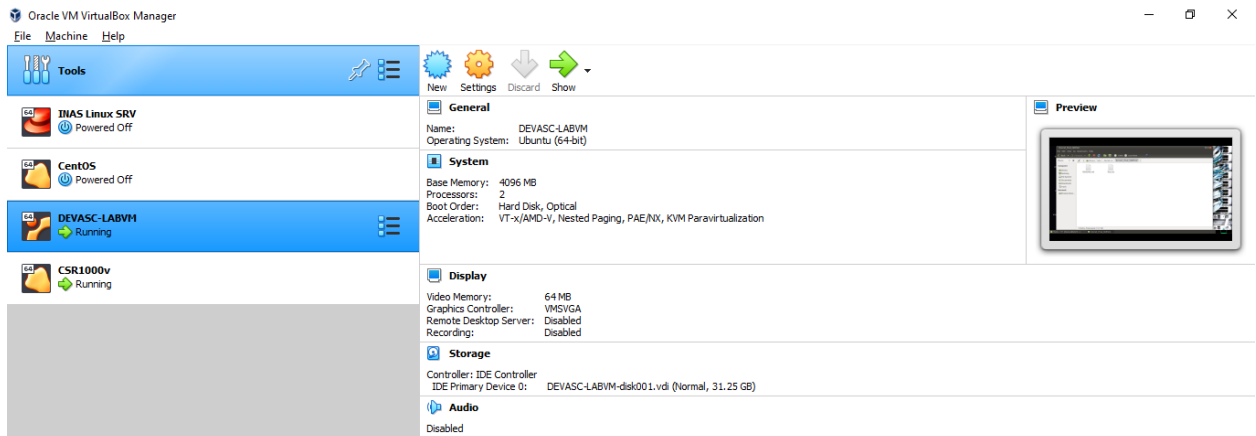


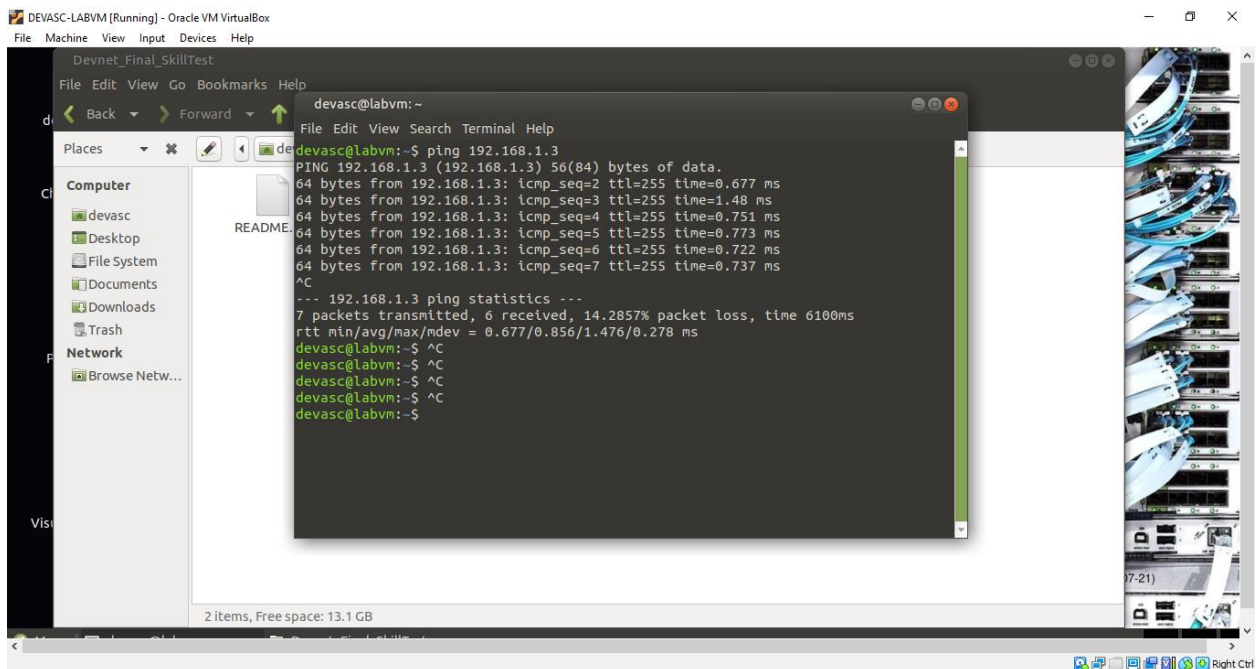
DEVNET SKILL EXAM - 2022

Tho Luong Duc

