МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра систем автоматизации управления

**Управление конфигурациями в Ansible**

Отчет по лабораторной работе №4  
по дисциплине

«Управление IT-проектами»

Выполнил студент гр. ИТб-4302-02-00 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Вершинин П.А./

(Подпись)

Руководитель ст. преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Родионов К.В./

(Подпись)

Работа защищена с оценкой «\_\_\_\_\_\_\_\_\_\_\_» «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2023 г.

Цель: получить практические навыки по использованию инструмента «Ansible» для автоматизации настройки и управления конфигурациями Linux-серверов.

В ходе работы необходимо выполнить следующее задание:

* изменить файлы конфигурации и разбить хосты на группы;
* создать YAML-файлы с переменными для каждой группы хостов;
* запустить «playbook» для установки веб-сервера и создания страницы «index.html»;
* разделить задачи по установке веб-сервера на роли.

Основные изменения в файлах конфигурации «Ansible» представлены на рисунках 1, 2.

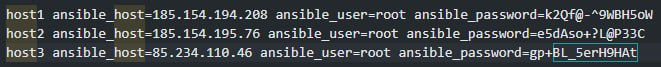


Рисунок 1 – Изменения в инвентарном файле «hosts»

В данном файле «hosts» прописаны адреса созданных Linux-систем в качестве управляемых серверов. В каждой строчке записывается имя хоста, в параметре «ansible\_host» IP-адрес, в параметрах «ansible\_user» и «ansible\_password» записывается логин и пароль соответственно пользователей управляемых систем.

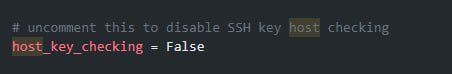


Рисунок 2 – Изменения в файле «ansible.cfg»

Строка «host\_key\_checking = False» в файле «ansible.cfg» отвечает за проверку ключей хостов при подключении к ним. Когда эта опция установлена в «False», «Ansible» не будет выполнять проверку хостовых ключей SSH.

Для проверки доступности хостов и их готовности к выполнению задач «Ansible» используем команду «ansible all -i hosts -m ping».

Результат выполнения представлен на рисунке 3.

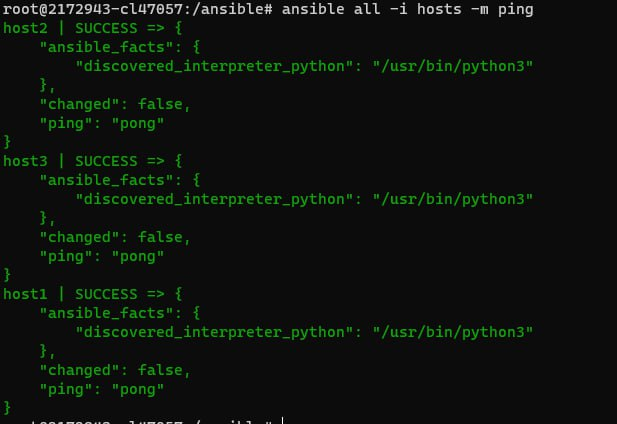


Рисунок 3 – Результат выполнения модуля «ping» на всех хостах

Данная команда выполняет «ping» на всех хостах, перечисленных в файле инвентаря «hosts», и выводит результат.

Для обращения к определенному множеству хостов в «Ansible» существует возможность создания групп. Для этого в инвентарном файле в квадратных скобках записывается название группы, а ниже адреса хостов.

Создание групп представлено на рисунке 4.

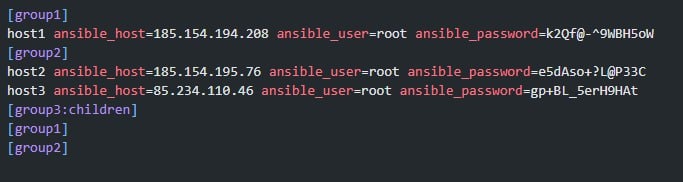


Рисунок 4 – Создание групп

Для проверки правильности написания необходимо создать файл и выполнить команду «ansible group2 -m copy -a "src=file.txt dest=/home mode=0777" -b».

Результат выполнения команды представлен на рисунке 5.

