

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б / -2021
	Екземпляр № 1	Арк 11 / 1

Лабораторна робота № 7. Використання NETCONF для доступу до пристрою IOS XE

Метою даної лабораторної роботи є отримання базових навичок по роботі з NETCONF.

Завдання на лабораторну роботу

- Побудувати мережу і встановити з'єднання
- Використати сеанс NETCONF для збирання інформації
- Використати ncclient для під'єднання до NETCONF
- Використати ncclient для отримання конфігурації
- Використати ncclient для налаштування пристрою
- Використати Ansible для налаштування пристрою

Хід роботи:

Протокол мережної конфігурації (NETCONF), визначений у документах RFC 4741 та 6241, використовує моделі даних YANG для зв'язку з різними пристроями в мережі. YANG - це мова моделювання даних. Ця мова визначає дані, які надсилаються через протоколи керування мережею, наприклад NETCONF. При використанні NETCONF для доступу до пристрою IOS XE дані повертаються у форматі XML.

В даній лабораторній роботі ви будете використовувати клієнт NETCONF, ncclient, який являє собою модуль Python для написання скриптів на стороні клієнта. Ви будете використовувати ncclient для того, щоб перевірити, чи налаштований NETCONF, отримати конфігурацію пристрою, а також змінити конфігурацію пристрою.

1. Побудувати мережу та перевірити під'єднання.

У цій частині ви запускаєте дві віртуальні машини та перевіряєте під'єднання. Потім ви встановіть з'єднання через захищену оболонку (SSH).

1.1 Запустіть віртуальні машини

Запустіть віртуальні машини DEVASC та CSR1000v

1.2. Перевірте з'єднання між віртуальними машинами.

а. У віртуальній машині CSR1kv натисніть «Enter», щоб отримати командний рядок, а потім використайте **show ip interface brief**, щоб переконатися, що IPv4-адреса - 192.168.56.101. Якщо адреса є іншою, тоді запишіть її.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 2

б. Відкрийте термінал у VS Code на віртуальній машині DEVASC.

с. Використайте команду ping для перевірки з'єднання CSR1kv.

```
devasc@labvm:~$ ping -c 5 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=254 time=1.37 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=254 time=1.15 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=254 time=0.981 ms
64 bytes from 192.168.56.101: icmp_seq=4 ttl=254 time=1.01 ms
64 bytes from 192.168.56.101: icmp_seq=5 ttl=254 time=1.14 ms

--- 192.168.56.101 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 0.981/1.130/1.365/0.135 ms
devasc@labvm:~$
```

1.3. Перевірте під'єднання через SSH до VM CSR1kv.

а. В терміналі для VM DEVASC через SSH під'єднайтеся до VM CSR1kv за допомогою наступної команди:

```
devasc@labvm:~$ ssh cisco@192.168.56.101
```

Примітка: Під час першого під'єднання SSH до CSR1kv ваша VM DEVASC попереджає вас про справжність CSR1kv. Оскільки ви підтверджуєте справжність CSR1kv, тому на підказку відповідайте - «Так».

```
The authenticity of host '192.168.56.101 (192.168.56.101)' can't be
established.
RSA key fingerprint is SHA256:HYv9K5Biw7PFiXeoCDO/LTqs3EfZKBuJdiPo34VXDUY.
Are you sure you want to continue connecting (yes/no/[fingerprint])?yes
Warning: Permanently added '192.168.56.101' (RSA) to the list of known hosts.
```

б. Введіть **cisco123!** як пароль, і тепер ви повинні знаходитися в привілейованому командному рядку EXEC для CSR1kv.

```
Password:
```

```
CSR1kv#
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 3

с. Залиште сеанс SSH відкритим для наступної частини лабораторної роботи.

2. Використання сеансу NETCONF для збирання інформації

2.1. Перевірте, чи працює NETCONF на CSR1kv.

а. NETCONF вже може бути запущеним, якщо інший студент увімкнув його, або, якщо пізніша версія IOS вмикає його за замовчуванням. В сеансі SSH з CSR1kv, використовуйте команду **show platform software yang-management process**, щоб побачити, чи запущена фонові служба SSH NETCONF (**ncsshd**).