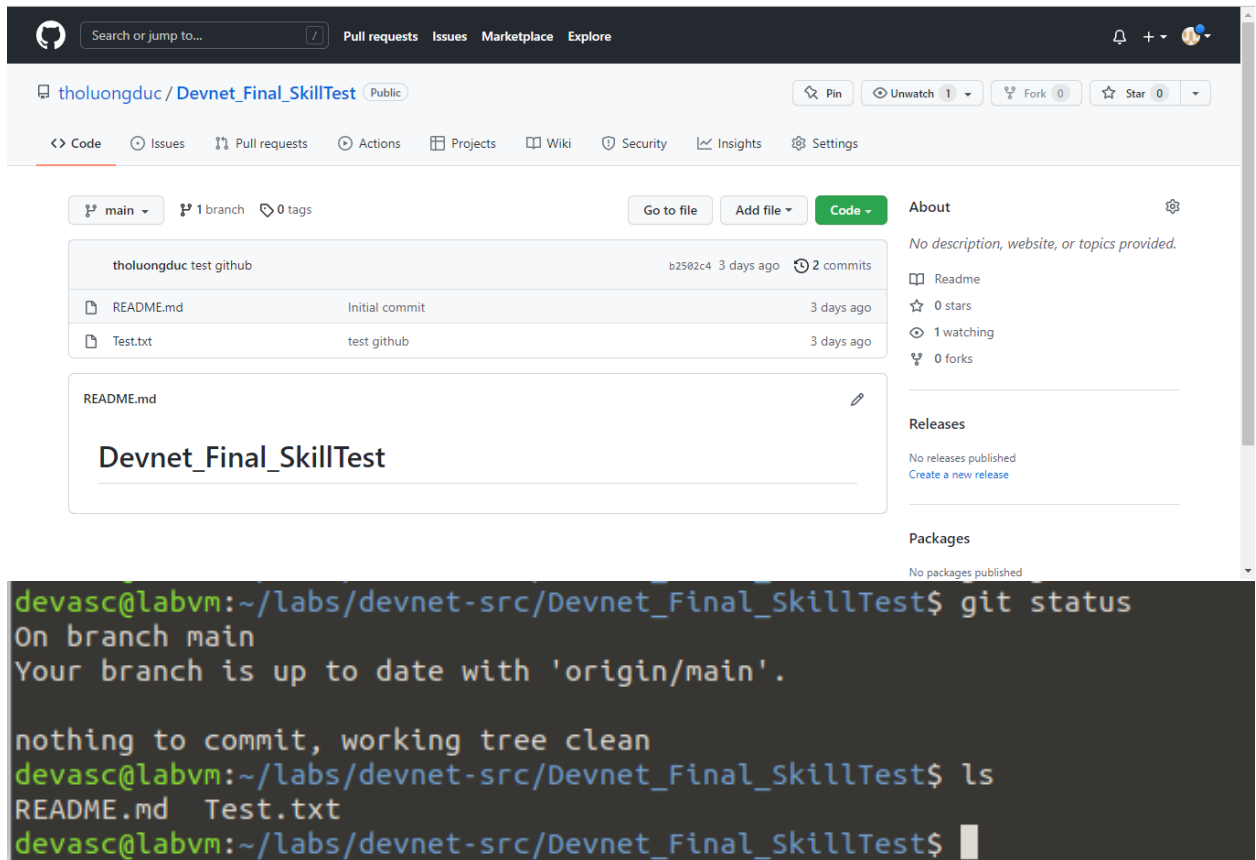
1. Open CSR1000v router on VirtualBox



Check connection to router: ping successful to router's IP address 192.168.1.3.



