|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| *Федеральное государственное бюджетное образовательное учреждение высшего образования* ***«МИРЭА – Российский технологический университет»***  **РТУ МИРЭА** |

Институт Информационных технологий (ИТ)

Кафедра Математического обеспечения и стандартизации информационных технологий (МОСИТ)

|  |
| --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 5** |
| **по дисциплине** |
| **«Технология разработки программных приложений»**  **Тема: «Системы конфигурационного управления»** |

|  |  |
| --- | --- |
| Выполнил студент группы ИКБО-11-22 | Гришин А.В. |
| Принял преподаватель | Копанева А.А |

|  |  |  |
| --- | --- | --- |
| Практическая работа выполнена | «\_\_»\_\_\_\_\_\_\_202\_\_г. | *(подпись студента)* |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_202\_\_ г. | *(подпись руководителя)* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Москва 2024г

**Оглавление**

[**Цель работы** 3](#_Toc166845333)

[**Вариант практической 14: gpg.** 3](#_Toc166845334)

[**Подготовка инфраструктуры** 3](#_Toc166845335)

[**Установка Ansible** 4](#_Toc166845336)

[**Настройка Ansible** 5](#_Toc166845337)

[**Конфигурация хостов при помощи Ansible** 6](#_Toc166845338)

[**Роли Ansible** 10](#_Toc166845339)

[**Задание** 12](#_Toc166845340)

[**Вывод** 15](#_Toc166845341)

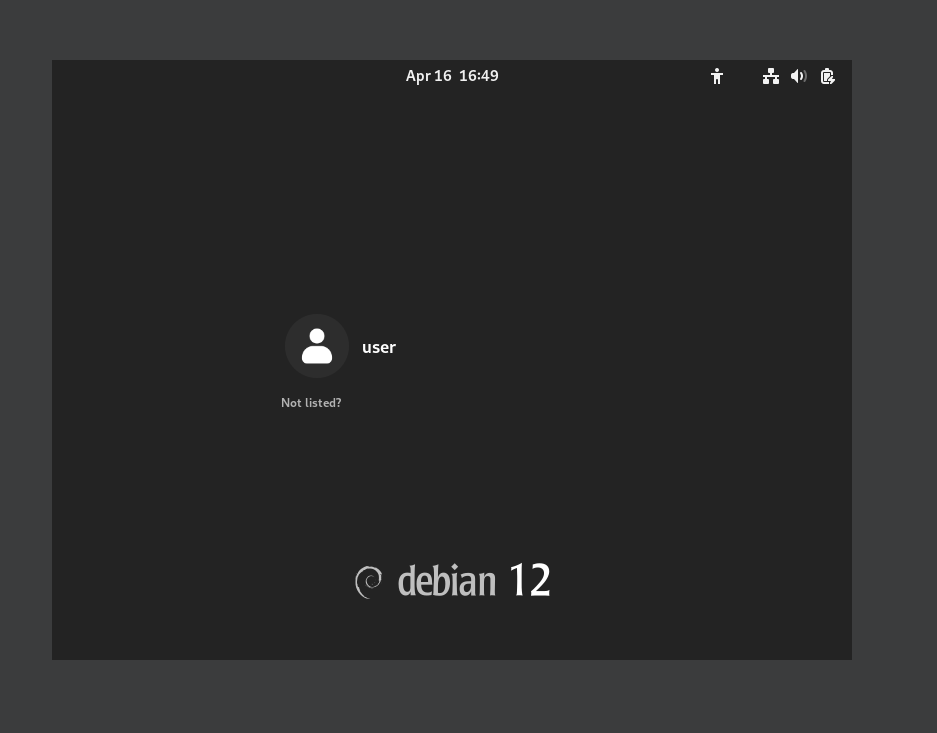
# **Цель работы**

Получить навыки настройки вычислительной инфраструктуры при помощи системы конфигурационного управления Ansible