```
CSR1kv# show platform software yang-management process
confd          : Running
nesd           : Running
syncfd        : Running
ncsshd        : Running
dmiauthd      : Running
nginx         : Running
ndbmand       : Running
pubd          : Running

CSR1kv#
```

б. Якщо NETCONF не запущений, як це відображено вище у виведених даних, тоді введіть глобальну команду конфігурації **netconf-yang**.

```
CSR1kv# config t
CSR1kv (config)# netconf-yang
```

с. Введіть **exit**, для того, щоб завершити сеанс SSH.

2.2: Отримайте доступ до процесу NETCONF через термінал SSH.

На даному етапі ви маєте відновити сеанс SSH за допомогою CSR1kv. Але цього разу ви вкажете порт 830 NETCONF і надішлете **netconf** як команду підсистеми.

Примітка: Для отримання додаткової інформації про ці параметри, перегляньте керівництво для SSH (**man ssh**).

а. Введіть наступну команду у вікні терміналу. Ви можете використовувати стрілку вгору для виклику останньої команди SSH, і просто додати параметри **-p** і **-s**, як показано. Потім введіть **cisco123!** в якості паролю.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 4

```
devasc@labvm:~$ ssh cisco@192.168.56.101 -p 830 -s netconf
cisco@192.168.56.101's password:
```

b. CSR1kv відповідь повідомленням - **привіт (hello)**, яке включає понад 400 рядків вихідних даних із переліком усіх можливостей NETCONF. Кінець повідомлень NETCONF позначається `]]>]]>`.

```
<?xml version="1.0" encoding="UTF-8"?>
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
    <capability>urn:ietf:params:netconf:base:1.1</capability>
    <capability>urn:ietf:params:netconf:capability:writable-running:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:xpath:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:validate:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:validate:1.1</capability>
    (output omitted)
  </capabilities>
</capabilities>
<session-id>20</session-id></hello>]]>]]>
```

2.3. Розпочніть сеанс NETCONF, надіславши привітальне повідомлення від клієнта.

Для того, щоб розпочати сеанс NETCONF, клієнту необхідно надіслати своє власне привітальне повідомлення. Привітальне повідомлення має містити версію базових можливостей NETCONF, яку клієнт хоче використовувати.

a. Скопіюйте та вставте наступний XML-код в сеанс SSH. Зверніть увагу, що кінець повідомлень NETCONF клієнту позначається `]]>]]>`.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
  </capabilities>
</hello>
]]>]]>
```

b. Перейдіть на VM CSR1kv і використайте команду **show netconf-yang sessions** для того, щоб переконатися, що сеанс NETCONF запущений. Якщо екран віртуальної машини CSR1kv темний, натисніть клавішу **Enter** для його пробудження.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 5

```
CSR1kv> en
```

```
CSR1kv# show netconf-yang sessions
```

```
R: Global-lock on running datastore
```

```
C: Global-lock on candidate datastore
```

```
S: Global-lock on startup datastore
```

```
Number of sessions : 1
```

```

session-id      transport      username      source-host      global-lock
-----
20              netconf-ssh   cisco         192.168.56.1     None

```

```
CSR1kv#
```

2.4. Надішліть повідомлення RPC на пристрій IOS XE.

Під час сеансу SSH клієнт NETCONF може використовувати повідомлення Віддаленого виклику процедур (Remote Procedure Call - RPC) для надсилання операцій NETCONF на пристрій IOS XE. В таблиці наведені деякі найпоширеніші операції NETCONF.

Операція	Опис
<get>	Отримати поточну конфігурації та інформацію про стан пристрою
<get-config>	Отримати весь або частину вказаного сховища даних конфігурації
<edit-config>	Завантажити всю або частину конфігурації до вказаного сховища даних конфігурації
<copy-config>	Замінити все сховища даних конфігурації на інше
<delete-config>	Видалити сховище даних конфігурації
<commit>	Скопіювати сховище даних кандидатів до запущеного сховища даних
<lock> / <unlock>	Блокувати або розблокувати всю систему сховища даних конфігурації
<close-session>	Коректно завершувати сеанс NETCONF
<kill-session>	Примусово припиняти сеанс NETCONF

