

DevOps

Л05. Ansible - basics

П05. Деплой веб-сервера

Виктор Моисеев
+7-902-83-145-30
t.me/v_paranoid
victorparanoid@gmail.com

План курса

1. Введение в DevOps
2. Базовое администрирование Linux
3. Системы контроля версионности кода (git)
4. Оркестровка (Ansible)
5. Контейнеризация (docker)
6. Микросервисная архитектура и оркестровка контейнеров (k8s)
7. Непрерывная интеграция и доставка (CI/CD, Github Actions, ArgoCD)
8. Инфраструктура как код (IaC, Terraform)
9. Мониторинг (Prometheus)

04. Ansible - basics

1. Форматы представления конфигурации
2. Что такое Ansible
3. Принцип работы
4. Терминология
5. Идемпотентность
6. Создание простого плейбука
7. Использование плейбука и отладка

XML

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE recipe>
<recipe name="хлеб" preptime="5min" cooktime="180min">
  <title>
    Простой хлеб
  </title>
  <composition>
    <ingredient amount="3" unit="стакан">Мука</ingredient>
    <ingredient amount="0.25" unit="грамм">Дрожжи</ingredient>
    <ingredient amount="1.5" unit="стакан">Тёплая вода</ingredient>
  </composition>
  <instructions>
    <step>
      Смешать все ингредиенты и тщательно замесить.
    </step>
    <step>
      Закрыть тканью и оставить на один час в тёплом помещении.
    </step>
  <!--
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="country">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="country_name"
          type="xs:string"/>
        <xs:element name="population"
          type="xs:decimal"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

INI

```
; for 16-bit app support
[386Enh]
woafont=dosapp.fon
EGA80WOA.FON=EGA80WOA.FON
EGA40WOA.FON=EGA40WOA.FON
CGA80WOA.FON=CGA80WOA.FON
CGA40WOA.FON=CGA40WOA.FON
```

```
[drivers]
wave=mmdrv.dll
timer=timer.drv
```

```
[mci]
```



```
; last modified 1 April 2001 by John Doe
```

[owner]

```
name = John Doe
organization = Acme Widgets Inc.
```

[database]

```
; use IP address in case network name resolution is
server = 192.0.2.62
port = 143
file = "payroll.dat"
```

[fruit.Date]

```
taste = novel
Trademark Issues="truly unlikely"
```

[fruit "Raspberry"]

```
anticipated problems = "logistics (fragile fruit)"
Trademark Issues=\
    possible
```

JSON

```
{
  "first_name": "John",
  "last_name": "Smith",
  "is_alive": true,
  "age": 27,
  "address": {
    "street_address": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postal_code": "10021-3100"
  },
  "phone_numbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "office",
      "number": "646 555-4567"
    }
  ],
  "children": [
    "Catherine",
    "Thomas",
    "Trevor"
  ],
  "spouse": null
}
```

YAML

```
--- # The Smiths
- {name: John Smith, age: 33}
- name: Mary Smith
  age: 27
- [name, age]: [Rae Smith, 4]
--- # People, by gender
men: [John Smith, Bill Jones]
women:
  - Mary Smith
  - Susan Williams
```

```
d: !!float 123
e: !!str 123
f: !!str Yes
```

```
---
receipt:      Oz-Ware Purchase Invoice
date:         2012-08-06
customer:
  first_name:  Dorothy
  family_name: Gale

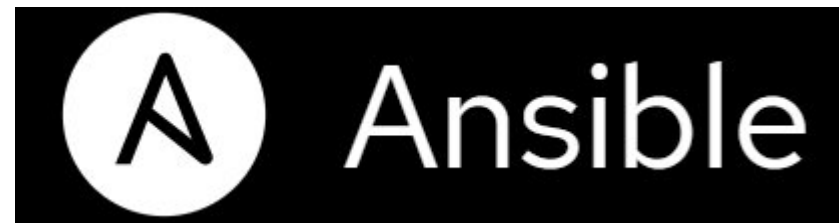
items:
  - part_no:   A4786
    descrip:   Water Bucket (Filled)
    price:     1.47
    quantity:  4

  - part_no:   E1628
    descrip:   High Heeled "Ruby" Slippers
    size:      8
    price:     133.7
    quantity:  1
```


Системы оркестровки и управления конфигурациями

Metrics	Chef	Puppet	Ansible	Saltstack
Availability	✓	✓	✓	✓
Ease of Setup	Not very easy	Not very easy	Easy	Not very easy
Management	Not very easy	Not very easy	Easy	Easy
Scalability	Highly Scalable	Highly Scalable	Highly Scalable	Highly Scalable
Configuration language	DSL(Ruby)	DSL(PuppetDSL)	YAML(Python)	YAML(Python)
Interoperability	High	High	High	High





