# Лабораторная работа №14. Работа с Ansible

Цель работы: формирование практических навыков работы с Ansible.

**Установка Ansible**

Для работы Ansible нужно установить на управляющем и управляемых хостах ssh и python. Сам Ansible нужно установить только на управляющем сервере.

OpenSSH сервер обычно установлен по умолчанию во всех версиях Linux, поэтому осталось установить Python 3+ и сам Ansible.

В начале необходимо проверить наличие python в данный момент на рабочей машине, для этого используется команда:

python3 --version

Для работы с Ansible требуется версия 2.6 и выше.

В случае несоответствия или отсутствия python необходимо установить его при помощи команды:

dnf install python3

Далее необходимо установить непосредственно сам Ansible:

dnf install epel-release
dnf install ansible

**Начало работы с Ansible**

При установке пакет будет создан каталог /etc/ansible со следующими конфигурационными файлами:

* /etc/ansible/hosts – здесь можно указано список хостов, которыми вы будет управлять через ansible;
* /etc/ansible/ansible.cfg — файл с настройками ansible.

В файле /etc/ansible/hosts можно создать несколько отдельных групп хостов. Например, все хосты с nginx, с базами mariadb и т.д. В этом примере мы создадим одну группу с именем
[ваше\_имя].

Название\_сервера1 ansible\_host=ip\_адрес1 ansible\_user=имя\_пользователя1

Название\_сервера2 ansible\_host= ip\_адрес2 ansible\_user= имя\_пользователя2

Название\_сервера3 ansible\_host= ip\_адрес3 ansible\_user= имя\_пользователя3

Чтобы вывести содержимое файла инвентаризации в виде дерева, выполните:

ansible-inventory --graph

По умолчанию при подключении к удаленным хостам используется SSH. Чтобы автоматически использовать ssh fingerprint и не вводить yes при стартовом доступе к хосту, необходимо добавить параметр в файл /etc/ansible/ansible.cfg:

"host\_key\_cheking = false"

Проверим доступность всех хостов в файле инвентаризации:

ansible all -m ping --ask-pass

В случае, если все хосты доступны, то вы всё выполнили верно.

Чтобы ansible мог аутентифицироваться по SSH с помощью пароля, нужно установить пакет sshpass:

dnf install sshpass

Иначе при использовании параметра --ask-pass будет появляться ошибка.

Ранее для подключения к удаленным хостам необходимо каждый раз вводить пароль пользователя. Чтобы при выполнении команд и плейбуков ansible не запрашивался пароль, нужно настроить [SSH аутентификацию по ключам](https://winitpro.ru/index.php/2019/11/13/autentifikaciya-po-ssh-klyucham-v-windows/).

Теперь вы можете выполнить команду через ansible без ввода пароля. Проверим uptime на всех серверах в группе название\_группы:

ansible название\_группы -a 'uptime'

Приведем несколько примеров интерактивного выполнения команд на хостах в файле инвентаризации.

Выполним инвентаризацию состояния хостов. В этом примере нам нужна только информацию о RAM на серверах:

ansible -m setup -a 'filter=ansible\_memtotal\_mb' all

С помощью модуля shell проверим uptime на всех хостах:

ansible -m shell -a 'uptime' all

**Плейбуки в Ansible**

Вы можете отправлять команды на управляемые сервера через консоль (ad-hoc) или с помощью специального файла playbook в формате YAML. В плейбуке можно описать требуемое состояние системы. Ansible при запуске будет проверять, соответствует ли конфигурация управляемого хоста описанию в плейбуке.

Рассмотрим пример простого плейбука, который должен установить на хостах файловый менеджер Midnight Commander (mc).

Создайте каталог для плейбуков:

mkdir -p /etc/ansible/playbooks

Теперь создайте YML файл:

touch /etc/ansible/playbooks/mc-deploy.yml

Далее необходимо заполнить данный yml файл:

- hosts: ваше\_имя\_группы

 become: yes

 become\_method: sudo

 tasks:

 - name: update

 yum: update\_cache=yes

 - name: Install mc

 yum: name=mc state=latest

В синтаксисе YML-файлов используется строгая система отступов, как в Python. Используйте пробелы, а не табуляцию.

Чтобы выполнить плейбук на хостах в группе:

$ ansible-playbook /etc/ansible/playbooks/mc-deploy.yml --ask-become-pass

**Задание по вариантам**

|  |  |
| --- | --- |
| № варианта | Задание |
| 1 | Выполнить отключение службы nginx |
| 2 | Проверить существует ли файл |
| 3 | Установить пакет chrony  |
| 4 | Включить firewalld |
| 5 | Распаковать архив |
| 6 | Добавить задание в модуль cron |
| 7 | Создать учетную запись |
| 8 | Установить расширение для python |
| 9 | Задать часовой пояс |
| 10 | Заменить строки в файлах |

В отчете обязательно приведите результат работы по варианту с демонстрацией работы плейбука.