Деплой Asterisk с помощью 1 команды

Обо мне

- Евгений Гостьков
- Ведущий системный администратор в zt.team (Защищенные телекоммуникации)

О чем доклад

- Проблемы, связанные с подготовкой сервера и установкой Asterisk
- Автоматизация процесса деплоя с использованием Ansible и других инструментов
- Варианты использования Ansible

Как все устанавливают Asterisk

- wget asterisk.tar.gz && make menuconfig && make install (много «мусора» в системе, сложно установить на n-машин)
- apt install asterisk / yum install asterisk (ничего лишнего в системе, удобно для установки на несколько хостов)

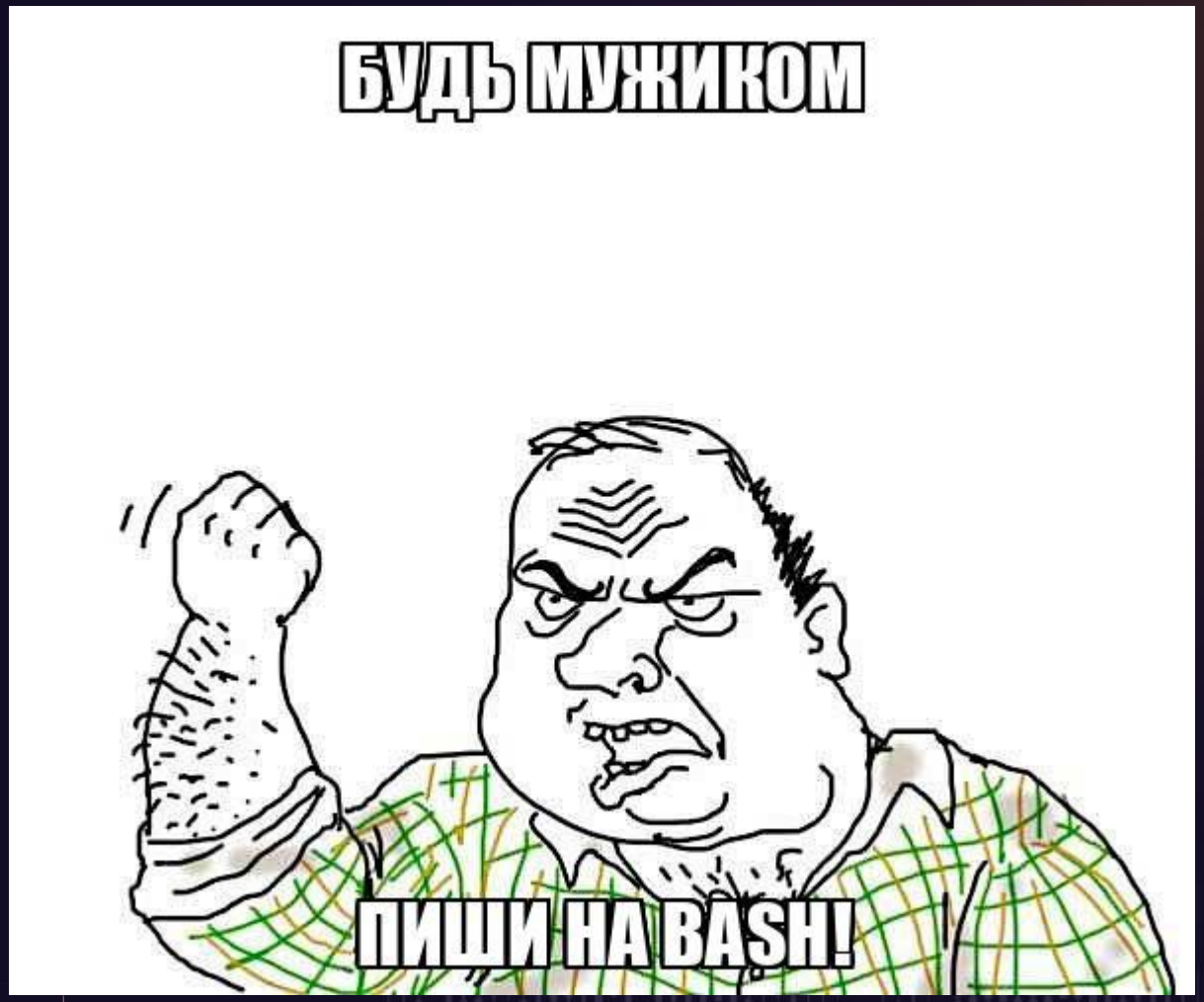
Что ещё нужно для работы?

