```
↑ demo-2025 / modul-2 / modul-2-4
```

# 04. Модуль 2. Задание 4

# Задание

### Сконфигурируйте ansible на сервере BR-SRV

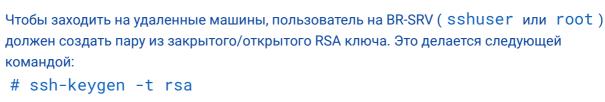
- ▶ Сформируйте файл инвентаря, в инвентарь должны входить HQ-SRV, HQ-CLI, HQ-RTR и BR-RTR
- ▶ Рабочий каталог ansible должен располагаться в /etc/ansible
- ► Все указанные машины должны без предупреждений и ошибок отвечать pong на команду ping в ansible посланную с BR-SRV

# Реализация

### Установка Ansible

```
1 | # dnf install ansible -y
```

## Создание пары SSH-ключей.



B результате в каталоге /home/sshuser/.ssh или /root/.ssh будут созданы файлы ключей:

```
ls -l ~/.ssh
id_rsa - закрытый ключ
id_rsa.pub - открытый ключ
```

### Заходим под пользователем sshuser

```
1 # su sshuser
```

#### Создаем пару ключей

1 | # ssh-keygen -t rsa



Koпируем открытый SSH-ключ на удаленные устройства под пользователем sshuser:



SSH-ключи для Ansible рекомендуется копировать для того пользователя, под которым будут выполняться команды Ansible . Обычно это обычный локальный пользователь , на котором запускаются команды

### Копируем ключ для пользователя sshuser на HQ-SRV

Ha HQ-SRV ssh порт изменен, указываем его

1 | ssh-copy-id -p 2024 sshuser@192.168.100.2

#### Копируем ключ для пользователя user на HQ-CLI

1 ssh-copy-id user@192.168.100.66

#### Копируем ключ для пользователя net\_admin на HQ-RTR

1 ssh-copy-id net\_admin@172.16.4.2

#### Копируем ключ для пользователя net\_admin на BR-RTR

1 | ssh-copy-id net\_admin@172.16.5.2

# Подготовка файла инвентаря (hosts)



Для того чтобы Ansible знал, на каких хостах необходимо выполнять определенные действия, используется файл инвентаря. По умолчанию /etc/ansible/hosts. В нем указываются IP-адреса или доменные имена необходимых серверов.

#### Открываем файл инвентаря /etc/ansible/hosts

nano /etc/ansible/hosts

Для создания файла инвентаря достаточно поместить в него список хостов (каждый новый хост – с новой строки). Имена хостов и IP-адреса взаимозаменяемы. В файле инвентаря распределим устройства по группам (HQ и BR).



Сначала указывается название группы в квадратных скобках, затем перечисляются хосты.

Имена хостов прописываются в виде dns-имени хоста или ip.
После названия хоста можно указать параметры подключения к хосту в одну строку.

#### И в начале файла пишем

```
GNU nano 7.2

[hq]
192.168.100.2 ansible_port=2024 ansible_user=sshuser
192.168.100.66 ansible_user=user
172.16.4.2 ansible_user=net_admin

[br]
172.16.5.2 ansible_user=net_admin
```



#### где:

ansible\_port - Номер порта ssh, если не 22 ansible\_user - Использовать имя пользователя ssh по умолчанию.

## Запуск команд

Что бы запустить модуль ping на всех хостах, перечисленных файле инвентаря /etc/ansible/hosts.

```
1 | ansible all -m ping
```

Может появиться предупреждение про обнаружение интерпретатора Python , на целевом хосте

```
[root@br-srv ]# ansible all -i /ctc/ansible/deno -m ping
[UARNING]: Platform linux on host 192.168.100.2 is using the discovered Python interpreter at /usr/bin/python3.11, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.16/reference_appendices/interpreter_discovery.html for more information.
```

Для управления поведением обнаружения в глобальном масштабе необходимо в файле конфигурации ansible /etc/ansible/ansible.cfg в разделе [defaults] прописать ключ interpreter\_python с параметром auto\_silent. В большинстве дистрибутивов прописываем вручную.

1 | # nano /etc/ansible/ansible.cfg

#### Прописываем вручную

```
# Since Ansible 2.12 (core):
# To generate an example config file (a "disabled" one with all default settings, commented out):
# $ ansible-config init --disabled > ansible.cfg
# Also you can now have a more complete file by including existing plugins:
# ansible-config init --disabled -t all > ansible.cfg
# For previous versions of Ansible you can check for examples in the 'stable' branches of each version
# Note that this file was always incomplete and lagging changes to configuration settings
# for example, for 2.9: https://github.com/ansible/ansible/blob/stable-2.9/examples/ansible.cfg

[defaults]
interpreter_python=auto_silent
```

44

#### Запускаем комунду ping на всех хостах

```
[sshuser@br-srv root]$
[sshuser@br-srv root]$ ansible all -m ping
192.168.200.1 | SUCCESS => (
"ansible_facts": (
         "discovered_interpreter_python": "/usr/bin/python3.11"
    "changed": false, 
"ping": "pong"
192.168.100.1 | SUCCESS => {
     "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.11"
    "changed": false,
"ping": "pong"
192.168.100.2 | SUCCESS => {
     "ansible_facts": {
          'discovered interpreter python": "/usr/bin/python3.11"
    },
"changed": false,
    "ping": "pong"
192.168.100.66 | SUCCESS => {
     "ansible_facts": {
         "discovered_interpreter_python": "/usr/bin/python3.11"
    },
"changed": false,
    "ping": "pong
[sshuser@br-srv root]$
```

Содержимое доступно в соответствии с Всеобщее достояние, от Кабинет 2.20. | Powered by Wiki.js