

## Практическая работа № 7. Доступ к сетевым устройствам по протоколу SSH и обеспечение безопасности

Топология сети в среде PNETLab:

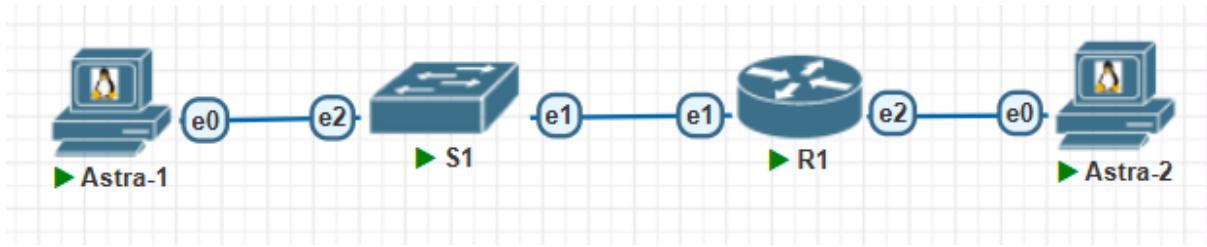


Схема IP-адресации устройств:

| Устройство        | IP-адрес/маска подсети |
|-------------------|------------------------|
| Astra-1           | 192.168.2.22/24        |
| S1                | 192.168.2.222/24       |
| R1 (интерфейс e1) | 192.168.2.2/24         |
| R1 (интерфейс e2) | 192.168.3.3/24         |
| Astra-2           | 192.168.3.33/24        |

### Задание:

Создайте топологию сети, состоящую из коммутатора, маршрутизатора и двух компьютеров, как показано на рисунке. Настройте IP-адреса в соответствии со схемой IP-адресации и отправьте эхо-запрос между всеми устройствами в сети.

Подключитесь по SSH с двух компьютеров (под управлением ОС Astra Linux) к коммутатору и маршрутизатору.

### Решение:

#### 1. Создание топологии сети

1.1. Добавьте все устройства на рабочую область так же, как вы делали в четвертой практической работе, КРОМЕ:

- иконки для компьютеров (необходимо выбрать иконку Desktop\_linux);
- образ для компьютеров – linux-Astra\_Spec\_Line\_Bolshoi\_shrift.

1.2. Соедините все устройства кабелями, как показано на топологии.

#### 2. Подключение к устройствам и конфигурирование сети

2.1. Подключитесь к S1 (на данном этапе устройство является маршрутизатором). Введите логин (admin) и пароль (password) пользователя по умолчанию.

2.2. Измените пароль для пользователя admin (это нужно для того, чтобы получить доступ к устройству):

```
vesr(change-expired-password) # password 12345678
```

2.3. Примените и сохраните изменения:

```
vesr(change-expired-password) # commit  
vesr(change-expired-password) # confirm  
vesr#
```

2.4. Войдите в режим конфигурирования и установите имя хоста на S1:

```
vesr# configure  
vesr(config) # hostname S1
```

2.5. Сохраните изменения в постоянную память устройства:

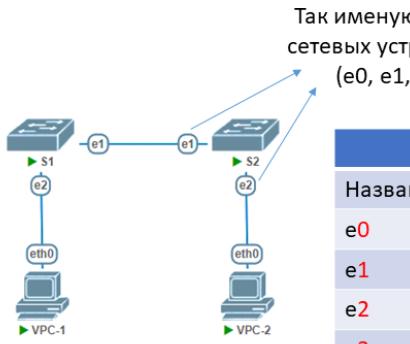
```
vesr(config) # exit  
vesr# commit  
S1# confirm
```

2.6. Проверьте доступ к интерфейсам S1, перейдя в debug-меню, и отобразите информацию про MAC-адреса интерфейсов:

```
S1# debug  
S1(debug) # show nic
```

2.7. Привяжите MAC-адреса интерфейсов gi1/0/1 и gi1/0/2 (2 первых интерфейса) к интерфейсам e1 и e2, которые используются в вашей топологии. Вместо xx:xx:xx:xx:xx:01 и xx:xx:xx:xx:02 указываются реальные значения MAC-адресов, которые вы увидели в предыдущем пункте. Для подтверждения изменений нажмите «у».

КАК ЭТО РАБОТАЕТ:



Так именуются интерфейсы сетевых устройств в PNETLab (e0, e1, e2, e3 и т.д.)

| Как могут выглядеть интерфейсы в консоли устройств |                   |              |
|--|-------------------|--------------|
| Название интерфейса                                | MAC-адрес         | С чем связан |
| e0   | XX:XX:XX:XX:XX:00 | gi1/0/1      |
| e1   | XX:XX:XX:XX:XX:01 | gi1/0/2      |
| e2   | XX:XX:XX:XX:XX:02 | gi1/0/3      |
| e3   | XX:XX:XX:XX:XX:03 | gi1/0/4      |

Каждому сетевому устройству добавляется на 2 интерфейса больше, чем используется. Первый и последний интерфейсы использовать не будут

| Как должно быть в реальности для интерфейсов e1 и e2 |                   |              |
|--|-------------------|--------------|
| Название интерфейса                                  | MAC-адрес         | С чем связан |
| e1   | XX:XX:XX:XX:XX:01 | gi1/0/1      |
| e2   | XX:XX:XX:XX:XX:02 | gi1/0/2      |

Для этого мы вводим:

```
nic bind mac xx:xx:xx:xx:xx:01 gi1/0/1
nic bind mac xx:xx:xx:xx:xx:02 gi1/0/2
```

```
S1(debug) # nic bind mac xx:xx:xx:xx:xx:01 gi1/0/1
S1(debug) # nic bind mac xx:xx:xx:xx:xx:02 gi1/0/2
S1(debug) # exit
```