2. Create a repository on Github and clone on local machine

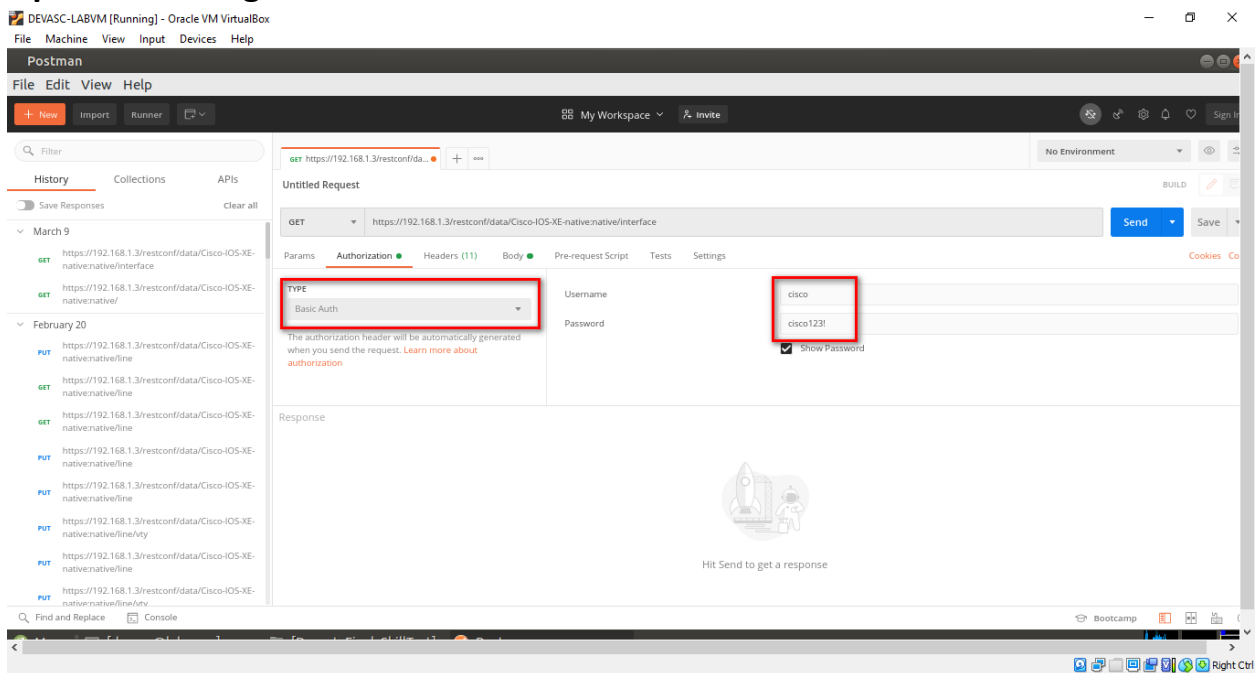


The top part of the image shows a GitHub repository page for 'tholuongduc / Devnet_Final_SkillTest'. The repository is public and has 1 branch (main) and 0 tags. It contains two files: README.md (Initial commit, 3 days ago) and Test.txt (test github, 3 days ago). The README.md file is displayed with the text 'Devnet_Final_SkillTest'. The right sidebar shows the repository's statistics: 0 stars, 1 watching, and 0 forks. Below the repository page, a terminal window shows the following commands and output:

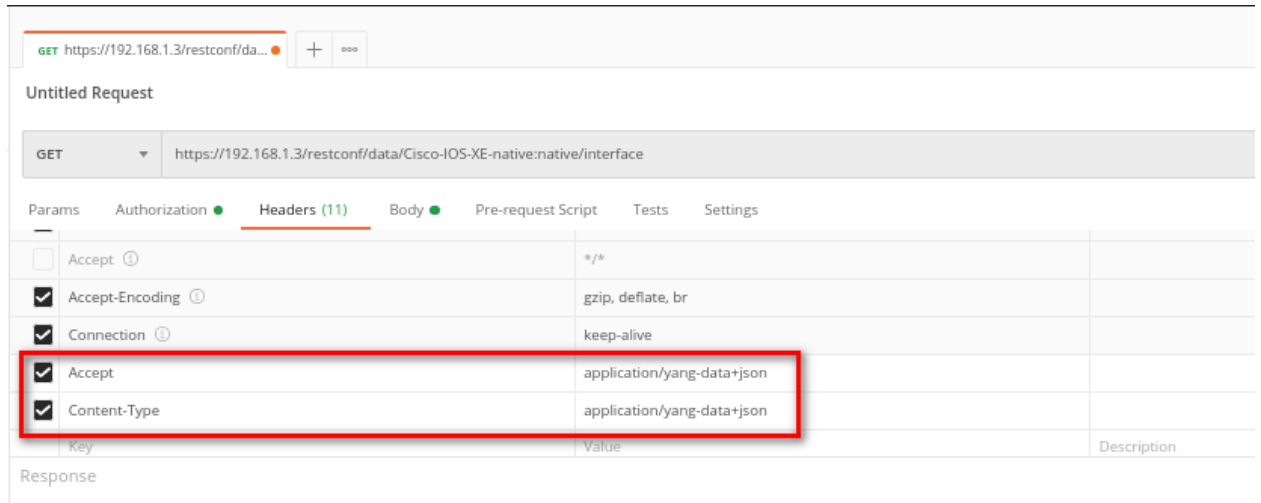
```
devasc@labvm:~/labs/devnet-src/Devnet_Final_SkillTest$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
devasc@labvm:~/labs/devnet-src/Devnet_Final_SkillTest$ ls
README.md  Test.txt
devasc@labvm:~/labs/devnet-src/Devnet_Final_SkillTest$
```