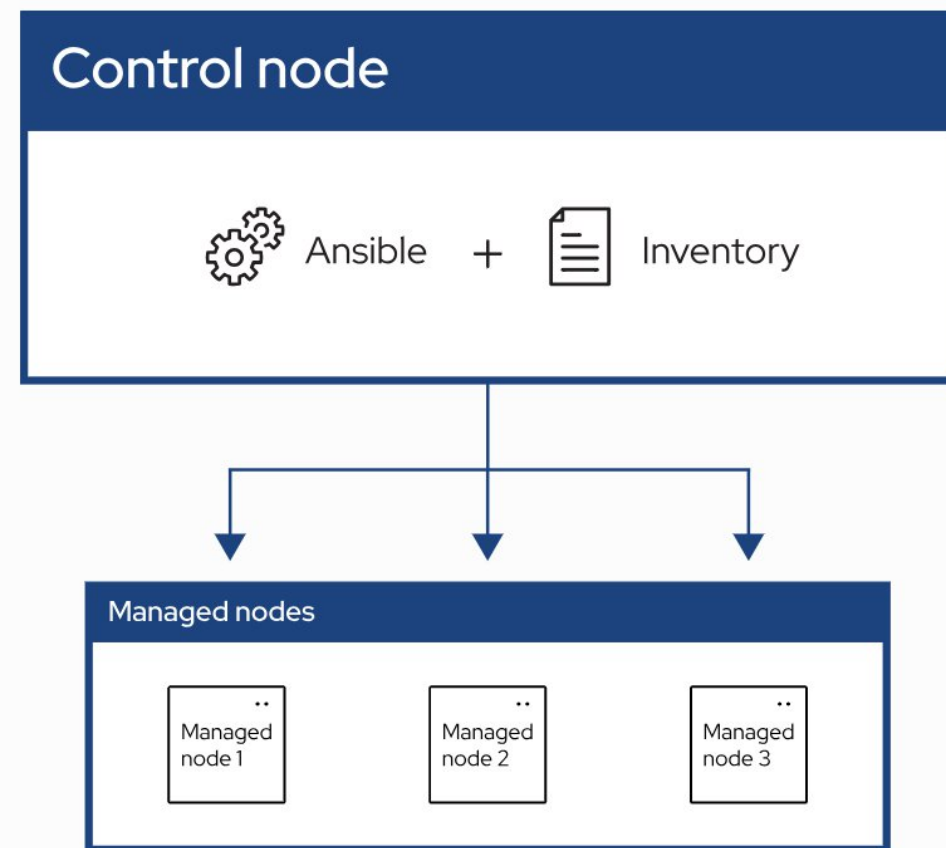
Управление конфигурацией
Automated provisioning
Оркестровка
Infrastructure as Code

Будьте внимательны с версиями Ансбла!

https://docs.ansible.com/ansible/latest/getting_started/index.html

Ansible – архитектура решения

- Push-method
- Agentless (обычно нужен python и ssh)
- Декларативность
- Идемпотентность



Ansible – термины

- **Control machine** - управляющий хост
- **Managed node** - управляемый хост
- **Inventory** - инвентарный файл, в котором описываются хосты, группы хостов, переменные
- **Playbook** - файл сценариев (play, role/task, handler)
- **Task** - задача, которая вызывает модуль с указанными параметрами
- **Module** - Модуль Ansible, который реализует определенные функции

Ansible – Inventory

```
inventory.ini
```

```
[webservers]
```

```
192.0.2.50
```

```
192.0.2.51
```

```
192.0.2.52
```

```
[db]
```

```
10.0.5.1
```

```
10.0.5.2
```

```
10.0.5.3
```

```
inventory.yml
```

```
myhosts:
```

```
  hosts:
```

```
    my_host_01:
```

```
      ansible_host: 192.0.2.50
```

```
    my_host_02:
```

```
      ansible_host: 192.0.2.51
```

```
    my_host_03:
```

```
      ansible_host: 192.0.2.52
```

```
webservers:
```

```
  hosts:
```

```
    webserver01:
```

```
      ansible_host: 192.0.2.140
```

```
      http_port: 80
```

```
    webserver02:
```

```
      ansible_host: 192.0.2.150
```

```
      http_port: 443
```

```
  vars:
```

```
    ansible_user: my_server_user
```

Ansible – Playbook

```
---
- name: play 1
  hosts: webserver

  tasks:
    - name: install Nginx
      apt:
        name: nginx-extras
        update_cache: yes
    - name: Start Nginx
      service: nginx
      state: enabled

- name: play 2
  hosts: db

  tasks:
    - name: install mysql
      apt:
        name: mysql
        update_cache: yes
    - name: Start mysql
      service: mysql
      state: enabled
```



Начинается с трёх минусов (YML)