- Создать пользователей
- Установить любимые утилиты (htop, mc, tcpdump, sngrep, etc…)
- Настроить firewall
- Скопировать/установить какие-либо скрипты
- Настроить сервисы (rsyslog, logrotate, Zabbix-agent)*
- Что-нибудь не стандартное с подключение сторонних репозиториев

Что тут сложного?

- Не все члены команды в курсе того, что вообще происходит
- Спустя год/два/три уже сложно вспомнить, как настраивали
- Нужно актуализировать документацию (не все и не всегда это делают)

Автоматизируем!



Системы управления конфигурациями

- Ansible – написан на python, имеет безагентную архитектуру
- Puppet – написан на ruby, имеет клиент-серверную архитектуру
- Chef – написан на ruby, имеет клиент-серверную архитектуру

Ansible

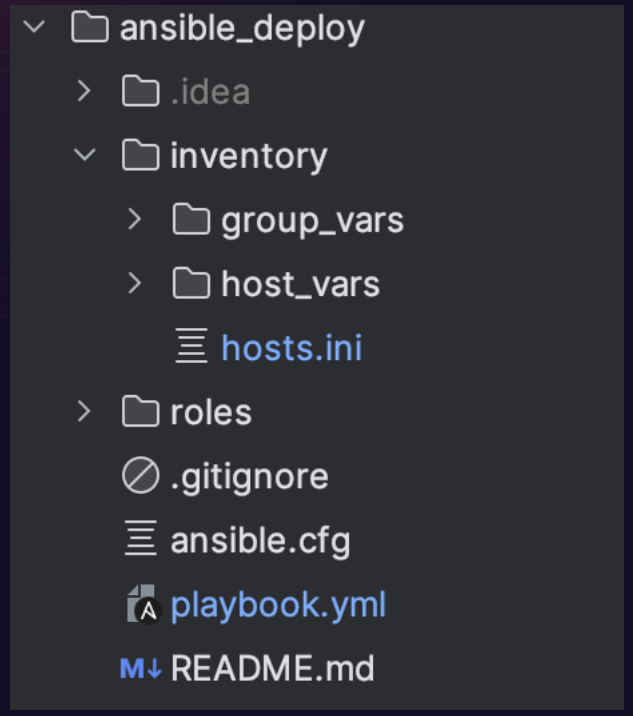
- Связь с хостами по ssh
- Написание конфигурации с помощью YAML
- Легкое освоение
- Jinja шаблонизатор