3. Open and configure POSTMAN

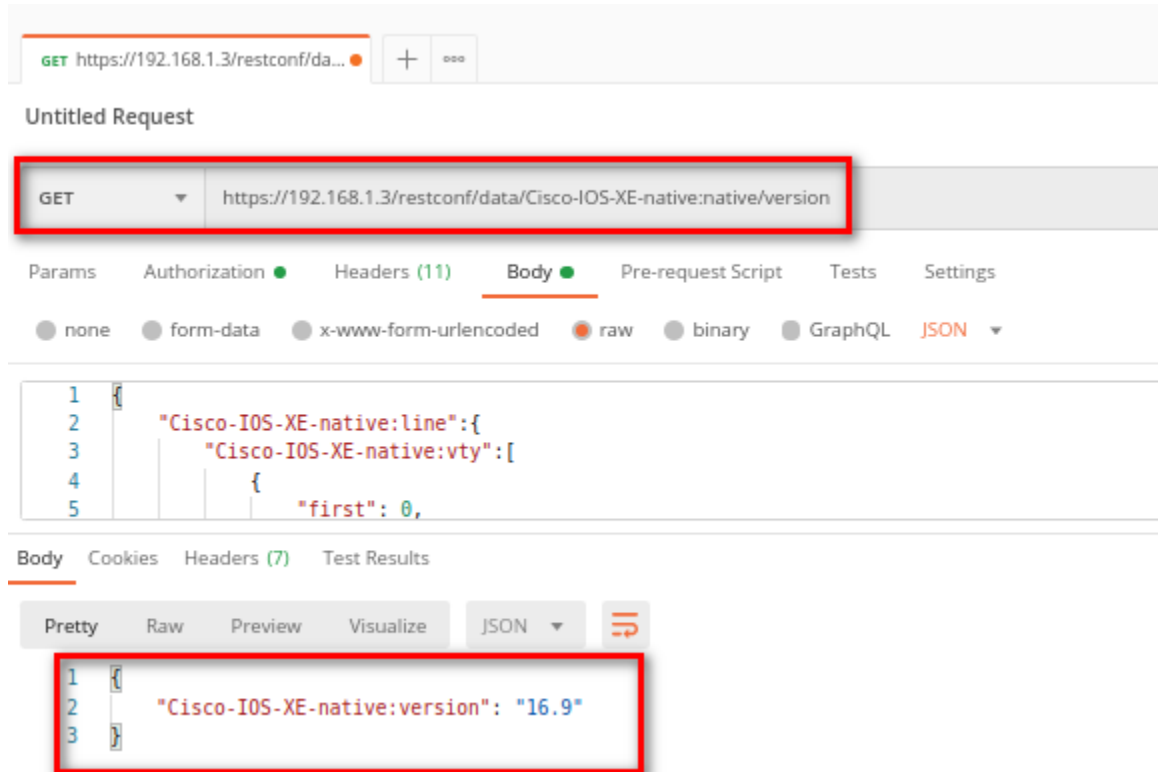


The bottom part of the image shows the Postman application interface. The 'Untitled Request' tab is active, showing a GET request to 'https://192.168.1.3/restconf/data/Cisco-IOS-XE-native:interface'. The 'Authorization' tab is selected, and the 'Basic Auth' type is chosen. The 'Username' field is set to 'cisco' and the 'Password' field is set to 'cisco123!'. The 'Show Password' checkbox is checked. The 'Response' tab is empty, and a message at the bottom says 'Hit Send to get a response'.



