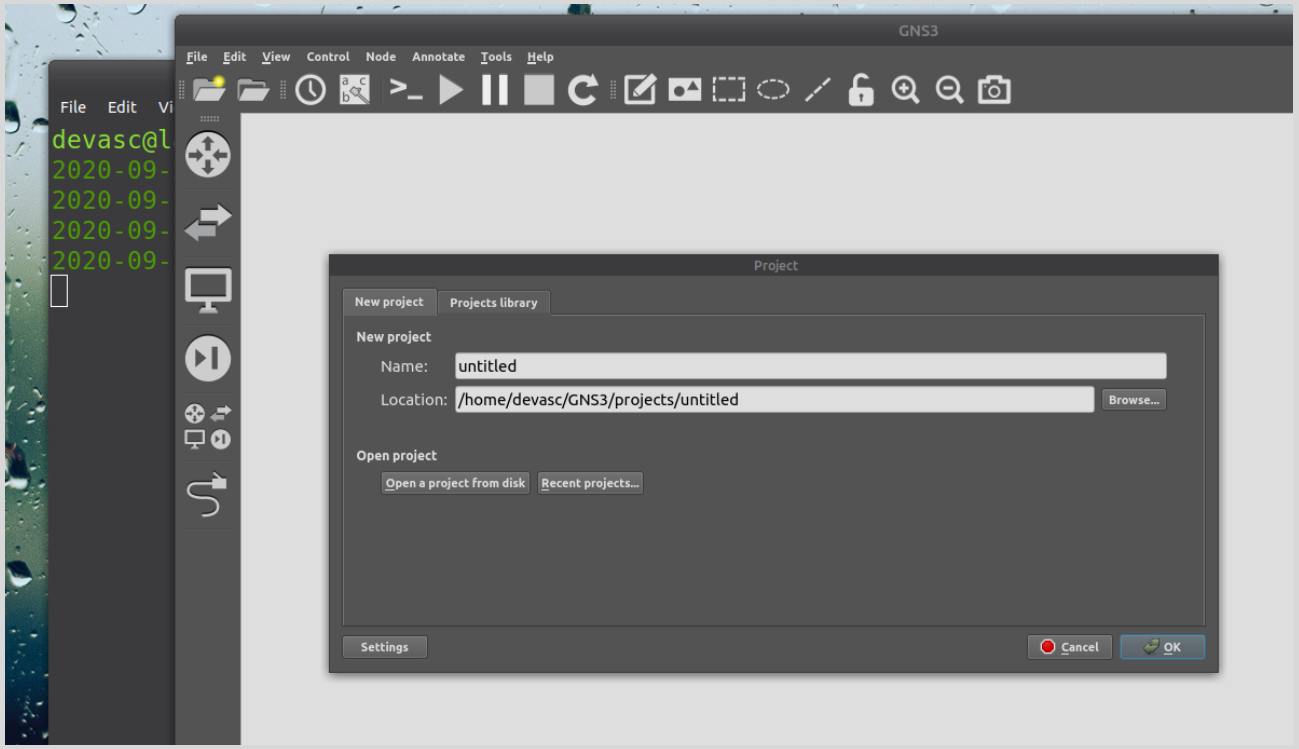
- Paramiko can be used to communicate with any device that supports ssh.
- Although Netmiko is easier to use than Paramiko, it supports only some devices.
- It supports however the most important and used vendors.
- If the platform is supported we could choose Netmiko because it's easier to handle.
- We write less code and reduce the possibility of having errors.
- If a devices that’s not being supported by Netmiko we could go ahead with Paramiko.