Ansible простой Playbook

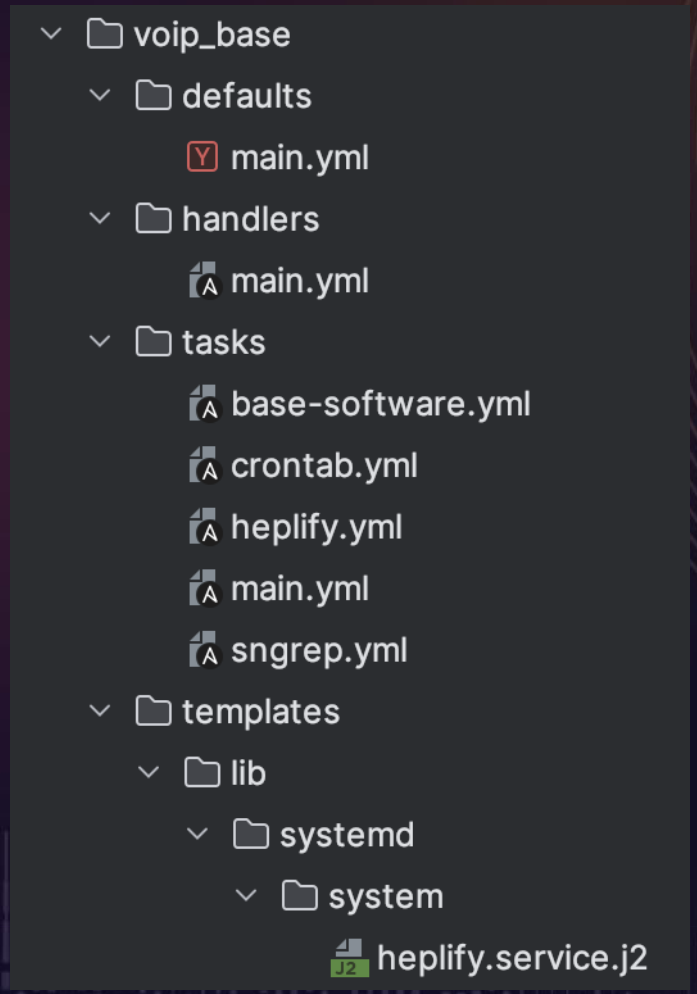
```yaml
---
# This playbook will install asterisk
tasks:
  - name: Install Asterisk
    apt:
      name: "{{ item }}"
      state: installed
    with_items:
     - asterisk
     - asterisk-config
     - asterisk-modules
     - asterisk-core-sounds-ru-wav

  - name: Make custom sip.conf file
    template:
      src: templates/etc/asterisk/sip.conf.j2
      dest: /etc/asterisk/sip.conf
    notify:
    - reload sip

handlers:
  - name: reload sip
    ansible.builtin.command: asterisk -rx "sip reload"
```

Ansible структура проекта

Проект



Роль



Ansible пример генерации пиров

Playbook.yml

```yaml
- hosts: asterisk
  vars:
    asterisk_bind_port: 7066
    # Описание пользователей АТС
    users:
    - name: 101
      pass: "password_for_101"
      redirect_to: 79191110000
    - name: 102
      pass: "password_for_102"
      redirect_to: 79191110000

    # Секция sip провайдеров
    uplinks:
    - login: 123456
      pass: password1
      type: novofon

    - login: 12345
      pass: password2
      type: multifon
      silence_ivr: "yes"

  tasks:
    - name: Configure sip.conf
      template:
        src: ../templates/sip.conf.j2
        dest: /etc/asterisk/sip.conf

```

```
[user_template](!)
context=internal
type=friend
host=dynamic

[multifon_template](!)
context=uplink
host=sbc.megafon.ru

[novofon_template](!)
context=uplink
host=sip.novofon.com

;;; внутренние sip'ы

{% for sip in users %}

[{{ sip.name }}](user_template)
defaultuser={{ sip.name }}
secret={{ sip.pass }}

{% endfor %}

;;; провайдеры
{% for uplink in uplinks %}

{% if uplink.type == 'novofon' %}
[{{ uplink.login }}](novofon_template)
{% elif uplink.type == 'multifon' %}
[{{ uplink.login }}](multifon_template)
{% endif %}
secret={{ uplink.pass }}
defaultuser={{ uplink.login }}
trunkname={{ uplink.login }}
fromuser={{ uplink.login }}
callbackextension={{ uplink.login }}

{% endfor %}
```

Ansible запуск из cli

ansible-lint playbook.yml

ansible-playbook playbook.yml --check

ansible-playbook playbook.yml

Ansible + (Jenkins || Gitlab)

```
pipeline {
    agent any

    stages {
        stage('Checkout') {
            steps {
                git branch: 'main', url: "git@github.com:example/jenkinsansiblebook.git"
            }
        }
        stage('Lint') {
          steps {
            sh 'ansible-lint playbook.yml'
          }
        }
        stage('Deploy') {
            steps {
                sh 'ansible-playbook playbook.yml'
            }
        }
    }
}
```

Итоги

- Определяемся со структурой проекта, и деплоем
- Заставляем писать роли/плейбуки
- Деплоим окружение 1 командой или 1 кнопкой
- Получаем стабильное воспроизводимое окружение

Спасибо за внимание!

Буду рад ответить на все ваши вопросы сейчас или свяжитесь со мной в будущем:

Гостьков Евгений (zt.team)

tg @gostkov