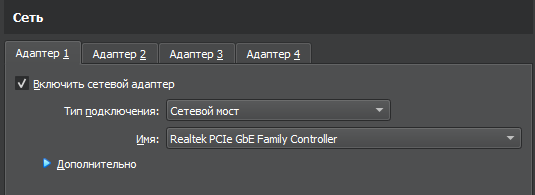
# **Вариант практической 14: gpg.**

# **Подготовка инфраструктуры**

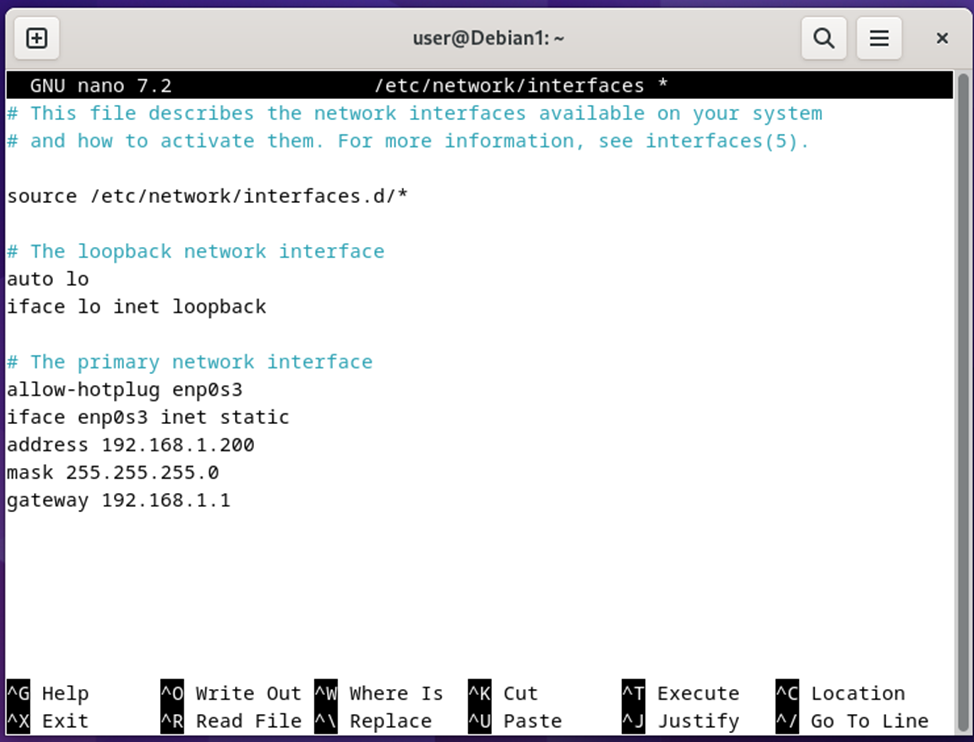
Вначале создали виртуальную машину Debian. Виртуальная машина необходима для практической работы.

  
Рисунок 1 – Установка ОС

Теперь настроим доступ к сети. Изменим в VirtualBox тип сетевого подключения на сетевой мост. Из-за этого виртуальная машина будет находится на уровне с физическими устройствами в локальной сети.

  
Рисунок 2 – Настройка сети VM

Будем производить настройку при помощи файла /etc/network/interfaces. Открыв его в редакторе nano, увидите набор параметров для двух интерфейсов lo и что-то похожее на enp0s3. Второй интерфейс необходимо настроить по следующему подобию. После заполнения перезапустите машину командой reboot. После этого установим ssh-сервер. Это можно сделать при помощи команды: apt install ssh

  
Рисунок 3 – Настройка интерфейса

Cоздадим управляющую машину для Ansible. Будем пользоваться WSL2. И перейдем к конфигурации машин под Ansible.

# **Установка Ansible**

Установим Ansible на управляющую машину при помощи команд: sudo apt install ansible.

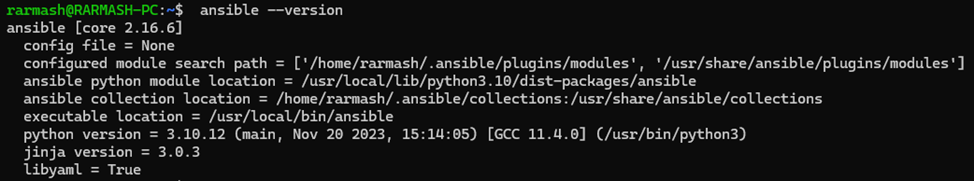
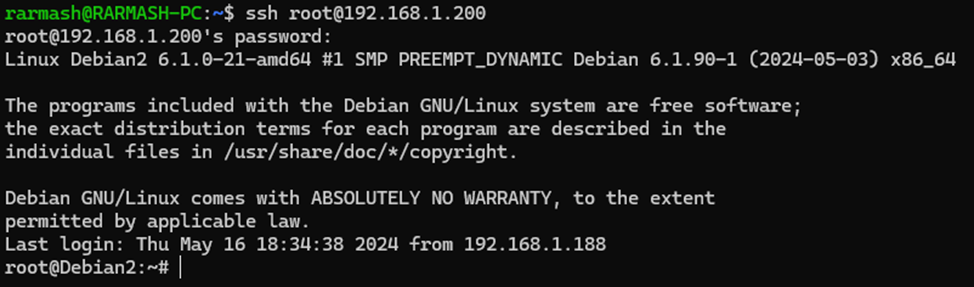


Рисунок 4 – Установка Ansible

После подключимся к управляемым машинам при помощи ssh. **ssh root@ip\_address.**

  
Рисунок 5 – Подключение к управляемой машине

Сгенерируем ключ для авторизации без пароля и передадим его на обе машины при помощи команд **ssh-keygen** и **ssh-copy-id root@ip\_address** соответственно.