GNS3 Project





```
devasc@labvm:~$ gns3
```



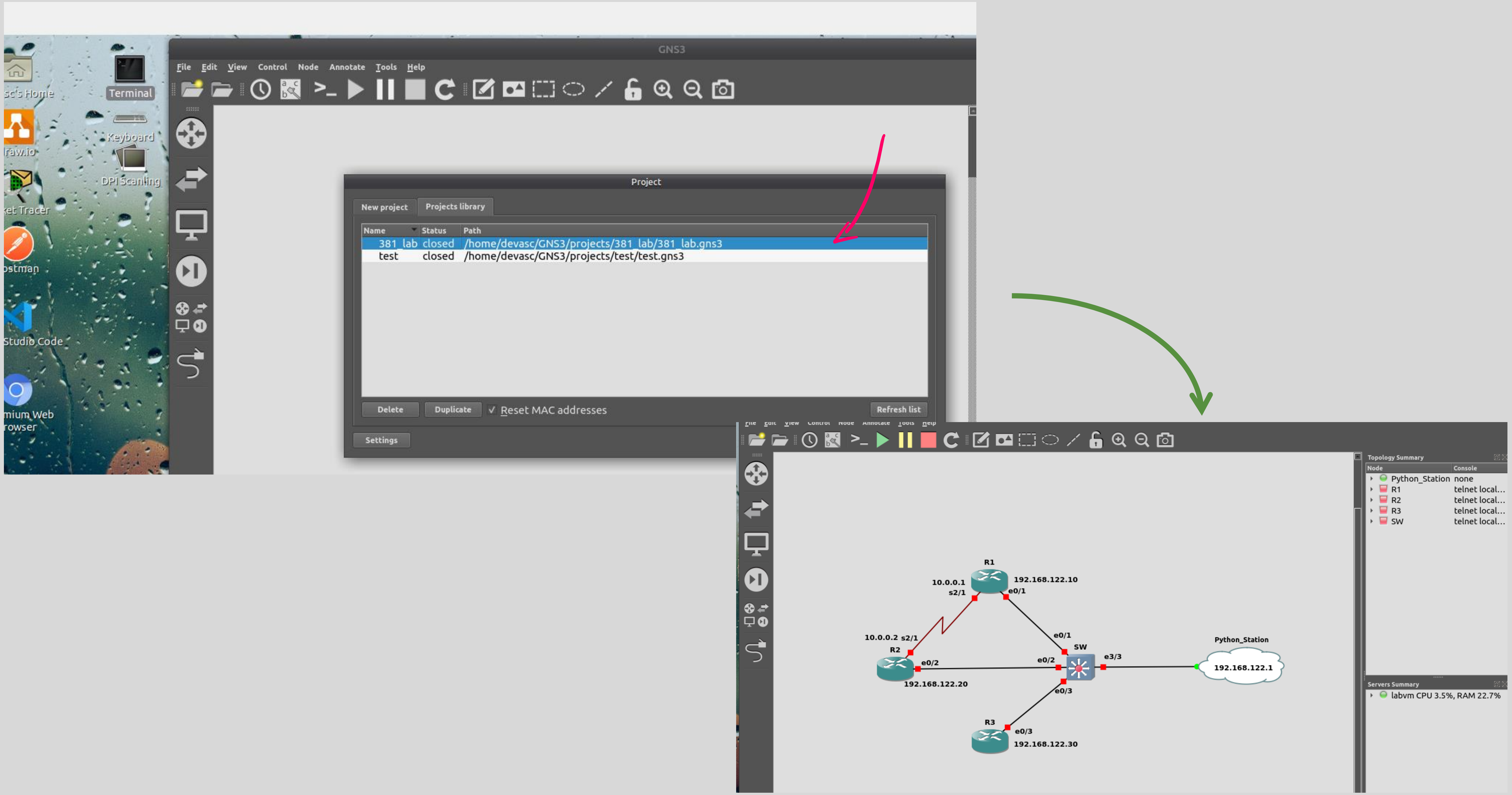
```
devasc@l
2020-09-
2020-09-
2020-09-
2020-09-
```

