

## Homework\_Lesson19\_Ansible\_1\_2

Цель 1: получить практический опыт работы с системой управления конфигурацией Ansible. Перед началом работы с Ansible, разверните еще одну виртуальную машину, которая будет управляться с помощью него.

Задание 1: установить Ansible

Обновите список пакетов: `sudo apt update`. Установите пакет `software-properties-common`, который позволит добавлять репозитории: `sudo apt install software-properties-common`. Добавьте репозиторий Ansible PPA: `sudo apt-add-repository ppa:ansible/ansible`. Обновите список пакетов: `sudo apt update`. Установите Ansible: `sudo apt install ansible`

Задание 2: сгенерировать SSH-ключ

- Откройте терминал на вашем контролирующем хосте.
- Запустите команду для генерации ключа: `ssh-keygen -t rsa -b 4096`

При выполнении этой команды вам будет предложено ввести путь к файлу, в котором будет сохранен ключ, а также парольная фраза (пароль не обязателен, но рекомендуется). После того, как ключ будет сгенерирован, скопируйте его на управляемый хост: `ssh-copy-id user@managed_host`. В этой команде "user" - это имя пользователя на управляемом хосте, а "managed\_host" - имя или IP-адрес управляемого хоста. • При выполнении этой команды вам будет предложено ввести пароль пользователя на управляемом хосте. • После успешной передачи ключа на управляемый хост вы можете использовать Ansible для управления им.

Задание 3: просмотрите и повторите все манипуляции, указанные в видеоуроке по ссылке [https://www.youtube.com/watch?v=O5R6EBdaZZg&list=PLg5SS\\_4L6LYufspdPupdynbMQTBnZd31N&index=4](https://www.youtube.com/watch?v=O5R6EBdaZZg&list=PLg5SS_4L6LYufspdPupdynbMQTBnZd31N&index=4). В данном ролике рассматриваются шаги подключения к серверам Linux.

Цель 2: настроить Ansible окружение на вашей машине, чтобы вы могли использовать его для автоматизации управления конфигурацией и развертывания приложений на удаленных серверах.

Задание: Настройка Ansible окружения

Шаги:

1. Установите Ansible на вашу локальную машину. В зависимости от операционной системы, которую вы используете, установка может отличаться. Инструкции по установке Ansible можно найти на официальном сайте: [https://docs.ansible.com/ansible/latest/installation\\_guide/index.html](https://docs.ansible.com/ansible/latest/installation_guide/index.html)

2. Создайте инвентарный файл, который содержит информацию о серверах, которые вы планируете управлять с помощью Ansible. Инвентарный файл может быть в формате ini или YAML. Например:

```
[servers]
```

```
server1 ansible_host=192.168.1.100 ansible_user=myuser
```

```
ansible_ssh_private_key_file=/path/to/key.pem
```

```
server2 ansible_host=192.168.1.101 ansible_user=myuser
```

```
ansible_ssh_private_key_file=/path/to/key.pem
```

В этом примере мы определяем два сервера с IP-адресами 192.168.1.100 и 192.168.1.101. Мы также указываем имя пользователя и путь к файлу с приватным ключом SSH, который Ansible будет использовать для аутентификации.

3. Создайте файл конфигурации Ansible, который будет содержать глобальные настройки Ansible. Файл конфигурации Ansible может быть расположен в нескольких местах, включая каталоги `/etc/ansible` и `~/.ansible/`. Например:

```
[defaults]
```

```
inventory = /path/to/inventory_file
```

```
remote_user = myuser
```

```
private_key_file = /path/to/key.pem
```

В этом примере мы указываем путь к инвентарному файлу, имя пользователя SSH и путь к файлу с приватным ключом SSH, который Ansible будет использовать для аутентификации.

4. Создайте простой плейбук Ansible, который будет выполнять простую задачу на одном из ваших серверов. Например, вы можете создать плейбук, который устанавливает пакеты на сервере. Пример:

---

- name: Install packages

hosts: servers

become: true

tasks:

- name: Install Nginx

apt:

name: nginx

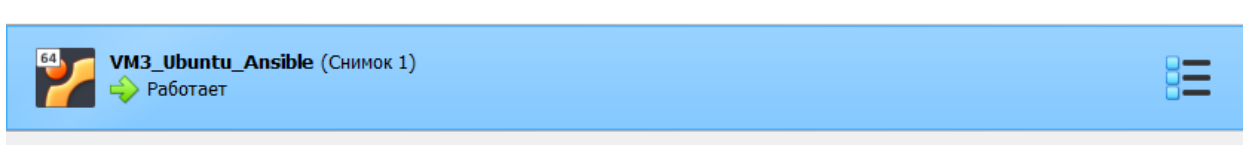
state: present

В этом примере мы создаем плейбук, который устанавливает пакет Nginx на всех серверах, определенных в инвентарном файле. Мы также указываем, что Ansible должен использовать привилегии root для выполнения.