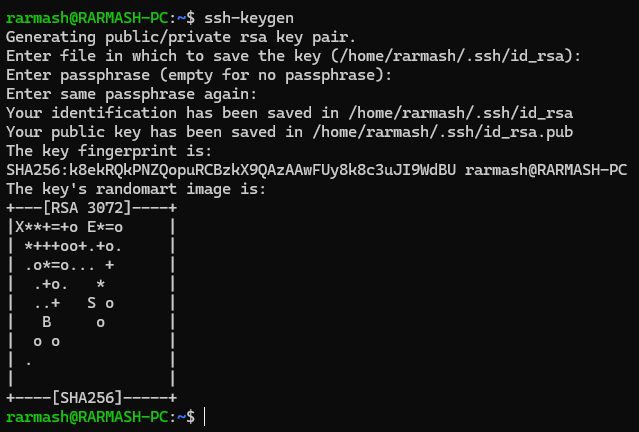


Рисунок 6 – генерация ключа

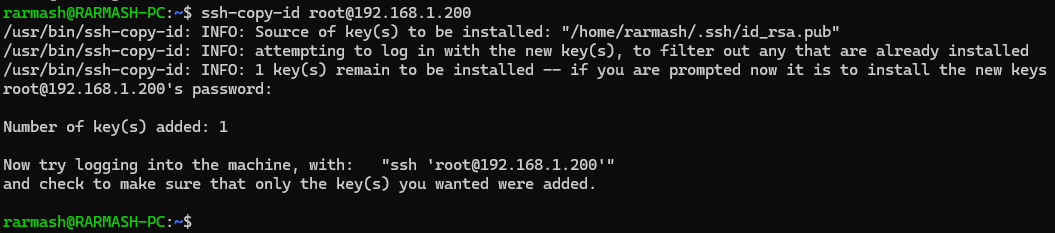


Рисунок 7 – передача ключа на машину

# **Настройка Ansible**

Настроим inventory-файл. Создадим директорию ansible в домашнем каталоге пользователя, будем считать эту директорию рабочей, и в ней создадим файл hosts.

  
Рисунок 8 – Файл Hosts

Проверим работу Ansible при помощи: **ansible -i ./hosts -m ping all.** Данная команда выполнит команду ping для всех хостов в inventory и выдаст результат выполнения.