New project

Name: untitled

Location: /home/devasc/GNS3/projects/untitled

Open project

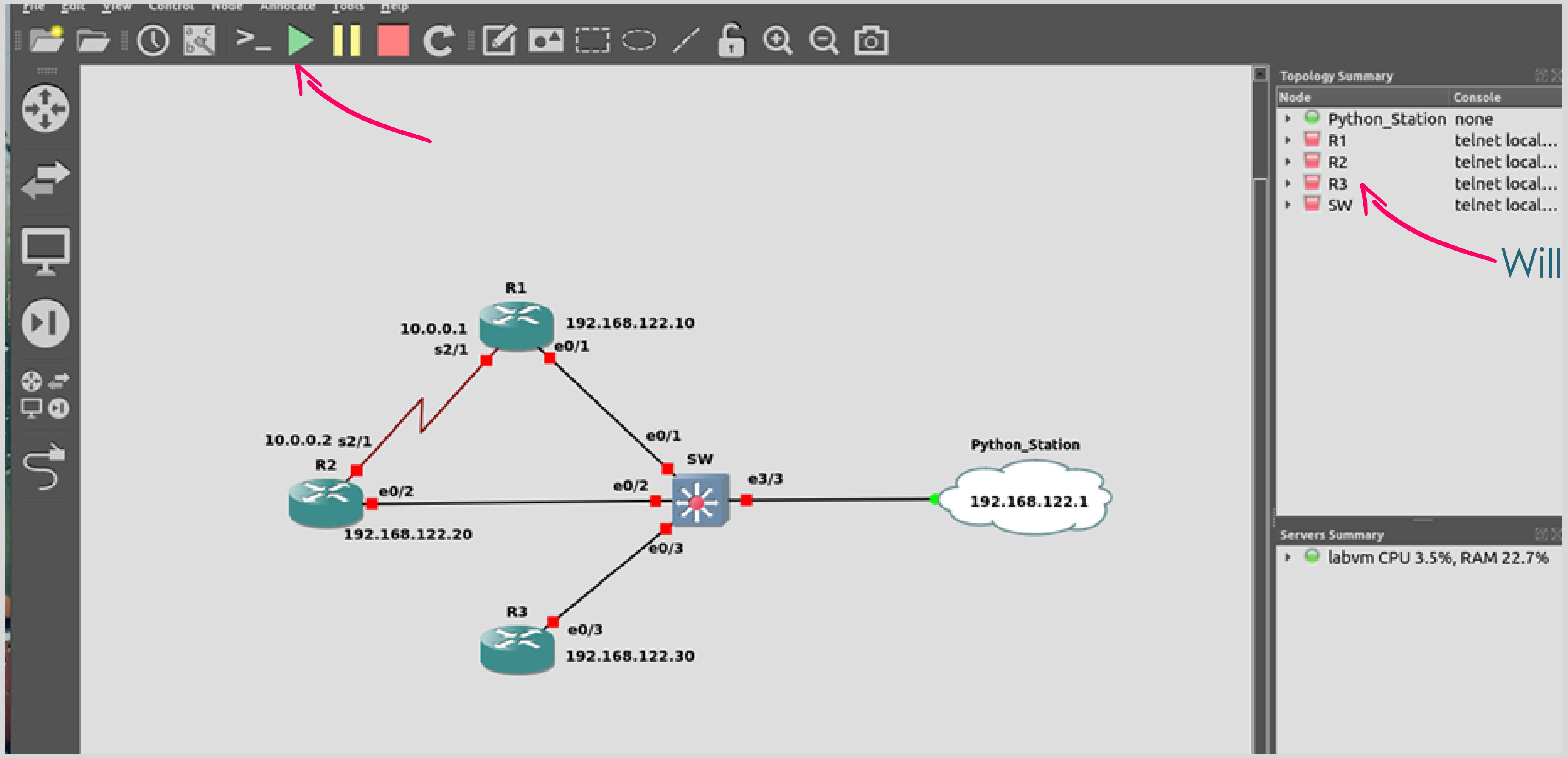


Name	Status	Path
381_lab	closed	/home/devasc/GNS3/projects/381_lab/381_lab.gns3
test	closed	/home/devasc/GNS3/projects/test/test.gns3

✓ Reset MAC addresses

Node	Console
Python_Station	none
R1	telnet local...
R2	telnet local...
R3	telnet local...
SW	telnet local...