4. Activities using POSTMAN

- Get device information
 - Version of IOS



- Routing Information

GET https://192.168.1.3/restconf/data/Cisco-IOS-XE-native:router

Status: 200 OK Time: 55 ms Size: 954 B Save Response

```
1 {
2   "Cisco-IOS-XE-native:router": {
3     "Cisco-IOS-XE-ospf:ospf": [
4       {
5         "id": 1,
6         "network": [
7           {
8             "ip": "10.1.1.0",
9             "mask": "0.0.0.255",
10            "area": 10
11          },
12          {
13            "ip": "10.2.1.0",
14            "mask": "0.0.0.255",
15            "area": 20
16          }
17        ]
18      }
19    ]
20  }
```

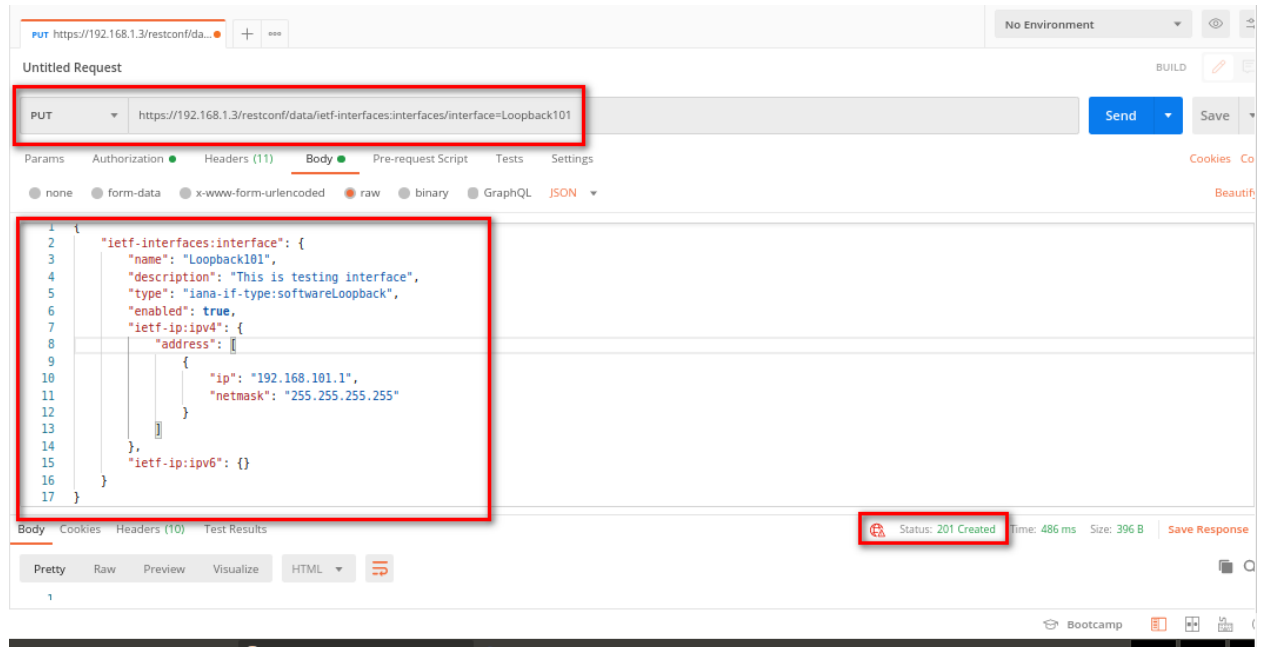
- Interface Information

GET https://192.168.1.3/restconf/data/Cisco-IOS-XE-native:interface

Status: 200 OK Time: 103 ms Size: 978 B Save Response

```
1 {
2   "Cisco-IOS-XE-native:interface": {
3     "GigabitEthernet": [
4       {
5         "name": "1",
6         "description": "VBox",
7         "ip": {
8           "address": {
9             "primary": {
10              "address": "192.168.1.3",
11              "mask": "255.255.255.0"
12            }
13          }
14        },
15        "mop": {
16          "enabled": false,
17          "sysid": false
18        }
19      }
20    ]
21  }
```