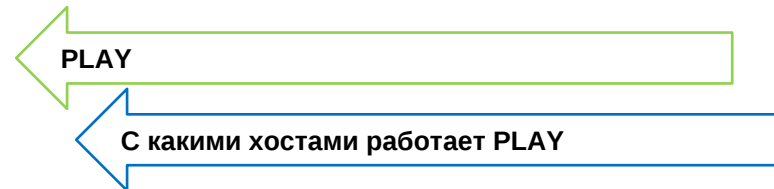
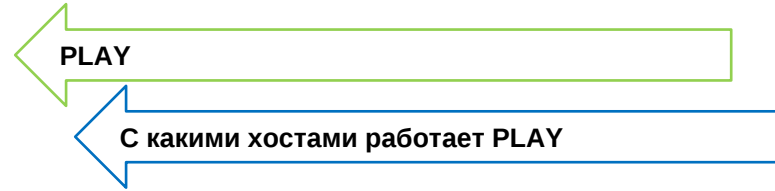
Ansible – Playbook

```
---
- name: play 1
  hosts: webserver

  tasks:
    - name: install Nginx
      apt:
        name: nginx-extras
        update_cache: yes
    - name: Start Nginx
      service: nginx
      state: enabled

- name: play 2
  hosts: db

  tasks:
    - name: install mysql
      apt:
        name: mysql
        update_cache: yes
    - name: Start mysql
      service: mysql
      state: enabled
```



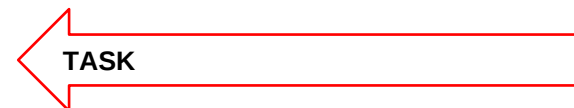
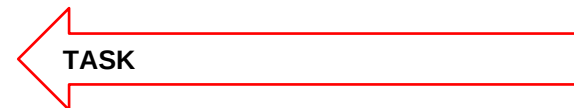
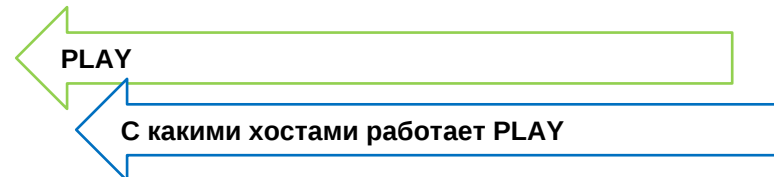
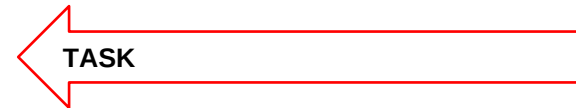
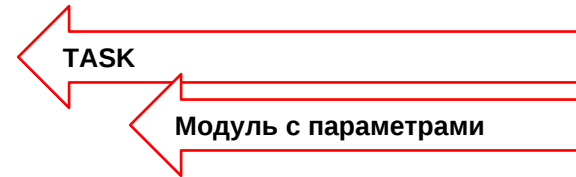
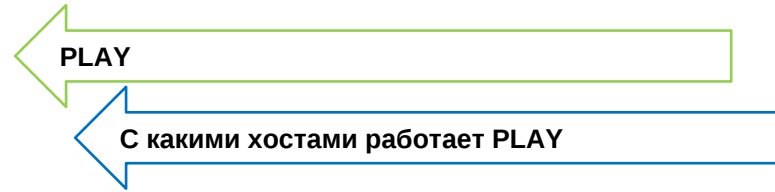
Ansible – Playbook

```
---
- name: play 1
  hosts: webserver

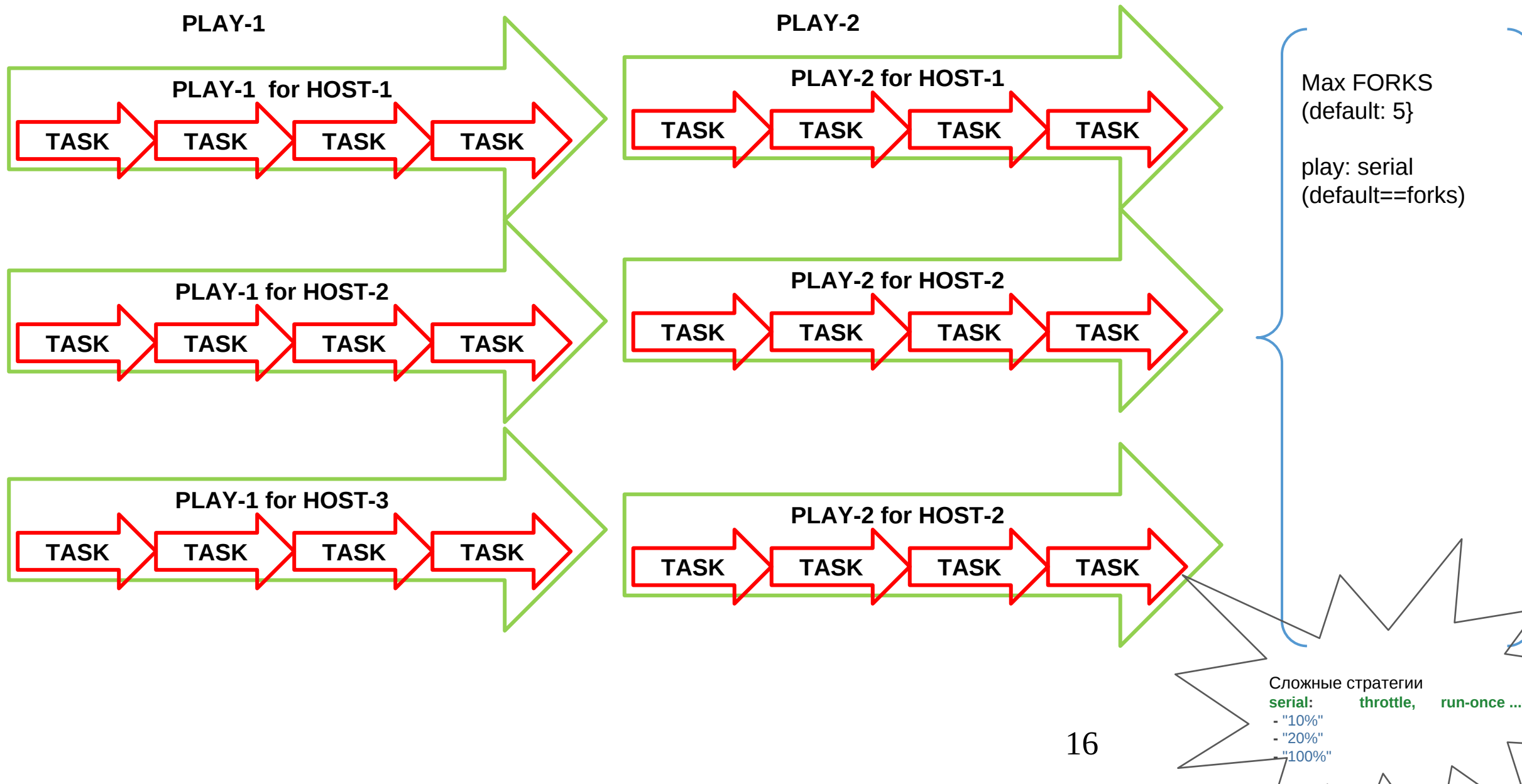
  tasks:
    - name: install Nginx
      ansible.builtin.apt:
        name: nginx-extras
        update_cache: yes
    - name: Start Nginx
      service: nginx
      state: enabled

- name: play 2
  hosts: db

  tasks:
    - name: install mysql
      ansible.builtin.apt:
        name: mysql
        update_cache: yes
    - name: Start mysql
      service: mysql
      state: enabled
```