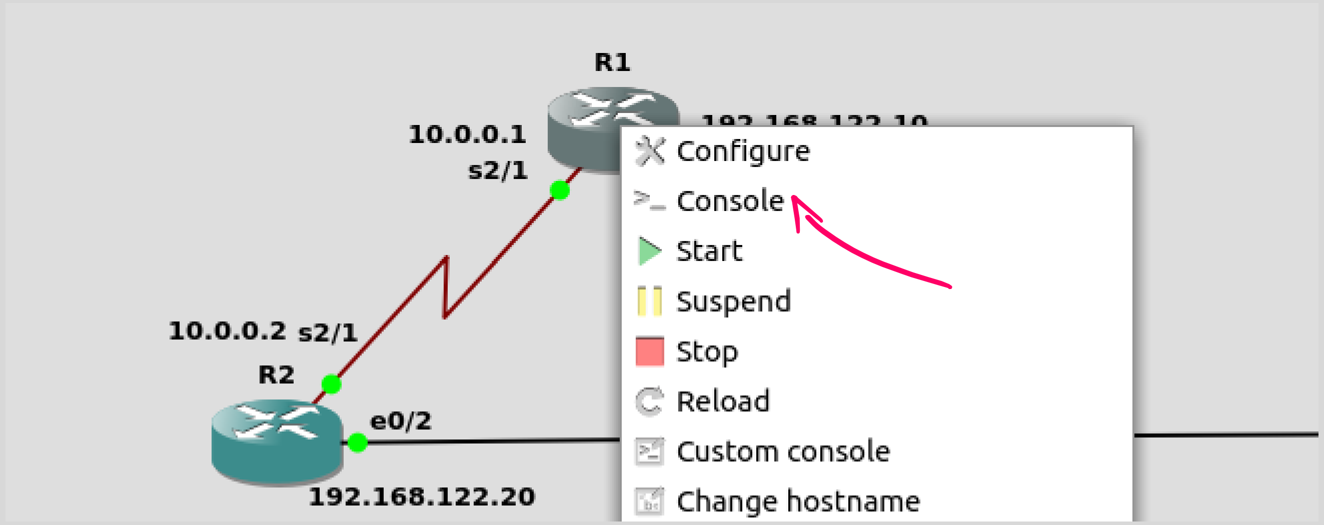
labvm CPU 3.5%, RAM 22.7%

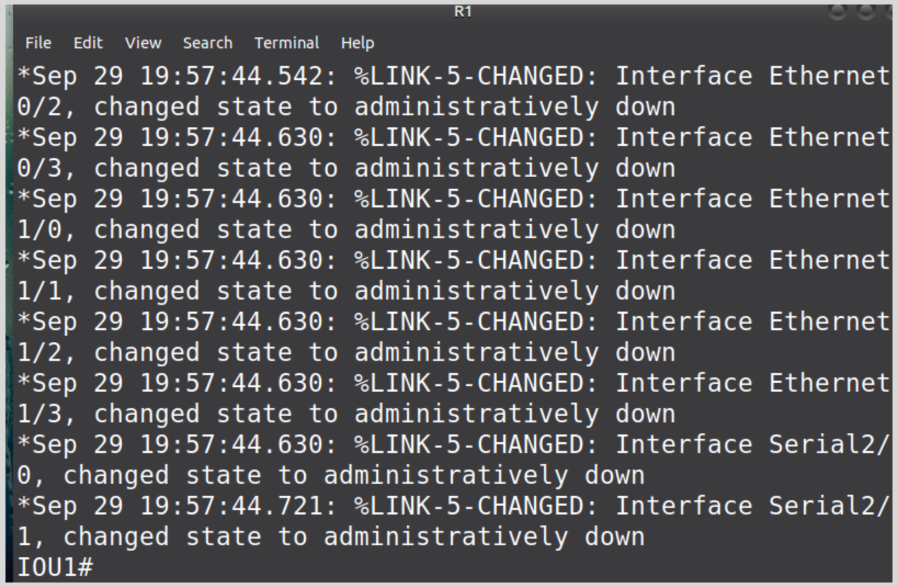


Node	Console
Python_Station	none
R1	telnet local...
R2	telnet local...
R3	telnet local...
SW	telnet local...