а. Скопіюйте та вставте наступне XML-код повідомлення RPC **get** в сеанс SSH терміналу для того, щоб отримати інформацію про інтерфейси на R1.

```

<rpc message-id="103" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter>
      <interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces"/>
    </filter>
  </get>
</rpc>
]]>]]>

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 6

б. XML не вимагає відступів або пробілів. Таким чином, CSR1kv поверне довгий рядок XML-даних.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-
id="103"><data><interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-
interfaces"><interface><name>GigabitEthernet1</name><description>VBox</description><ty
pe xmlns:ianaift="urn:ietf:params:xml:ns:yang:iana-if-
type">ianaift:ethernetCsmacd</type><enabled>true</enabled><ipv4
xmlns="urn:ietf:params:xml:ns:yang:ietf-ip"></ipv4><ipv6
xmlns="urn:ietf:params:xml:ns:yang:ietf-
ip"></ipv6></interface></interfaces></data></rpc-reply>]]>]]>
```

с. Скопіюйте повернутий XML, за виключенням останніх символів «]]>]]>». Ці символи не є частиною XML, який повертає маршрутизатор.

д. Виконайте пошук в Інтернеті для «prettify XML». Знайдіть відповідний сайт і скористайтеся його інструментом, щоб перетворити ваш XML у більш зручний для читання формат, наприклад:

```
<?xml version="1.0"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="103">
  <data>
    <interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
      <interface>
        <name>GigabitEthernet1</name>
        <description>VBox</description>
        <type xmlns:ianaift="urn:ietf:params:xml:ns:yang:iana-if-
type">ianaift:ethernetCsmacd</type>
        <enabled>true</enabled>
        <ipv4 xmlns="urn:ietf:params:xml:ns:yang:ietf-ip"/>
        <ipv6 xmlns="urn:ietf:params:xml:ns:yang:ietf-ip"/>
      </interface>
    </interfaces>
  </data>
</rpc-reply>
```

2.5. Закрийте сеанс NETCONF.

а. Щоб закрити сеанс NETCONF, клієнту потрібно надіслати наступне повідомлення RPC:

```
<rpc message-id="9999999" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <close-session />
</rpc>
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 7

б. Через кілька секунд ви повернетесь до підказки терміналу. Поверніться до запиту `CSR1kv` і покажіть відкриті сеанси `netconf`. Ви побачите, що сеанс був закритий.

```
CSR1kv# show netconf-yang sessions
```

```
There are no active sessions
```

```
CSR1kv#
```

3: Використання `ncclient` для під'єднання до NETCONF

Робота з NETCONF не вимагає роботи з необробленими повідомленнями NETCONF RPC та XML. В цій частині ви дізнаєтесь, як використовувати модуль `ncclient` Python для легкої взаємодії з мережними пристроями за допомогою NETCONF. Крім того, ви дізнаєтесь, як визначити, які моделі YANG підтримуються пристроєм. Дана інформація корисна при побудові системи автоматизації виробничої мережі, яка вимагає підтримки конкретних моделей YANG конкретним мережним пристроєм.

3.1. Переконайтеся, що `ncclient` встановлений і готовий до використання.

В терміналі DEVASC-VM введіть команду `pip3 list --format=column` для того, щоб побачити всі встановлені на даний момент модулі Python. Перенаправте (pipe) вивід команди до більшого (**more**). Ваш результат може відрізнятись від наведеного нижче. Але ви повинні побачити `ncclient` у списку, як це відображено. Якщо цього немає, скористайтесь командою `pip3 install ncclient` для її встановлення.

```
devasc@labvm:~$ pip3 list --format=columns | more
```

```
Package              Version
-----
ansible              2.9.6
apache-libcloud      2.8.0
appdirs              1.4.3
argcomplete          1.8.1
astroid               2.3.3
(output omitted)
ncclient              0.6.7
netaddr               0.7.19
netifaces            0.10.4
netmiko               3.1.0
ntlm-auth             1.1.0
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 8

```
oauthlib          3.1.0
(output omitted)
xmltodict         0.12.0
zip              1.0.0
devasc@labvm:~$
```

3.2. Створіть скрипт для використання `ncclient` для під'єднання до служби **NETCONF**.