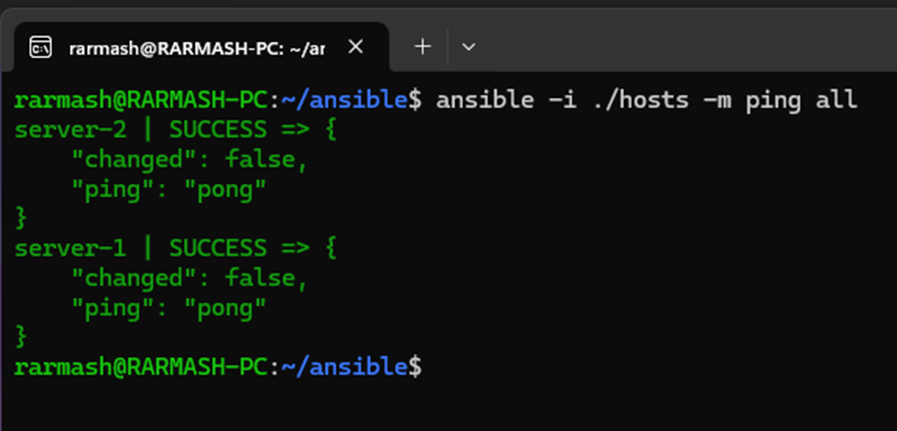


Рисунок 9 – Результат выполнения команды

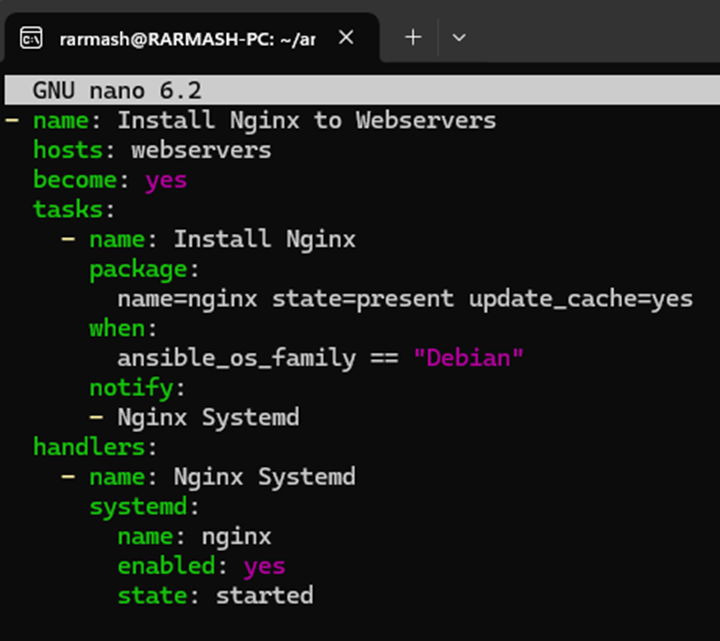
# **Конфигурация хостов при помощи Ansible**

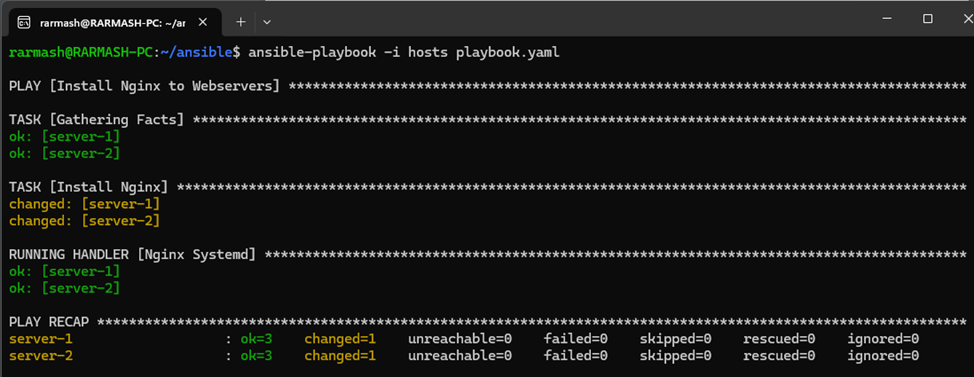
Факты – это параметры, которыми можно управлять в реализуемых сценариях.

Соберем возможные факты с управляемого хоста server-1: **ansible server-1 -i ./hosts -m command -m setup**.