labvm CPU 3.5%, RAM 22.7%

Will go Green





```
*Sep 29 19:57:44.542: %LINK-5-CHANGED: Interface Ethernet
0/2, changed state to administratively down
*Sep 29 19:57:44.630: %LINK-5-CHANGED: Interface Ethernet
0/3, changed state to administratively down
*Sep 29 19:57:44.630: %LINK-5-CHANGED: Interface Ethernet
1/0, changed state to administratively down
*Sep 29 19:57:44.630: %LINK-5-CHANGED: Interface Ethernet
1/1, changed state to administratively down
*Sep 29 19:57:44.630: %LINK-5-CHANGED: Interface Ethernet
1/2, changed state to administratively down
*Sep 29 19:57:44.630: %LINK-5-CHANGED: Interface Ethernet
1/3, changed state to administratively down
*Sep 29 19:57:44.630: %LINK-5-CHANGED: Interface Serial2/
0, changed state to administratively down
*Sep 29 19:57:44.721: %LINK-5-CHANGED: Interface Serial2/
1, changed state to administratively down
IOU1#
```

Netmiko class

```
from netmiko import Netmiko

connection = Netmiko(host='192.168.122.10', port='22', username='cisco', password='cisco’,
                                        device_type='cisco_ios’)

output = connection.send_command('sh ip int brief')
print(output)

print('Closing connection')
connection.disconnect()
print('#'*40)
```

Support dev

Output

Close Conn

```
Interface          IP-Address       OK?        Method     Status                    Protocol
Ethernet0/0        unassigned       YES        NVRAM      administratively down     down
Ethernet0/1        192.168.122.10   YES        NVRAM      up                        up
.....
Closing connection
########################################
```

```
from netmiko import Netmiko

connection = Netmiko(host='192.168.122.10', port='22', username='cisco', password='cisco',
                     device_type='cisco_ios')

output = connection.send_command('sh ip int brief')
print(output)

print('Closing connection')
connection.disconnect()
print('#'*40)
```

VS

```
import paramiko
import time
ssh_client = paramiko.SSHClient() # creating an ssh client object
ssh_client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
router = {'hostname': '192.168.122.10', 'port': '22', 'username':'cicso', 'password':'cisco'}
ssh_client.connect(**router, look_for_keys=False, allow_agent=False)
print(f'Connecting to {router["hostname"]}')

shell = ssh_client.invoke_shell()
shell.send('show version\n')

time.sleep(1)
output = shell.recv(10000)
print(output)

if print(ssh_client.get_transport().is_active()) == True:
   print('Closing connection')
   ssh_client.close()
```

```
Interface          IP-Address       OK?      Method    Status                     Protocol
Ethernet0/0        unassigned       YES      NVRAM     administratively down      down
Ethernet0/1        192.168.122.10   YES      NVRAM     up                         up
.....
Closing connection
########################################
```

```
from netmiko import ConnectHandler

router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                  'secret': 'cisco', 'verbose': True}

connection = ConnectHandler(**router)
output = connection.send_command('sh run')
print(output)

print('Closing connection')
connection.disconnect()
print('#'*40)
```

$\longrightarrow$ sh run instead

```
Interactive SSH session established
^
% Invalid input detected at '^' marker.

Closing connection
########################################
```

enter enable mode

```
from netmiko import ConnectHandler

router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                'secret': 'cisco', 'verbose': True}

connection = ConnectHandler(**router)
connection.enable()
output = connection.send_command('sh run')
print(output)

print('Closing connection')
connection.disconnect()
print('#'*40)
```

```
transport input telnet ssh
!
!
end
Closing connection
########################################
```

✓

```
from netmiko import ConnectHandler

router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                    'secret': 'cisco', 'verbose': True}

connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if '>' in prompt:
    connection.enable()

output = connection.send_command('sh run')
print(output)

print('Closing connection')
connection.disconnect()
print('#'*40)
```

R1> ???!