**Цель 1-ая:** получить практический опыт работы с системой управления конфигурацией Ansible.

**Выполнение 1-ого задания:**

Создана отдельная ВМ на VirtualBox с которой будет менеджиться Ansible. ВМ VM\_Ubuntu\_Ansible: 4 ГБ ОЗУ, 1 CPU, 25 ГБ жесткий диск, развернута на ней ОС Ubuntu 22.04 и настроено две адаптера сети NAT и Сетевой мост (IP-192.168.100.14/24).



На WSL (наш контролирующий хост) выполняем следующие действия:

```
$ sudo apt update
```

```
$ sudo apt install software-properties-common
```

 \\ уст. пакета для доб. репозиториев

```
$ sudo apt-add-repository ppa:ansible/ansible
```

 \\ добавление репозитория Ansible

```
$ sudo apt update
```

```
$ sudo apt install ansible
```

 \\ установка Ansible

```
$ ansible --version
```

 \\ проверка версии Ansible

```
makarov@DESKTOP-UG6J7T7:~$ ansible --version
ansible [core 2.14.18]
  config file = None
  configured module search path = ['/home/makarov/.ansible/plugins/modules',
  '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/makarov/.ansible/collections:/usr/shar
  e/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.11.2 (main, Nov 30 2024, 21:22:50) [GCC 12.2.0] (/usr/b
  in/python3)
  jinja version = 3.1.2
  libyaml = True
makarov@DESKTOP-UG6J7T7:~$ |
```

Версия установленного Ansible 2.17.7

## Выполнение 2-ого задания:

На WSL выполняем следующие действия:

```
$ ssh-keygen -t rsa -b 4096           \ \ создание ключа
$ ssh-copy-id user@192.168.100.14     \ \ копируем наш ключ на ВМ
```

где, user - это пользователь на ВМ (VM\_Ubuntu\_Ansible) и его IP-адрес 192.168.100.14.

```
makarov@DESKTOP-UG6J7T7:~$ ssh-copy-id user@192.168.100.14
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/makarov/.ssh/id_rsa.pub"
The authenticity of host '192.168.100.14 (192.168.100.14)' can't be established.
ED25519 key fingerprint is SHA256:LDt1pxiMRWuYj6riQsBewRJx8+WxIIA4mh6C+Db306g.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
user@192.168.100.14's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'user@192.168.100.14'"
and check to make sure that only the key(s) you wanted were added.
```

Копирование ssh-ключа на ВМ 192.168.100.14

## Выполнение 3-его задания:

```
$ mkdir ansible
$ cd ./ansible
$ touch hosts.txt
```

Вносим данные в файл hosts.txt:

Создаем группу [staging\_servers] и дальше в строчку прописываем его alias name (user-VirtualBox), ansible\_host=IP-адрес (192.168.100.14), ansible\_user=user (пользователь которым подключаешься к ВМ (user - на стороне ВМ 192.168.100.14) + ansible\_pass=password, но он нам не понадобится, т.к. мы уже прокинули ключи по SSH. Также можно использовать ключи “.pem” и вместо ansible\_pass прописывается ansible\_ssh\_private\_key\_file=/\*//\*.pem (путь, где лежат \*.pem ключи).

```
[staging_servers]
user-VirtualBox      ansible_host=192.168.100.14      ansible_user=user
```

```
$ ansible -i hosts.txt all -m ping
```

где, -i - использование инвентори файла (в нашей ситуации, это hosts.txt)

all - указывает на все группы

-m - указания модуля

```
makarov@DESKTOP-UG6J7T7:~/ansible$ ansible -i hosts.txt all -m ping
user-VirtualBox | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
makarov@DESKTOP-UG6J7T7:~/ansible$ |
```

Успешное подключение к клиенту

Создаем файл конфигурационный файл `ansible.cfg`. В нем можно конфигурировать и настраивать под себя настройки.

\$ `ansible --version`

```
makarov@DESKTOP-UG6J7T7:~/ansible$ ansible --version
ansible [core 2.14.18]
  config file = None
  configured module search path = ['/home/makarov/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/makarov/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.11.2 (main, Nov 30 2024, 21:22:50) [GCC 12.2.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
makarov@DESKTOP-UG6J7T7:~/ansible$ |
```

Здесь мы видим, что конфигурационного настроенного файла нету

Создаем конфигурационный файл `.cfg`:

\$ `nano ansible.cfg`

```
makarov@DESKTOP-UG6J7T7:~/ansible$ cat ansible.cfg

[defaults]
host_key_checking = false
inventory          = ./hosts.txt
```

`host_key_checking = false`    \\\

`inventory = ./hosts.txt`    \\\ указывает какой файл инвентори используем

Пробуем запустить `ansible`:

\$ `ansible all -m ping`

```
makarov@DESKTOP-UG6J7T7:~/ansible$ ansible all -m ping
user-VirtualBox | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
makarov@DESKTOP-UG6J7T7:~/ansible$ |
```

Теперь не нужно указывать `inventory` файл, т.к. мы его уже прописали в конфигурационном файле `ansible.cfg`