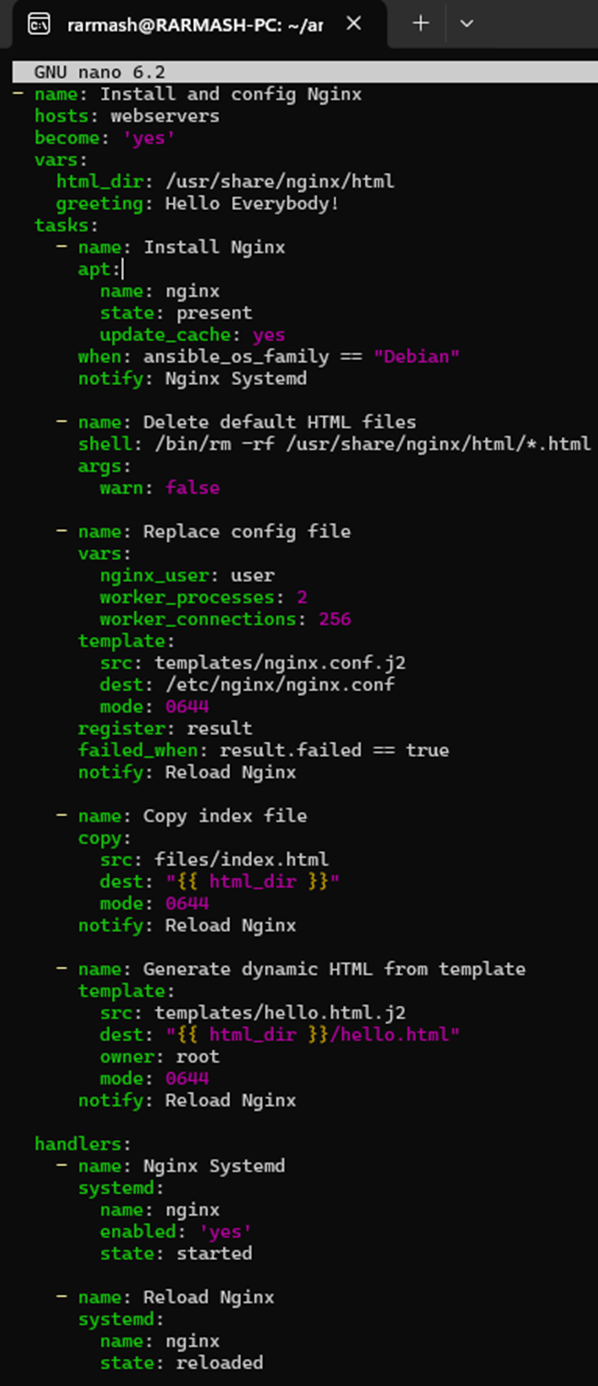
  
Рисунок 10 – Результат выполнения команды

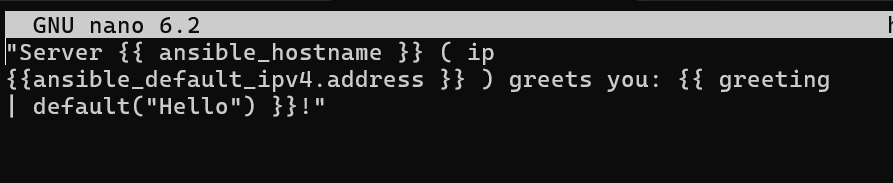
Playbook – это конфигурационный сценарии, написанный на языке YAML, который впоследствии будет выполнятся на управляемых хостах. Напишем playbook, который установит веб-сервер Nginx на управляемые хосты.

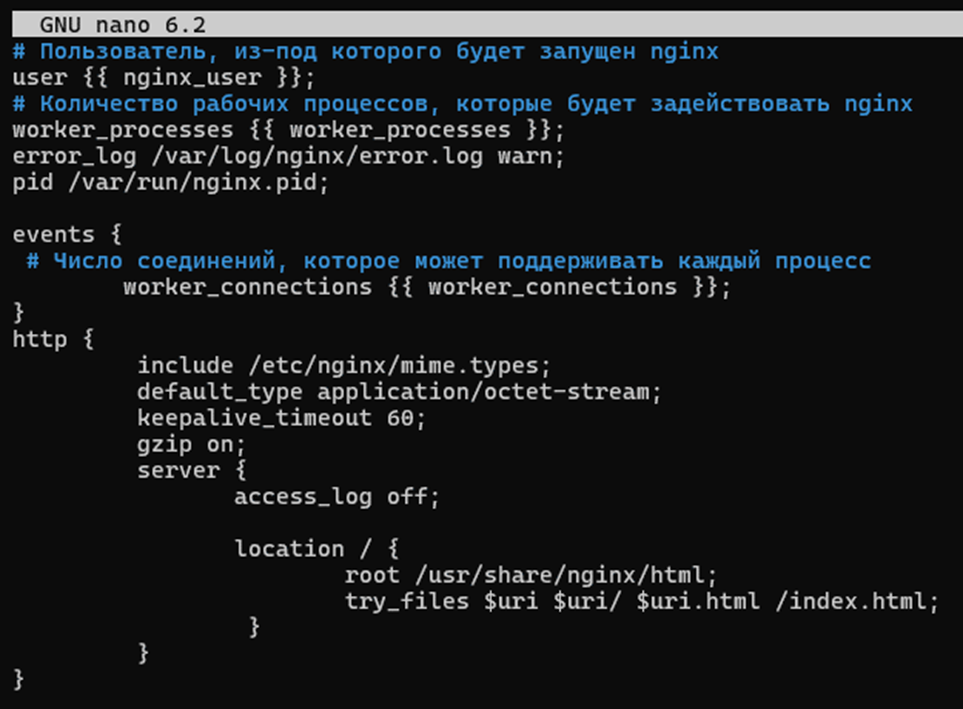
  
Рисунок 11 – Первый playbook

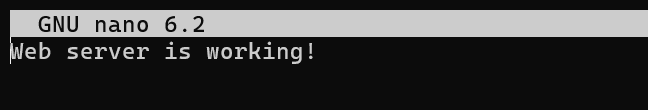
  
Рисунок 12 – Результат выполнения playbook

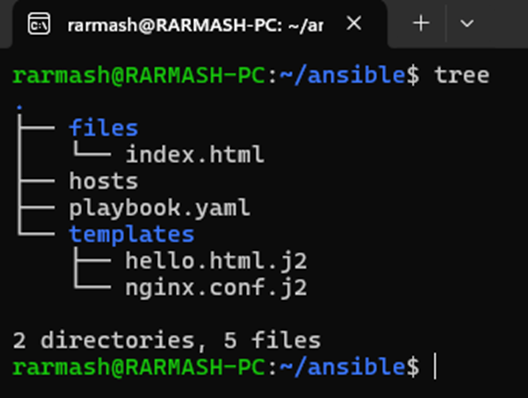
Напишем более сложный playbook и рассмотрим отдельные его составляющие.