Show current mode

```
transport input telnet ssh
!
!
end
Closing connection
########################################
```

```
 from netmiko import ConnectHandler

 router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                              'secret': 'cisco', 'verbose': True}

connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if '>' in prompt:
   connection.enable()

output = connection.send_command('userna cisco1 secret cisco')
print(output)
print('Closing connection')
connection.disconnect()
print('#'*40)
```

create username

```
Interactive SSH session established
^
% Invalid input detected at '^' marker.

Closing connection
########################################
```

```
 from netmiko import ConnectHandler

 router = {'device_type': 'cisco_ios', 'host': '10.1.1.10’, 'username': 'u1','password': 'cisco','port': 22,
                                    'secret': 'cisco', 'verbose': True}
connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if  '>' in prompt:
    connection.enable()

connection.config_mode()
output = connection.send_command(‘userna cisco1 secret cisco’)
connection.exit_config_mode()
print('Closing connection')
connection.disconnect()
print('#'*40)
```

Config

End

```
Closing connection
########################################
```

✓

```
from netmiko import ConnectHandler

router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                                   'secret': 'cisco', 'verbose': True}
connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if  '>' in prompt:
    connection.enable()

connection.config_mode()
output = connection.send_command('int lo 0')
connection.exit_config_mode()
print('Closing connection')
connection.disconnect()
print('#'*40)
```

→ Create Lo 0

nothing → freeze

```
from netmiko import ConnectHandler

router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                                'secret': 'cisco', 'verbose': True}
connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if '>' in prompt:
    connection.enable()

connection.config_mode()
output = connection.send_command('int lo 0')
connection.exit_config_mode()
print('Closing connection')
connection.disconnect()
print('#'*40)
```

ctrl-c

```
 File "/home/devasc/.local/lib/python3.8/site-packages/netmiko/base_connection.py", line 1425,
in send_command
   time.sleep(delay_factor * loop_delay)
KeyboardInterrupt
```

keep sleeping

```
from netmiko import ConnectHandler

router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                                    'secret': 'cisco', 'verbose': True}
connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if  '>' in prompt:
    connection.enable()

connection.config_mode()
output = connection.send_config_set('int lo 0', 'exit')
connection.exit_config_mode()
print('Closing connection')
connection.disconnect()
print('#'*40)
```

→ get out of int config mode

```
Closing connection
########################################
```

```
from netmiko import ConnectHandler

router = {'device_type': 'cisco_ios', 'host': '10.1.1.10', 'username': 'u1','password': 'cisco','port': 22,
                'secret': 'cisco', 'verbose': True}
connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if '>' in prompt:
    connection.enable()
if not connection.check_config_mode()
    connection.config_mode()
output = connection.send_config_set('int lo 0', 'exit')
connection.exit_config_mode()
print('Closing connection')
connection.disconnect()
print('#'*40)
```

$\longrightarrow$ not in global config !!?

```
Closing connection
########################################
```

Command Sets

```
……………..
cmd = '''int lo 0
ip add 1.1.1.1 255.255.255.255
exit
username cisco1 secret cisco
'''
if  not connection.check_config_mode():
        connection.config_mode()

connection.send_config_set(cmd.split('\n'))
connection.exit_config_mode()

print('Closing connection')
connection.disconnect()
print('#'*40)
```

→ Split by newline

```
Closing connection
########################################
```

```
router ospf  1
router-id 1.1.1.1
net 0.0.0.0 0.0.0.0 area 0
distance 80
default-information originate
```