Модуль `ncclient` надає клас менеджера з методом `connect()` для налаштування віддалених з'єднань **NETCONF**. Після успішного під'єднання повернутий об'єкт являє собою з'єднання **NETCONF** з віддаленим пристроєм.

a. В VS code послідовно виберіть елементи **Файл (File) > Відкрити папку... (Open Folder...)** і перейдіть до каталогу **devnet-src**. Натисніть **ОК**.

b. Відкрийте вікно терміналу в VS Code: **Термінал (Terminal) > Новий термінал (New Terminal)**.

c. Створіть підкаталог з назвою **netconf** в каталозі **/devnet-src**.

```
devasc@labvm:~/labs/devnet-src$ mkdir netconf
devasc@labvm:~/labs/devnet-src$
```

d. На панелі **EXPLORER** під розділом **DEVNET-SRC**, натисніть правою кнопкою миші на каталог **netconf** і виберіть пункт **Новий файл (New File)**.

e. Назвіть файл **ncclient-netconf.py**.

f. У ваш файл скрипту, імпортуйте клас **manager** з модуля **ncclient**. Потім створіть змінну **m** для представлення методу **connect()**. Метод **connect()** включає в себе всю інформацію, яка необхідна для під'єднання до служби **NETCONF**, яка працює на **CSR1kv**. Зверніть увагу на те, що для **NETCONF** використовується порт **830**.

```
from ncclient import manager

m = manager.connect (
    host="192.168.56.101",
    port=830,
    username="cisco",
    password="cisco123!",
    hostkey_verify=False
)
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 9

Якщо для **hostkey_verify** встановлено значення «True», CSR1kv попросить вас перевірити відбиток SSH. В лабораторних умовах, можна безпечно встановити це значення на «False», як ми це тут і зробили.

g. Збережіть та запустіть програму, щоб переконатися, що в ній немає помилок. Ви ще не побачите жодного результату.

```
devasc@labvm:~/labs/devnet-src$ cd netconf/
devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
devasc@labvm:~/labs/devnet-src/netconf$
```

h. Ви можете переконатися, що CSR1kv прийняв запит на сеанс NETCONF. У віртуальній машині CSR1kv повинно бути повідомлення системного журналу **%DMI-5-AUTH_PASSED**. Якщо екран чорний, натисніть клавішу **Enter**, щоб активізувати маршрутизатор. Повідомлення системного журналу можна побачити над баннером.

3.3: Додайте функцію друку до сценарію, щоб можна було виводити на друк список можливостей NETCONF для CSR1kv.

Об'єкт **m**, що повертається функцією **manager.connect()**, представляє віддалений сеанс NETCONF. Як ви вже бачили раніше, у кожному сеансі NETCONF сервер спочатку надсилає свої можливості, які представляють собою список у форматі XML підтримуваних моделей YANG. За допомогою модуля **ncclient**, отриманий список можливостей зберігається в списку **m.server_capabilities**.

a. Використайте цикл **for** і функцію **print**, щоб відобразити можливості пристрою:

```
print("#Supported Capabilities (YANG models):")
for capability in m.server_capabilities:
    print(capability)
```

b. Збережіть і запустіть програму. Кінцевий результат є аналогічним тому, який ви раніше отримували від надсилання складного повідомлення **hello**, але без відкриття і закриття тега XML **<capability>** у кожному рядку.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 10

```

devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
#Supported Capabilities (YANG models):
urn:ietf:params:netconf:base:1.0
urn:ietf:params:netconf:base:1.1
urn:ietf:params:netconf:capability:writable-running:1.0
urn:ietf:params:netconf:capability:xpath:1.0
<output omitted>
urn:ietf:params:xml:ns:netconf:base:1.0?module=ietf-netconf&revision=2011-06-01
urn:ietf:params:xml:ns:yang:ietf-netconf-with-defaults?module=ietf-netconf-with-
defaults&revision=2011-06-01

urn:ietf:params:netconf:capability:notification:1.1

devasc@labvm:~/labs/devnet-src/netconf$

```

4: Використання ncclient для отримання конфігурації

У цій частині ви будете використовувати: **ncclient** NETCONF для отримання конфігурації для CSR1kv, модуль **xml.dom.minidom** для форматування конфігурації та фільтр з `get_config()` для отримання частини поточної конфігурації.

4.1: Скористайтеся функцією `get_config()`, щоб отримати поточну конфігурацію для R1.

а. Якщо ви хочете пропустити виведення можливостей (понад 400 рядків), зробіть коментарем блок операторів, які друкують можливості, як показано це нижче

