

Установка и настройка

- [Установка](#)
- [Настройка ansible](#)

Установка

На Debian-подобных системах:

```
apt-get install ansible
```

Настройка управляемых хостов

На каждом хосте:

1. Если sudo нет, то

```
apt install sudo
```

2. Если создается новый пользователь, то

```
sudo useradd -G sudo -s /bin/bash ansibleuser
```

3. Если существующий пользователь, то добавить в группу sudo:

```
usermod -aG sudo sergey
```

4. Разрешить пользователю повышать привилегии без ввода пароля. Создать файл

```
sudo nano /etc/sudoers.d/ansibleuser
```

5. И добавить текст

```
ansibleuser ALL=(ALL) NOPASSWD:ALL
```

Теперь можно использовать в скриптах become: true

Настройка сервера

```
ssh-keygen
```

```
ssh-copy-id username@remote_host
```

1. Создать ключ доступа
2. Скопировать на каждый управляемый хост, используя логин на хост. Удобнее, если для задач управления на всех хостах сделать одинаковый логин.

В настройке каждого хоста нужно использовать соответствующий логин, для которого настроен ключ доступа.

Настройка ansible

Приоритет поиска файла настроек

1. ANSIBLE_CONFIG (environment variable if set)
2. ansible.cfg (in the current directory)
3. ~/.ansible.cfg (in the home directory)
4. /etc/ansible/ansible.cfg (default)

Генерация настроек

```
ansible-config init --disabled > ansible.cfg
```

С плагинами

```
ansible-config init --disabled -t all > ansible.cfg
```

и ; комментарии. Но ; обычно используется для комментария значения по умолчанию.

Файл инвентаризации

hosts

Размещение файла может задаваться переменной окружения \$ANSIBLE_HOSTS, либо ключ -i при запуске

После : пишется тип данных для группы. Например группа [example], [example:vars] это переменные

:vars переменные для группы

:children группы-потомки

Пример

```
[example] группа
www.example.com сервер

# Group 'multi' with all servers
[multi:children]
example

#5 servers in one line [a:z] or [A:Z], or numbers with specific digits, such as [001:250].
[dyngroup]
Node[0:4].lab.edu

# Variables that will be applied to all servers
[multi:vars]
```

```
ansible_ssh_user=vagrant
```

Файлы размещения переменных:

Файлы с переменными групп хранятся в директории "group_vars/имя_группы";

Файлы с переменными хостов в директории "hosts_vars/имя_хоста";

Формат файла одинаковый, дочерние переменные заменяются на родительские.

```
---
ansible_user=setup
ansible_private_ssh_key=/home/user/ansible.key
```

Может применяться и плоское определение, и вложенное, тогда в скрипте {{ db.user }}:

```
db: III
  user: ppp
```

Динамический реестр:

Должен поддерживать следующий интерфейс:

```
--host=hostname - реестр выдает список в JSON { "ansible_ssh_host": "127.0.0.1",
"ansible_ssh_port": 2200, "ansible_ssh_user": "vagrant"}
--list реестр выдает список групп {"staging": [ "ontaro.example.com",
"quebec.example.com"], "vagrant": [ "vagrant1", "vagrant2", "vagrant3"]}
"_meta" : {"hostvars" : {"vagrant1" : {"ansible_ssh_host": "127.0.0.1", "ansible_ssh_port":
2222, "ansible_ssh_user": "vagrant"}, "vagrant2": {"ansible_ssh_host": "127.0.0.1",
"ansible_ssh_port": 2200, "ansible_ssh_user": "vagrant"}}}
```

Если создать папку inventory и добавить в ansible.cfg параметр hostfile = inventory то статический hosts и динамический будут объединены

Если включить сбор фактов gather_facts: True то можно во время выполнения задач группировать хосты (например debian/centos)

Использование динамического реестра (в примере my_dynamic_inventory.py должен быть исполняемым скриптом)

```
ansible-playbook -i my_dynamic_inventory.py my_playbook.yml
```

В Ссылках есть материал по динамическим реестрам

Предустановленные переменные

Переменная	Использование
ansible_user	пользователь, от имени которого выполняются задачи на хостах данной группы. Значение в файле задач игнорируется.
ansible_host	Сопоставление ip-имени
ansible_port	Порт доступа по ssh

ansible_connection	Может быть ssh, local, docker (запуск команд непосредственно на контейнере)
ansible_become	+ sudo
ansible_become_user	sudo -> another user
ansible_ssh_private_key_file	адрес ключа

Ссылки:

[Описание основного config файла](#)

[Динамический реестр](#)