Ansible – Стратегии выполнения



Ansible – Playbook

```
---  
- name: play 1  
  hosts: webserver  
  remote_user: runner  
  become: true  
  become_method: sudo  
  gather_facts: no  
  
tasks:  
- name: install Nginx  
  ansible.builtin.apt:  
    name: nginx-extras  
    update_cache: yes  
- name: Start Nginx  
  service: nginx  
  state: enabled  
- name: Print message  
  ansible.builtin.debug:  
    msg: Hello world
```



В какого юзера логиниться



Повышать ли привилегии и как



Собирать ли информацию о системе



Отладка на контрольной ноде

Ansible – Пример

Тестовый прогон без изменений:

```
ansible-playbook ./playbook.yml -i ./inventory.ini --check --diff
```

Продуктовый прогон:

```
ansible-playbook ./playbook.yml -i ./inventory.ini
```

--ask-become-pass – если на удаленной стороне требуется ввод пароля

Ansible ...



П04. Деплой веб-сервера с помощью Ansible

Сделать ansible-плейбук, который установит на целевую машину nginx и скопирует туда индексный файл

1. Сделать вторую виртуалку

1. Склонировать первую вм с опцией «сгенерировать новые мак-адреса»
2. Создать в virtual box – инструменты – Сеть NAT
3. Обе вм переключить в новую Сеть NAT
4. *(При необходимости) задать статические IP адреса на обеих вм*
5. *(При необходимости) Организовать проброс портов 22 на обе вм и 80 на вторую вм*

2. Создать и скопировать с первой на вторую вм SSH ключ (в пользователя runner)