```

'''
print("#Supported Capabilities (YANG models):")
for capability in m.server_capabilities:
    print(capability)
'''

```

б. Ви можете використати метод `get_config()` об'єкта сеансу `m` NETCONF, щоб отримати конфігурацію для CSR1kv. Метод `get_config()` очікує вихідний параметр рядка, який визначає джерело сховища даних NETCONF. Використайте функцію друку для відображення результатів. Наразі єдиним сховищем даних NETCONF в CSR1kv є запущене (**running**) сховище даних. Ви це можете перевірити за допомогою команди **show netconf-yang datastores**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 11

```
netconf_reply = m.get_config(source="running")
print(netconf_reply)
```

с. Збережіть і запустіть вашу програму. Результат матиме значно більше 100 рядків, тому IDLE може їх стискати. Двічі натисніть на **Squeezed text** повідомлення у вікні оболонки IDLE, щоб розширити вихідні дані.

```
devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:3f31bedc-5671-47ca-9781-4d3d7aadae24"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"><data><native
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native"><version>16.9</version><boot-
start-marker/><boot-end-marker/><banner><motd><banner>
(output omitted)
devasc@labvm:~/labs/devnet-src/netconf$
```

d. Зверніть увагу, що повернутий XML не відформатований. Ви можете скопіювати його на той самий сайт, який ви знайшли раніше, щоб покращити цей XML-файл.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:3f31bedc-5671-
47ca-9781-4d3d7aadae24">
  <data>
    <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
      <version>16.9</version>
      <boot-start-marker/>
      <boot-end-marker/>
      <banner>
        <motd>
          <banner>^C </banner>
        </motd>
      </banner>
      <service>
        <timestamps>
          <debug>
            <datetime>
              <msec/>
            </datetime>
          </debug>
          <log>
            <datetime>
              <msec/>
            </datetime>
          </log>
        </timestamps>
      </service>
    </native>
  </data>
</rpc-reply>
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 12

```

    </timestamps>
</service>
<platform>
  <console xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-platform">
    <output>virtual</output>
  </console>
</platform>
<hostname>CSR1kv</hostname>
(output omitted)

```

4.2. Використайте Python для упорядкування XML.

Python має вбудовану підтримку роботи з XML-файлами. Модуль **xml.dom.minidom** може використовуватися для попереднього налаштування виведення результатів за допомогою функції **toprettyxml()**.

а. На початку вашого скрипту додайте оператор для імпорту модуля **xml.dom.minidom**.

```
import xml.dom.minidom
```

б. Замініть просту функцію друку **print(netconf_reply)** на версію, яка виводить на друк попередньо налаштовані вихідні дані XML.

```
print(xml.dom.minidom.parseString(netconf_reply.xml).toprettyxml())
```

с. Збережіть і запустіть вашу програму. XML відображається в більш зручному для читання форматі.

```

devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
<?xml version="1.0" ?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:3a5f6abc-76b4-
436d-9e9a-7758091c28b7">
  <data>
    <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
      <version>16.9</version>
      <boot-start-marker/>
      <boot-end-marker/>
      <banner>
        <motd>
          <banner>^C </banner>
        </motd>
      </banner>
    </data>
  </rpc-reply>

devasc@labvm:~/labs/devnet-src/netconf$

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 13

4.3: Використайте фільтр із `get_config()`, щоб отримати лише визначену модель YANG.

Адміністратор мережі може захотіти отримати лише частину запущеної конфігурації на пристрої. NETCONF підтримує повернення лише даних, визначених в параметрі фільтра функції `get_config()`.