2.8. Для применения настроек необходимо выполнить перезагрузку устройства.

```
S1# reload system
```

2.9. Измените маршрутизируемые порты на «порты коммутатора».

```
S1# configure
S1(config)# interface gi1/0/1-2
S1(config-if-gi)# mode switchport
S1(config-if-gi)# exit
```

2.10. Создайте новый сетевой мост bridge 1 и присвойте ему IP-адрес, а также измените MAC-адрес моста и отключите firewall:

```
S1(config)# bridge 1
S1(config-bridge)# vlan 1
S1(config-bridge)# enable
S1(config-bridge)# ip address 192.168.2.222/24
```

```
S1(config-bridge)# mac-address a2:20:00:00:00:00
S1(config-bridge)# ip firewall disable
S1(config-bridge)# end
S1# commit
S1# confirm
```

2.11. Подключитесь к R1. Введите логин (admin) и пароль (password) пользователя по умолчанию.

2.12. Измените пароль для пользователя admin (это нужно для того, чтобы получить доступ к устройству):

```
vesr(change-expired-password)# password 12345678
```

2.13. Примените и сохраните изменения:

```
vesr(change-expired-password)# commit
vesr(change-expired-password)# confirm
vesr#
```

2.14. Войдите в режим конфигурирования и установите имя хоста на R1:

```
vesr# configure
vesr(config)# hostname R1
```

2.15. Сохраните изменения в постоянную память устройства:

```
vesr(config)# exit
vesr# commit
R1# confirm
```

2.16. Проверьте доступ к интерфейсам R1, перейдя в debug-меню, и отобразите информацию про MAC-адреса интерфейсов:

```
R1# debug
R1(debug)# show nic
```

2.17. Привяжите MAC-адреса интерфейсов gi1/0/1 и gi1/0/2 (2 первых интерфейса) к интерфейсам e1 и e2, которые используются в вашей топологии. Вместо xx:xx:xx:xx:xx:01 и xx:xx:xx:xx:xx:02 указываются реальные значения MAC-адресов, которые вы увидели в предыдущем пункте. Для подтверждения изменений нажмите «у».

```
R1(debug)# nic bind mac xx:xx:xx:xx:xx:01 gi1/0/1  
R1(debug)# nic bind mac xx:xx:xx:xx:xx:02 gi1/0/2  
R1(debug)# exit
```

2.18. Для применения настроек необходимо выполнить перезагрузку устройства.

```
R1# reload system
```

2.19. Настройте IP-адресацию интерфейсов и отключите firewall на маршрутизаторе R1:

```
R1# configure  
R1(config)# interface gi1/0/1  
R1(config-if-gi)# ip address 192.168.2.2/24  
R1(config-if-gi)# ip firewall disable  
R1(config-if-gi)# exit  
R1(config)# interface gi1/0/2  
R1(config-if-gi)# ip address 192.168.3.3/24  
R1(config-if-gi)# ip firewall disable  
R1(config-if-gi)# end  
R1# commit  
R1# confirm
```

### **3. Конфигурация хостов и проверка подключения**

3.1. Подключитесь к компьютеру Astra-1. Логин – student, пароль – student.

3.2. Настройте ему IP-адрес, маску подсети и шлюз по умолчанию:

```
student@astra:~& sudo ip addr add 192.168.2.22/24 dev eth0  
student@astra:~& sudo ip route add 0.0.0.0/0 via 192.168.2.2
```

3.3. Подключитесь к компьютеру Astra-2. Логин – student, пароль – student.

3.4. Настройте ему IP-адрес, маску подсети и шлюз по умолчанию:

```
student@astra:~& sudo ip addr add 192.168.3.33/24 dev eth0  
student@astra:~& sudo ip route add 0.0.0.0/0 via 192.168.3.3
```

3.5. Отправьте эхо-запрос с компьютера Astra-1 до двух интерфейсов маршрутизатора, коммутатора и до второго компьютера:

```
student@astra:~& ping 192.168.2.2  
student@astra:~& ping 192.168.3.3  
student@astra:~& ping 192.168.2.222  
student@astra:~& ping 192.168.3.33
```

Эхо-запрос во всех случаях должен проходить успешно.

#### **4. Подключение к маршрутизатору и коммутатору по SSH**

4.1. Подключитесь с компьютера Astra-1 на маршрутизатор R1 по протоколу SSH. Для выхода необходимо ввести команду **exit**.

```
student@astra:~& ssh admin@192.168.2.2
```

4.2. Подключитесь с компьютера Astra-1 на коммутатор S1 по протоколу SSH.

```
student@astra:~& ssh admin@192.168.2.222
```

4.3. Подключитесь с компьютера Astra-2 на маршрутизатор R1 по протоколу SSH.

```
student@astra:~& ssh admin@192.168.3.3
```

4.4. Задайте маршрут по умолчанию на коммутаторе S1 (это необходимо, чтобы была возможность подключаться к нему из удаленной сети). При использовании реального коммутатора вместо данной команды будет использоваться задание шлюза по умолчанию (команда **ip default-gateway**).

```
S1(config)# ip route 0.0.0.0/0 192.168.2.2  
S1(config)# end  
S1# commit  
S1# comfirm
```

4.5. Подключитесь с компьютера Astra-2 на коммутатор S1 по протоколу SSH.

```
student@astra:~& ssh admin@192.168.2.222
```

**Результат практической работы:** показать возможность подключения по SSH к маршрутизатору и коммутатору с обоих компьютеров.