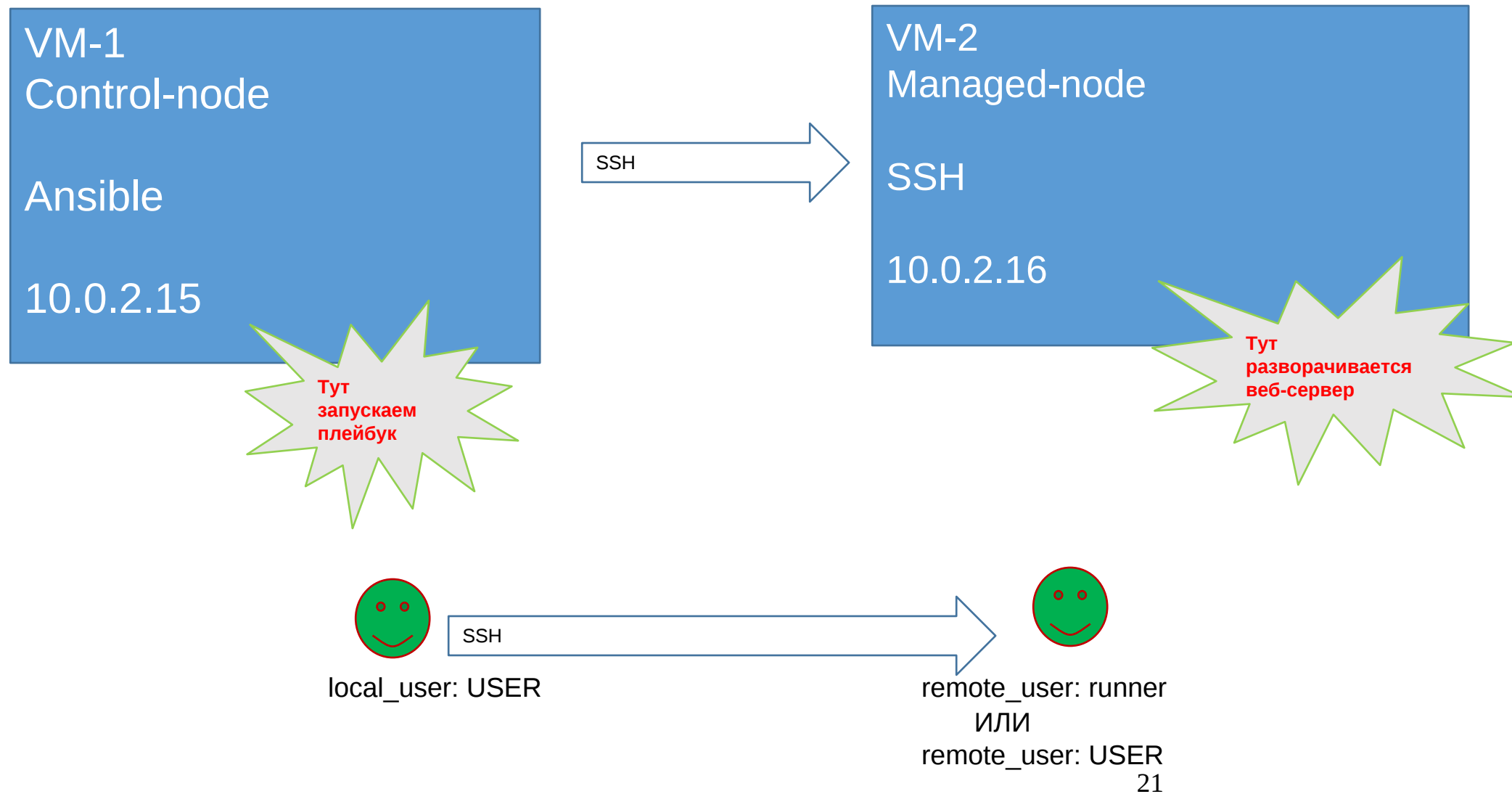
3. На второй вм создать юзера runner с правами sudo без пароля