а. Створіть змінну під назвою `netconf_filter`, яка отримує лише дані, визначені власною Cisco IOS XE Native YANG моделлю.

```
netconf_filter = ""
<filter>
    <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native" />
</filter>
"""
netconf_reply = m.get_config(source="running", filter=netconf_filter)
print(xml.dom.minidom.parseString(netconf_reply.xml).toprettyxml())
```

б. Збережіть і запустіть вашу програму. Початок виведення такий же, як показано нижче. Проте, цього разу відображається лише елемент `<native>` XML. Раніше були показані всі моделі YANG, доступні на CSR1kv. Фільтрування отриманих даних лише для відображення власного (native) модуля YANG значно зменшує обсяг виведення вихідних даних. Це пояснюється тим, що власний модуль YANG включає лише підмножину всіх моделей Cisco IOX XE YANG.

```
devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
<?xml version="1.0" ?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:4da5b736-1d33-47c3-8e3c-349414be0958">
    <data>
        <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
            <version>16.9</version>
            <boot-start-marker/>
            <boot-end-marker/>
            <banner>
                <motd>
                    <banner>^C </banner>
                </motd>
            </banner>
            <service>
                <timestamps>
                    <debug>
                        <datetime>
                            <msec/>
                        </datetime>
                    </debug>
                </timestamps>
            </service>
            <log>
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 14

```

                                <datetime>
                                    <msec/>
                                </datetime>
                            </log>
                        </timestamps>
                    </service>
                <platform>
                    <console
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-platform">
                                <output>virtual</output>

```

5. Використання ncclient для налаштування пристрою

5.1. Скористайтеся ncclient для редагування назви вузла (hostname) CSR1kv.

Щоб оновити існуючий параметр в конфігурації для CSR1kv, ви можете отримати місцезнаходження параметра конфігурації, яка була отримана раніше. На цьому кроці вам необхідно встановити змінну для зміни значення **<hostname>**.

```

devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
<?xml version="1.0" ?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:4da5b736-1d33-
47c3-8e3c-349414be0958">
    <data>
        <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
(output omitted)
                <hostname>CSR1kv</hostname>
(output omitted)

```

б. Використайте функцію **edit_config()** об'єкта сеансу m NETCONF, щоб надіслати конфігурацію та зберегти результати у змінній **netconf_reply** таким чином, щоб їх можна було вивести на друк. Параметри функції **edit_config()** наступні:

- **target** - цільове сховище даних NETCONF, яке необхідно оновити
- **config** - модифікація конфігурації, яку необхідно надіслати

```
netconf_reply = m.edit_config(target="running", config=netconf_hostname)
```

с. Функція **edit_config()** повертає повідомлення XML RPC із відповіддю **<ok/>**, що вказує на те, що зміна була успішно застосованою. Повторіть попередній оператор друку (**print**), щоб відобразити результати.

```
print(xml.dom.minidom.parseString(netconf_reply.xml).toprettyxml())
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 15

с. Функція `edit_config()` повертає повідомлення XML RPC із відповіддю `<ok/>`, що вказує на те, що зміна була у

```
print(xml.dom.minidom.parseString(netconf_reply.xml).toprettyxml())
```

d. Збережіть і запустіть вашу програму. Ви повинні отримати результат, подібний до результату, відображеного нижче. Ви також можете переконаватися у тому, що назва вузла змінилася, перемиканням на віртуальну машину CSR1kv

```
devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
(output omitted)
<?xml version="1.0" ?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:e304b225-7951-
4029-afd5-59e8e7edb00">
  <ok/>
</rpc-reply>

devasc@labvm:~/labs/devnet-src/netconf$
```

e. Відредагуйте ваш скрипт, щоб знову змінити назву вузла на CSR1kv. Збережіть і запустіть вашу програму. Ви також можете просто закоментувати код із попереднього кроку, якщо ви хочете уникнути повторної зміни назви вузла

5.2: Використайте ncclient, щоб створити новий інтерфейс «зворотньої петлі» (loopback) на R1.

a. Створіть нову змінну `<config>` для зберігання конфігурації нового інтерфейсу «зворотньої петлі». Додайте наступне до вашого скрипту `ncclient_netconf.py`.

Примітка: Ви можете використовувати довільний **description**. Проте використовуйте тільки буквено-цифрові символи, інакше вам потрібно буде відокремлювати їх за допомогою оберненої скісної риски (`\`).

```
netconf_loopback = """
<config>
  <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
    <interface>
      <Loopback>
        <name>1 </name>
        <description>My first NETCONF loopback</description>
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 16

```

<ip>
  <address>
    <primary>
      <address>10.1.1.1</address>
      <mask>255.255.255.0 </mask>
    </primary>
  </address>
</ip>
</Loopback>
</interface>
</native>
</config>
"""

```

б. Додайте наступну функцію `edit_config()` до вашого `ncclient_netconf.py`, щоб надіслати нову конфігурацію «loopback» на R1, а потім вивести на друк результати.