  
Рисунок 13 – Файл playbook.yaml

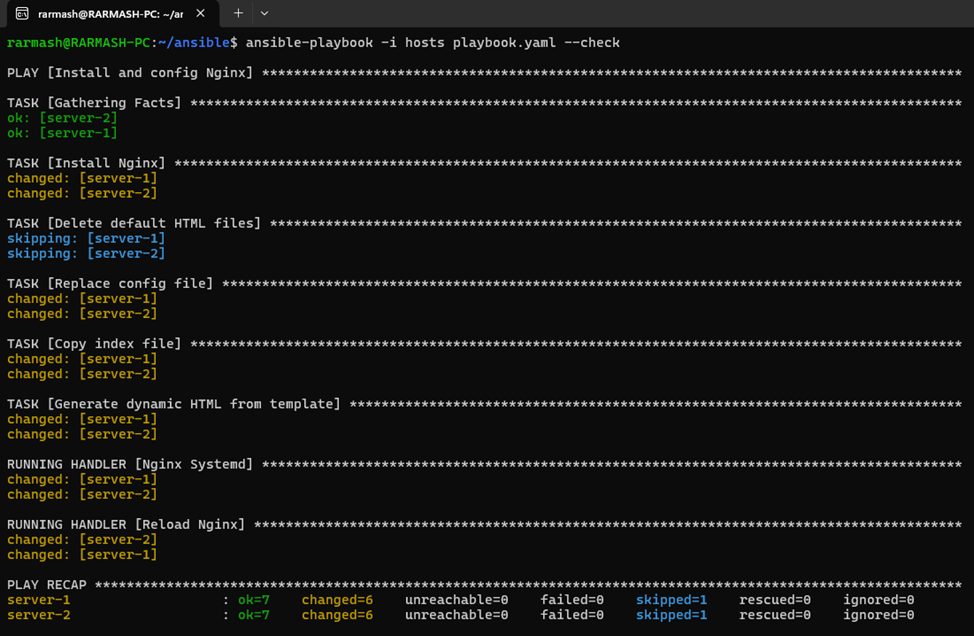
  
Рисунок 14 – Файл hello.html.j2

  
Рисунок 15 – Файл nginx.conf.j2

  
Рисунок 16 – файл index.html

  
Рисунок 17 – Cодержимое директории ansible

Воспользуемся командой для запуска playbook’а с пробным прогоном, который позволит проверить корректность написанного playbook’а без внесения изменений на целевые узлы.

  
Рисунок 18 – Вывод тестового прогона

Теперь внесём изменения в конфигурацию первого сервера, убрав ключ --check из предыдущей команды, а затем запросим базовую страницу при помощи утилиты curl.

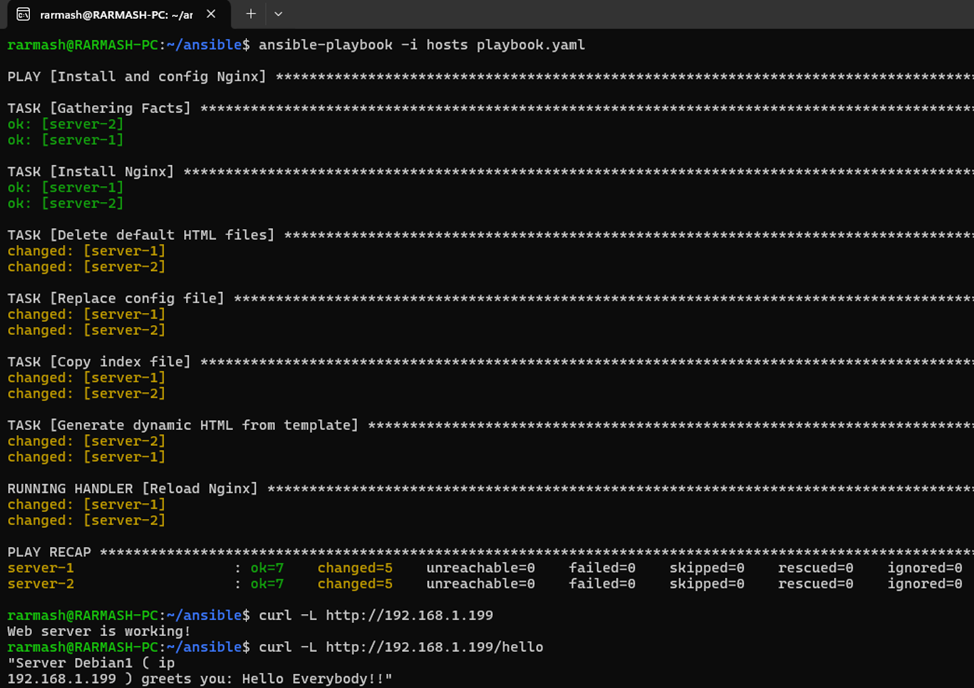


Рисунок 19 – Вывод прогона

# **Роли Ansible**

Данный механизм позволяет систематизировать конфигурации путём выделения каждого механизма (задач, переменных, обработчиков и т. д.) в отдельные части.