4. На первой вм

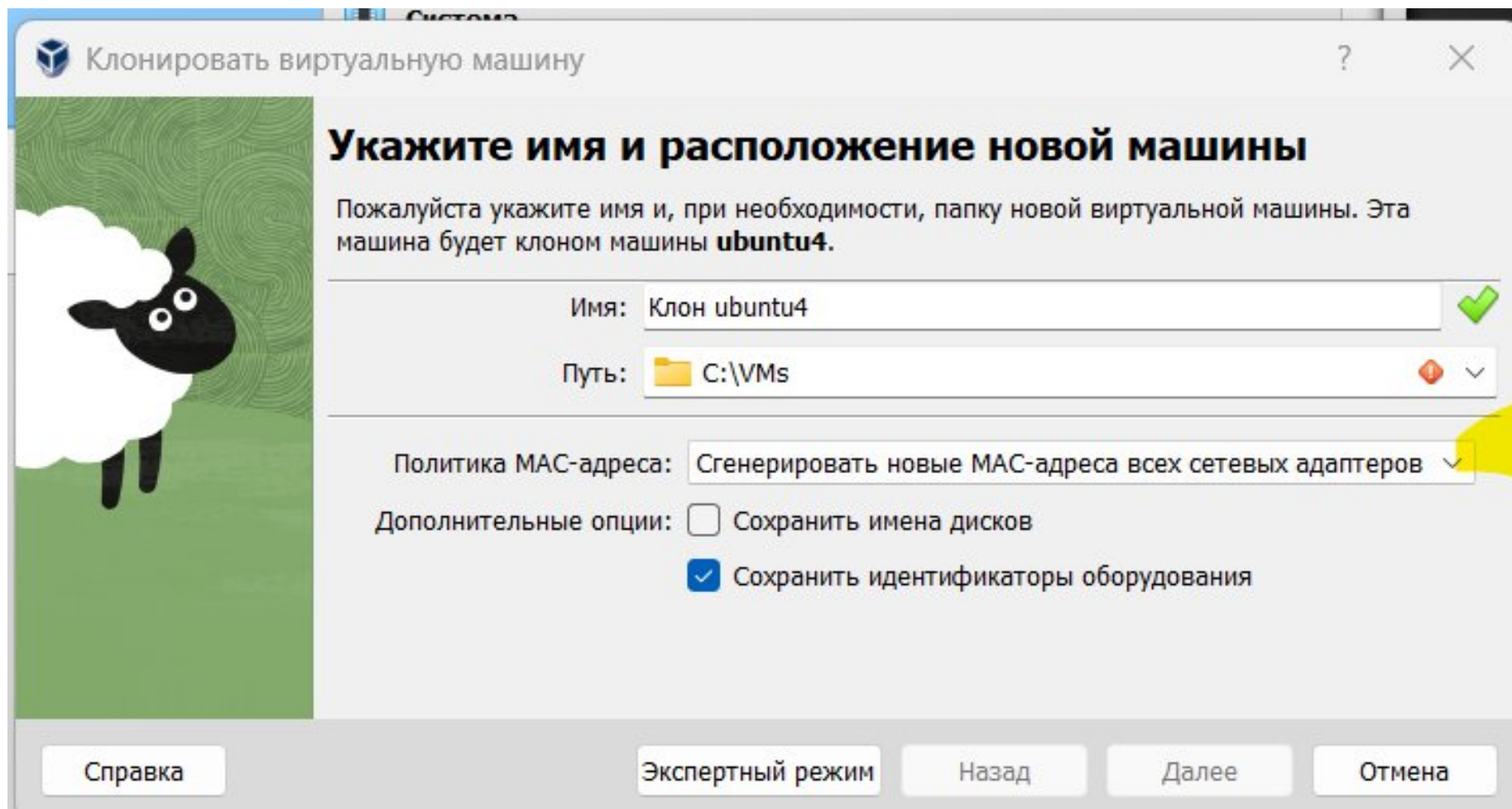
1. установить ансибл
2. Написать плейбук – install nginx
3. Создать файл inventory с адресом второй вм
4. Положить шаблон index.html

5. Выполнить плейбук

Схема взаимодействия



Клонирование вм с рандомизацией MAC-адресов



Клонировать виртуальную машину

Укажите имя и расположение новой машины

Пожалуйста укажите имя и, при необходимости, папку новой виртуальной машины. Эта машина будет клоном машины **ubuntu4**.

Имя: Клон ubuntu4 ✓

Путь: C:\VMs ⚠

Политика MAC-адреса: Сгенерировать новые MAC-адреса всех сетевых адаптеров ✓

Дополнительные опции:

- ☐ Сохранить имена дисков
- ☒ Сохранить идентификаторы оборудования

Справка Экспертный режим Назад Далее Отмена

Настройка общей сети и проброс портов

Oracle VM VirtualBox Менеджер

Файл Машина Сеть Справка

Инструменты

ubuntu Выключена

ubuntu3 Выключена

Создать Удалить Свойства

Виртуальные сети хоста Сети NAT Облачные сети

Имя	IPv4 префикс	IPv6 префикс	DHCP сервер
NatNetwork	10.0.2.0/24		Включен

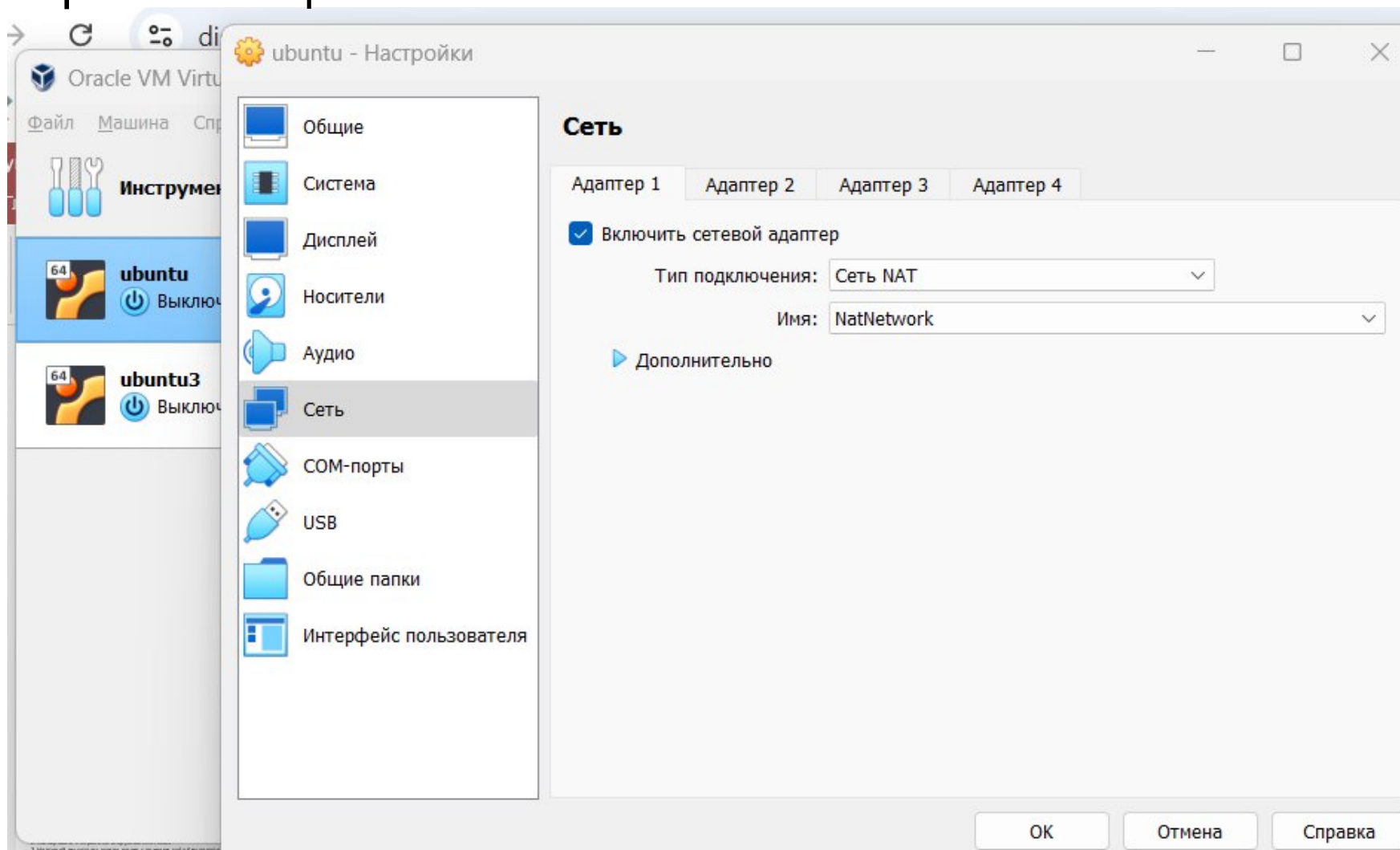
Основные опции Проброс портов

IPv4 IPv6

Имя	Протокол	Адрес хоста	Порт хоста	Адрес гостя	Порт гостя
Rule 1	TCP	127.0.0.1	2222	10.0.2.15	22
Rule 2	TCP	127.0.0.1	80	10.0.2.16	80

Применить Сбросить

Настройка общей сети



Настройка сети внутри вм – переключение с DHCP на статику (optional)
Можно поправить уже существующий файл yml в этом каталоге

```
sudo nano /etc/netplan/xxxxxxx.yml
```

На одной 10.0.2.15

На второй 10.0.2.16

```
# This file is generated from information provided by t
# to it will not persist across an instance reboot. To
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg
# network: {config: disabled}
network:
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [ 10.0.2.15/24 ]
      routes: [ { to: default, via: 10.0.2.1 } ]
      nameservers:
        addresses: [ 1.1.1.1 ]
  version: 2
```

```
sudo netplan apply
```

Проверка применения настроек

```
ip addr
```

На второй вм создаем пользователя runner

```
victor@ubuntu:~$ sudo useradd -m runner
victor@ubuntu:~$ sudo passwd runner
New password:
Retype new password:
passwd: password updated successfully
victor@ubuntu:~$ sudo usermod -a -G sudo runner
victor@ubuntu:~$
```

```
victor@ubuntu:~$ sudo nano /etc/sudoers
victor@ubuntu:~$
```

>>>>

```
GNU nano 7.2 /etc/sudoers *

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

victor  ALL=(ALL) NOPASSWD:ALL
runner  ALL=(ALL) NOPASSWD:ALL

# See sudoers(5) for more information on "@include" di

@includedir /etc/sudoers.d

^G Help      ^O Write Out ^W Where Is  ^K Cut
^X Exit      ^R Read File ^\ Replace   ^U Paste
```

С первой вм копируем пользователя на вторую вм
В целевого пользователя (свой, либо runner)

```
victor@ubuntu:~/lab-ansible-5$ ssh-copy-id runner@10.0.2.16
```