```

netconf_reply = m.edit_config(target="running", config=netconf_loopback)
print(xml.dom.minidom.parseString(netconf_reply.xml).toprettyxml())

```

с. Збережіть і запустіть вашу програму. Ви повинні отримати результат, подібний до наступного:

```

devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
(output omitted)
<?xml version="1.0" ?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:98437f47-7a93-
4cac-9b9e-9bc8afc9dfa1">
  <ok/>
</rpc-reply>

```

```
devasc@labvm:~/labs/devnet-src/netconf$
```

д. На CSR1kv переконайтеся, що створений новий інтерфейс «зворотньої петлі».

```

CSR1kv>en
CSR1kv# show ip interface brief
Interface                IP-Address      OK? Method  Status        Protocol
GigabitEthernet1        192.168.56.101  YES DHCP    up            up
Loopback1                 10.1.1.1        YES other   up            up
CSR1kv# show run | section interface Loopback1
interface Loopback1
  description My first NETCONF loopback
  ip address 10.1.1.1 255.255.255.0
CSR1kv#

```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 17

5.3: Спробуйте створити новий інтерфейс «зворотньої петлі» з тією ж IPv4-адресою.

а. Створіть нову змінну з назвою **netconf_newloop**. Він буде містити конфігурацію, яка створює новий інтерфейс loopback 2, але з тією ж IPv4-адресою, що і в loopback 1: 10.1.1.1 /24. В CLI маршрутизатора це призведе до помилки через спробу присвоїти інтерфейсу повторювану IP-адресу.

```
netconf_newloop = """
<config>
  <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
    <interface>
      <Loopback>
        <name>2 </name>
        <description>My second NETCONF loopback</description>
        <ip>
          <address>
            <primary>
              <address>10.1.1.1</address>
              <mask>255.255.255.0 </mask>
            </primary>
          </address>
        </ip>
      </Loopback>
    </interface>
  </native>
</config>
"""
```

б. Додайте наступну функцію **edit_config()** до вашого **ncclient_netconf.py**, щоб надіслати нову конфігурацію «loopback» на CSR1kv. Для цього кроку не потрібний оператор друку.

```
netconf_reply = m.edit_config(target="running", config=netconf_newloop)
```

с. Збережіть і запустіть програму. Ви повинні отримати повідомлення про помилку, подібне до наведеного нижче, із повідомленням про помилку RPCError - «Пристрій відмовив одній або кільком командам» («Device refused one or more commands»).

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.06- 05.01/121.00.1/Б /-2021
	Екземпляр № 1	Арк 11 / 18

```

devasc@labvm:~/labs/devnet-src/netconf$ python3 ncclient-netconf.py
Traceback (most recent call last):
  File "ncclient-netconf.py", line 80, in <module>
    netconf_reply = m.edit_config(target="running", config=netconf_newloop)
  File "/home/devasc/.local/lib/python3.8/site-packages/ncclient/manager.py", line 231, in execute
    return cls(self._session,
  File "/home/devasc/.local/lib/python3.8/site-packages/ncclient/operations/edit.py", line 69, in request
    return self._request(node)
  File "/home/devasc/.local/lib/python3.8/site-packages/ncclient/operations/rpc.py", line 348, in _request
    raise self._reply.error
ncclient.operations.rpc.RPCError: inconsistent value: Device refused one or more
commands
devasc@labvm:~/labs/devnet-src/netconf$

```

d. NETCONF не застосовуватиме жодної конфігурації, яка надсилається, якщо одну або кілька команд відхилено. Щоб перевірити це, введіть команду **show ip interface brief** на R1. Зверніть увагу, що ваш новий інтерфейс не був створений.

```

CSR1kv# show ip interface brief
Interface          IP-Address      OK? Method      Status          Protocol
GigabitEthernet1  192.168.56.101  YES DHCP        up              up
Loopback1          10.1.1.1        YES other       up              up

```

Запишіть висновки по виконаній роботі.