Ansible имеет похожий на GitHub сервис, называемый Ansible Galaxy. Там находится множество ролей для Ansible, которыми можно воспользоваться. Для использования данного сервиса в Ansible встроена команда ansible-galaxy.

Загрузим в систему роль для установки MySQL.

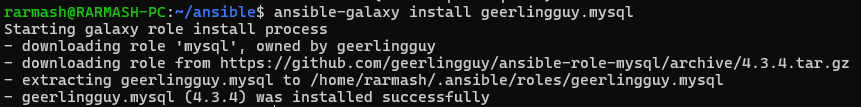


Рисунок 20 – загрузка роли для установки MySQL

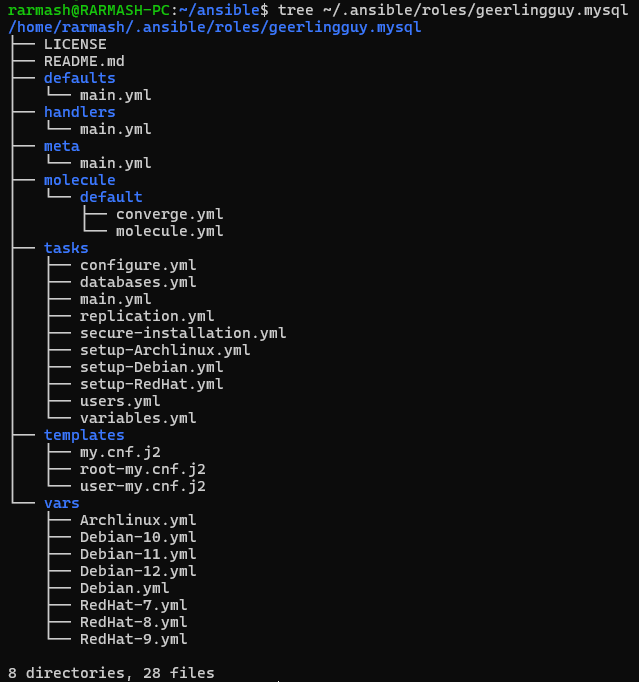


Рисунок 21 – содержимое папки установленной роли

Попробуем создать свою роль для установки ранее разобранного playbook’а nginx. Сперва вернёмся в рабочую папку со всеми файлами для Ansible и создадим там директорию roles. После чего перейдём в эту директорию и инициируем роль стандартной структуры при помощи команды ansible-galaxy init nginx.

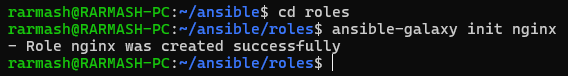
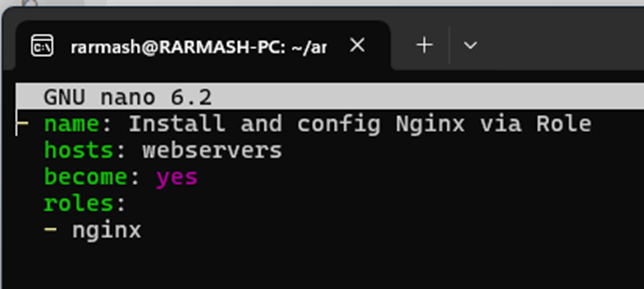
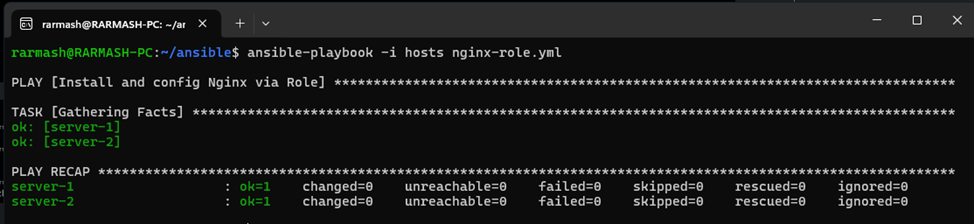


Рисунок 22 – инициализация роли стандартной структуры

Перейдём к созданию роли. Заполним соответствующие файлы данными из секций playbook’а и соответствующие директории ранее созданными файлами. Это значит, что в директории tasks файл main.yml должен быть заполнен данными из секции tasks. В директории vars – из секции vars. В директории handlers – из секции handlers. Также файлы из директорий files и templates должны быть перемещены в директории files и templates в папке роли.