Если ключа нет, то его надо создать
ssh-keygen


```
sudo apt install ansible
```

```
victor@ubuntu:~$ ansible-playbook --version
ansible-playbook [core 2.16.3]
  config file = None
  configured module search path = ['/home/victor/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/victor/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible-playbook
  python version = 3.12.3 (main, Sep 11 2024, 14:38:01) [GCC 13.2.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
```


Заготовки под плейбук и инвентарь

```
victor@ubuntu:~$ mkdir lab-ansible
victor@ubuntu:~$ cd lab-ansible/
victor@ubuntu:~/lab-ansible$
victor@ubuntu:~/lab-ansible$
victor@ubuntu:~/lab-ansible$ touch playbook.yml
victor@ubuntu:~/lab-ansible$ touch inventory.ini
victor@ubuntu:~/lab-ansible$
```

GNU nano 7.2

inventory.ini *

[webserver]

10.0.2.16

Вариант подключения текущим юзером (без указания remote user)

**Обращайте внимание на отступы в секциях yamI*

```
GNU nano 7.2                                playbook.yml
---
- name: Deploy nginx
  hosts: webservers
  become: yes
  become_method: sudo

  tasks:
  - name: Install nginx
    ansible.builtin.apt:
      name: nginx
      state: latest
      update_cache: yes
```

Тестовый прогон плейбука

```
victor@ubuntu:~/lab-ansible$ ansible-playbook playbook.yml -i inventory.ini --check

PLAY [Deploy nginx] *****

TASK [Gathering Facts] *****
ok: [10.0.2.16]

TASK [Install nginx] *****
ok: [10.0.2.16]

PLAY RECAP *****
10.0.2.16 : ok=2    changed=0    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0
```

Боевой прогон плейбука

```
victor@ubuntu:~/lab-ansible$ ansible-playbook playbook.yml -i inventory.ini

PLAY [Deploy nginx] *****

TASK [Gathering Facts] *****
ok: [10.0.2.16]

TASK [Install nginx] *****
ok: [10.0.2.16]

PLAY RECAP *****
10.0.2.16 : ok=2    changed=0    unreachable=0    failed=0
cued=0    ignored=0

victor@ubuntu:~/lab-ansible$
```

Заготовка индексного файла на первой вм

```
GNU nano 7.2                                index.html *
<HTML>
  <BODY>
    <h1>Hello from Ansible playbook!</h1>
  </BODY>
</HTML>
```

Дописываем
плейбук

```
---  
- name: Deploy nginx  
  hosts: webserver  
  become: yes  
  become_method: sudo  
  
  tasks:  
    - name: Install nginx  
      ansible.builtin.apt:  
        name: nginx  
        state: latest  
        update_cache: yes  
  
    - name: Copy index  
      ansible.builtin.copy:  
        src: ./index.html  
        dest: /var/www/html/index.nginx-debian.html  
        owner: root  
        group: root  
        mode: '0666'
```



```
victor@ubuntu:~/lab-ansible$ ansible-playbook playbook.yml -i inventory.ini

PLAY [Deploy nginx] *****
***

TASK [Gathering Facts] *****
***
ok: [10.0.2.16]

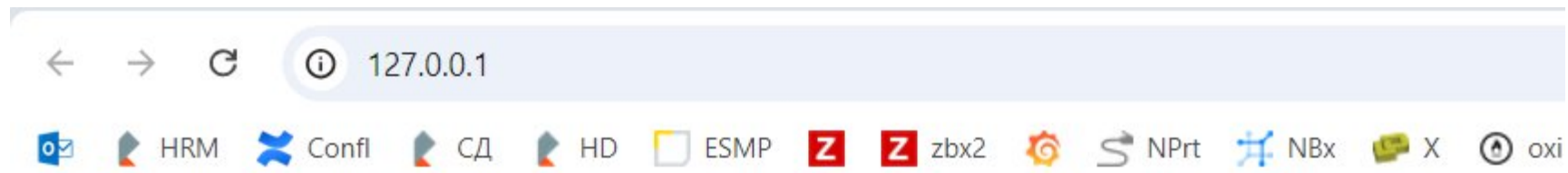
TASK [Install nginx] *****
***
ok: [10.0.2.16]

TASK [Copy index] *****
***
changed: [10.0.2.16]

PLAY RECAP *****
***
10.0.2.16      : ok=3    changed=1    unreachable=0    failed=0
  skipped=0    rescued=0    ignored=0
```

Проверка: с первой машины обращаемся к новому веб серверу на второй машине

```
victor@ubuntu:~/lab-ansible$ curl http://10.0.2.16
<HTML>
  <BODY>
    <h1>Hello from Ansible playbook!</h1>
  </BODY>
</HTML>
victor@ubuntu:~/lab-ansible$
```



Hello from Ansible playbook!