- Configure some parameters:
 - Create interface loopback 101



5. Configure device for RESTCONF Access

- Use python script to send GET request:

```
Python_script_GET_request.py X
home > devasc > labs > devnet-src > Devnet_Final_SkillTest > Python_script_GET_request.py
1 import json
2 import requests
3 import sys
4
5 requests.packages.urllib3.disable_warnings()
6
7 headers = { "Accept": "application/yang-data+json",
8             "Content-type": "application/yang-data+json"
9           }
10 basicauth = ("cisco", "cisco123!")
11 sys.stdout = open("Python_Get_Request_Result.txt", "w")
12 #Get IOS Version
13 api_url0 = "https://192.168.1.3/restconf/data/Cisco-IOS-XE-native:native/version"
14 resp0 = requests.get(api_url0, auth=basicauth, headers=headers, verify=False)
15 response0_json = resp0.json()
16 print(f'IOS Version: \n {response0_json}')
17 #Get Routing Information
18 api_url1 = "https://192.168.1.3/restconf/data/Cisco-IOS-XE-native:native/router"
19 resp1 = requests.get(api_url1, auth=basicauth, headers=headers, verify=False)
20 response1_json = resp1.json()
21 print(f'Routing Information: \n {response1_json}')
22 #Get Interfaces Information
23 api_url2 = "https://192.168.1.3/restconf/data/Cisco-IOS-XE-native:native/interface"
24 resp2 = requests.get(api_url2, auth=basicauth, headers=headers, verify=False)
25 response2_json = resp2.json()
26 print(f'Interface Information: \n {response2_json}')
27 sys.stdout.close()
```

- Use a python script to send PUT request:

```

Python_script_PUT_Request.py X
home > devasc > labs > devnet-src > Devnet_Final_SkillTest > Python_script_PUT_Request.py
1  import json
2  import requests
3  import sys
4
5  requests.packages.urllib3.disable_warnings()
6
7  headers = { "Accept": "application/yang-data+json",
8             "Content-type": "application/yang-data+json"
9             }
10 basicauth = ("cisco", "cisco123!")
11 sys.stdout = open("Python_PUT_Request_Result.txt", "w")
12
13 api_url = "https://192.168.1.3/restconf/data/ietf-interfaces:interfaces/interface=Loopback102"
14
15 sys.stdout = open("Python_PUT_Request_Result.txt", "w")
16 #Create interface Loopback 102
17 yangConfig = {
18     "ietf-interfaces:interface": {
19         "name": "Loopback102",
20         "description": "This is created by Python script",
21         "type": "iana-if-type:softwareLoopback",
22         "enabled": True,
23         "ietf-ip:ipv4": {
24             "address": [
25                 {
26                     "ip": "192.168.102.1",
27                     "netmask": "255.255.255.255"
28                 }
29             ]
30         },
31         "ietf-ip:ipv6": {}
32     }
33 }
34 resp = requests.put(api_url, data=json.dumps(yangConfig), auth=basicauth, headers=headers, verify=False)
35
36 if(resp.status_code >= 200 and resp.status_code <= 299):

```

6. Upload all files to Github

Github link: https://github.com/tholuongduc/Devnet_Final_SkillTest.git

```

devasc@labvm:~/labs/devnet-src/Devnet_Final_SkillTest$ git add *
devasc@labvm:~/labs/devnet-src/Devnet_Final_SkillTest$ git commit -m "Devnet Skill Exam 2022"
[main 4772ab9] Devnet Skill Exam 2022
5 files changed, 166 insertions(+)
create mode 100644 POSTMAN_Get_Result.txt
create mode 100644 Python_Get_Request_Result.txt
create mode 100644 Python_PUT_Request_Result.txt
create mode 100644 Python_script_GET_request.py
create mode 100644 Python_script_PUT_Request.py
devasc@labvm:~/labs/devnet-src/Devnet_Final_SkillTest$ git push origin main
Username for 'https://github.com': tholuongduc
Password for 'https://tholuongduc@github.com':
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 2 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 2.58 KiB | 2.58 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To https://github.com/tholuongduc/Devnet_Final_SkillTest.git
b2502c4..4772ab9 main -> main
devasc@labvm:~/labs/devnet-src/Devnet_Final_SkillTest$

```