  
Рисунок 23 – реализованный nginx-role.yml

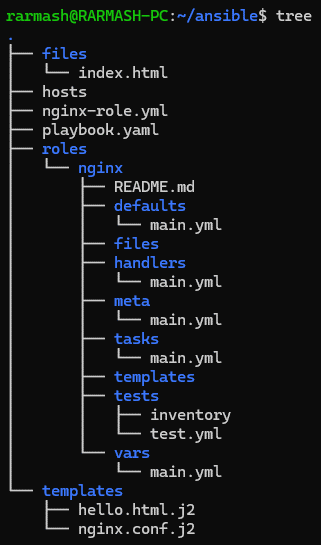
Роль отработала и показала полное соответствие текущего состояния.

  
Рисунок 24 – Вывод работы роли

# **Задание**

Написать роль для запуска сервера nginx, написать playbook для применения роли, провести тестовый запуск playbook’а, в случае успешного прохождения теста, применить playbook к серверам. Необходимо добавить переменную, содержащую ФИО, номер группы и номер варианта. Данная переменная должна выводиться в шаблонный файл nginx. Установка пакета выполняется при помощи модуля APT, используемого для установки nginx в базовой роли

Добавьте в playbook task по установке пакета согласно варианту: 14. gpg

  
Рисунок 25 – Роль для запуска сервера nginx

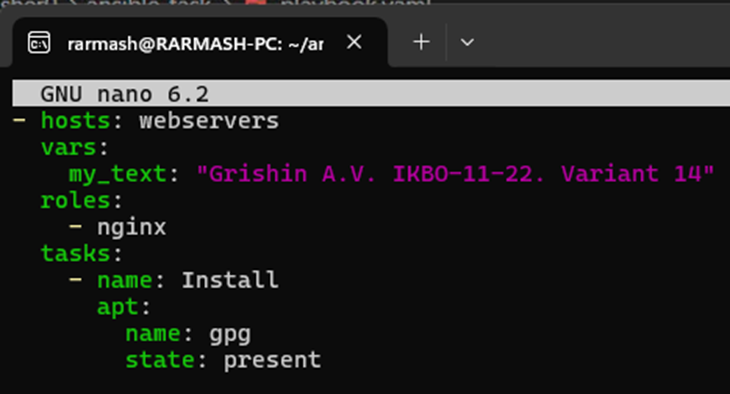
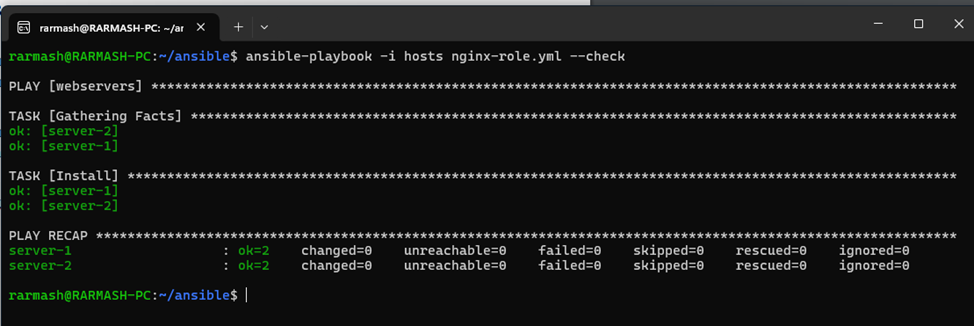
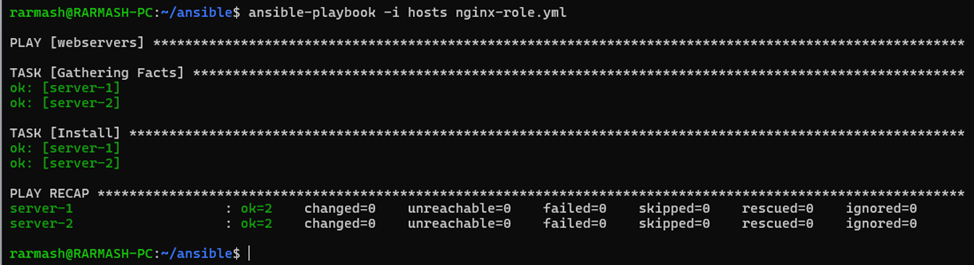


Рисунок 26 – Содержимое файла nginx-role.yml

  
Рисунок 27 – Тестовый прогон

  
Рисунок 28 – Применение playbook к серверу

# **Вывод**

В ходе практической работы были получены навыки настройки вычислительной инфраструктуры с помощью Ansible.