Save to 192.168.122.10_ospf.txt

```
……………..
connection = ConnectHandler(**router)
prompt = connection.find_prompt()
if '>' in prompt:
    connection.enable()

print('Sending commands from file ...')
output = connection.send_config_from_file('192.168.122.10_ospf.txt')
print(output)

print('Closing connection')
connection.disconnect()
print('#'*40)
```

$\longrightarrow$ Read commands from file

```
R1#
Closing connection
########################################
```

```
router ospf  1
router-id 1.1.1.1
net 0.0.0.0 0.0.0.0 area 0
distance 80
default-information originate
```

Save to 192.168.122.10_ospf.txt

```
router ospf  1
router-id 2.2.2.2
net 0.0.0.0 0.0.0.0 area 0
distance 80
default-information originate
```

192.168.122.20_ospf.txt

```
router ospf  1
router-id 3.3.3.3
net 0.0.0.0 0.0.0.0 area 0
distance 80
default-information originate
```

192.168.122.30_ospf.txt

```
[{
    'device_type': 'cisco_ios',
    'host': '192.168.122.10',
    'username': 'cisco',
    'password': 'cisco',
    'port': 22,
    'secret': 'cisco',
    'verbose': True
},
{
    'device_type': 'cisco_ios',
    'host': '192.168.122.20',
    'username': 'cisco',
    'password': 'cisco',
    'port': 22,
    'secret': 'cisco',
    'verbose': True
},
{
    'device_type': 'cisco_ios',
    'host': '192.168.122.30',
    'username': 'cisco',
    'password': 'cisco',
    'port': 22,
    'secret': 'cisco',
    'verbose': True
}]
```

Save to routers.txt

```
from netmiko import ConnectHandler
import myNewParamiko as m

routers = m.get_list_from_file ('routers.txt')
for router in routers:
    connection = ConnectHandler(**router)
    prompt = connection.find_prompt()
    if '>' in prompt:
        connection.enable()
    print('Sending commands from file ...')
    output = connection.send_config_from_file(router['host']+'_ospf.txt')
    print(output)
    print('Closing connection')
    connection.disconnect()
    print('#'*40)
```

192.168.122.xx-ospf.txt

```
R3#
Closing connection
########################################
```

```
from netmiko import ConnectHandler
import myNewParamiko as m
import time
start = time.time()
routers = m.get_list_from_file ('routers.txt')
for router in routers:
    connection = ConnectHandler(**router)
    prompt = connection.find_prompt()
    if '>' in prompt:
        connection.enable()
    print('Sending commands from file ...')
    output = connection.send_config_from_file(router['host']+'_ospf.txt')
    print(output)
    print('Closing connection')
    connection.disconnect()
    print('#'*40)
end = time.time()
print(f'Total execution time:{end-start}')
```

time before run

time after run

accumulation time

```
R3#
Closing connection
########################################
Total execution time:28.538666725158 69
```

~ 9 x 3 seconds

```
from netmiko import ConnectHandler
import myNewParamiko as m
import time
import threading

start = time.time()
def config_ospf(router):
    connection = ConnectHandler(**router)
    prompt = connection.find_prompt()
    if '>' in prompt:
        connection.enable()
    print('Sending commands from file ...')
    output = connection.send_config_from_file(router['host']+'_ospf.txt')
    print(output)
    print('Closing connection')
    connection.disconnect()
    print('#'*40)

routers = m.get_list_from_file ('routers.txt')
threads = list()
for router in routers:
    th = threading.Thread(target=config_ospf, args=(router,))
    threads.append(th)

for th in threads:
    th.start()

for th in threads:
    th.join()

end = time.time()
print(f'Total execution time:{end-start}')
```

make a function

task(,)

Split to 3 threads

```
R2(config-router)#end
R2#
Closing connection
###########################################
###########################################
###########################################
Total execution time:9.560632228851318
```

VS 29s