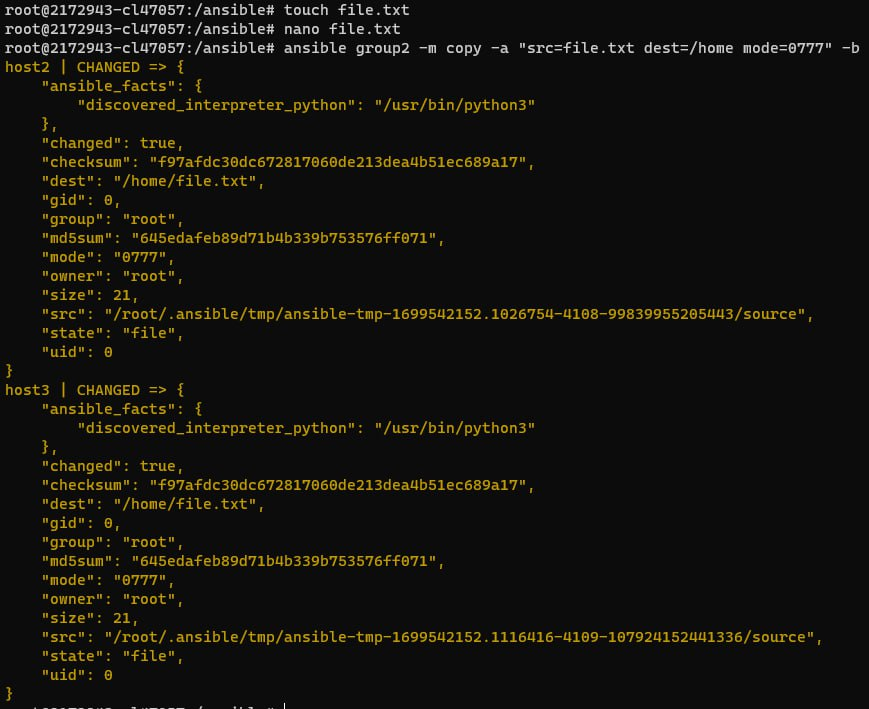


Рисунок 5 – Создание и копирование текстового файла

Данная команда скопирует файл и отправит на хосты, которые включены в группу.

Создадим «yaml» файлы для каждой группы с различными переменным, и выведем все переменные «group\_name» для каждой группы (рис. 6).

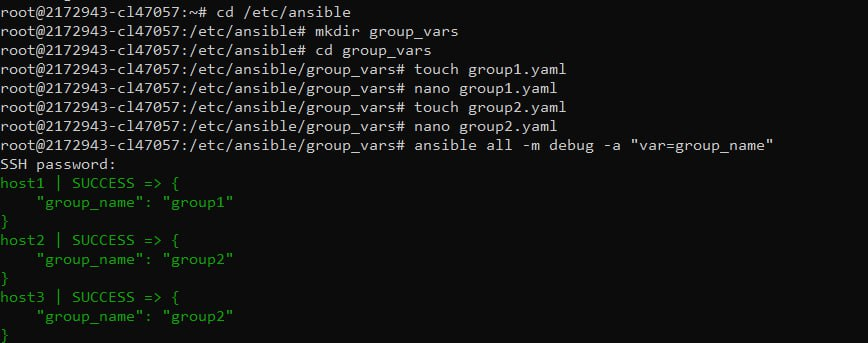


Рисунок 6 – Создание «yaml» файлов и новой переменной с помощью модуля «debug»

Запишем в «playbook.yml» следующий код (рисунок 7).

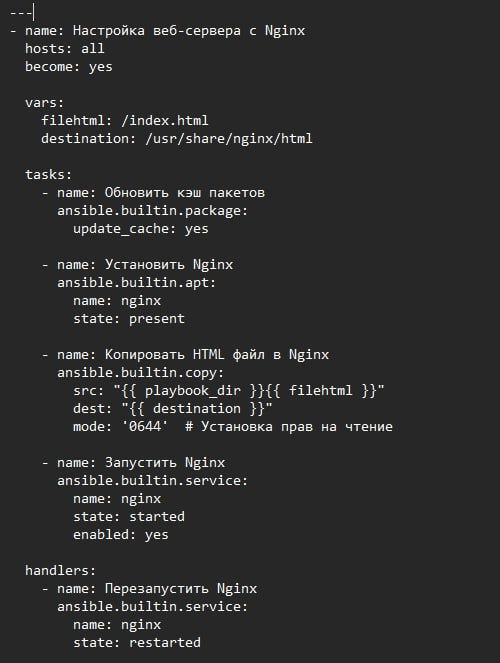


Рисунок 7 – Изменение в файле «playbook.yml»

Перед запуском «playbook» необходимо создать файл «index.html» в основной директории с простым кодом, который будет передаваться на веб-сервер:

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta http-equiv="X-UA-Compatible" content="IE=edge">  <meta name="viewport" content="width=device-width, initialscale=1.0">  <title>Document</title>  </head>  <body>  <br> Web Server installed on 2172953-cl47057 </br>  <br>Server's group: group1</br>  </body>  </html> |

После запуска «playbook.yml» на всех трех серверах будут запущены веб-серверы. При переходе в браузере по их адресам на странице отобразится информация, относящаяся к конкретному удаленному серверу.

Результат запуска «playbook.yml» представлен на рисунках 8, 9.



Рисунок 8 – Результат запуска «playbook.yml» в браузере

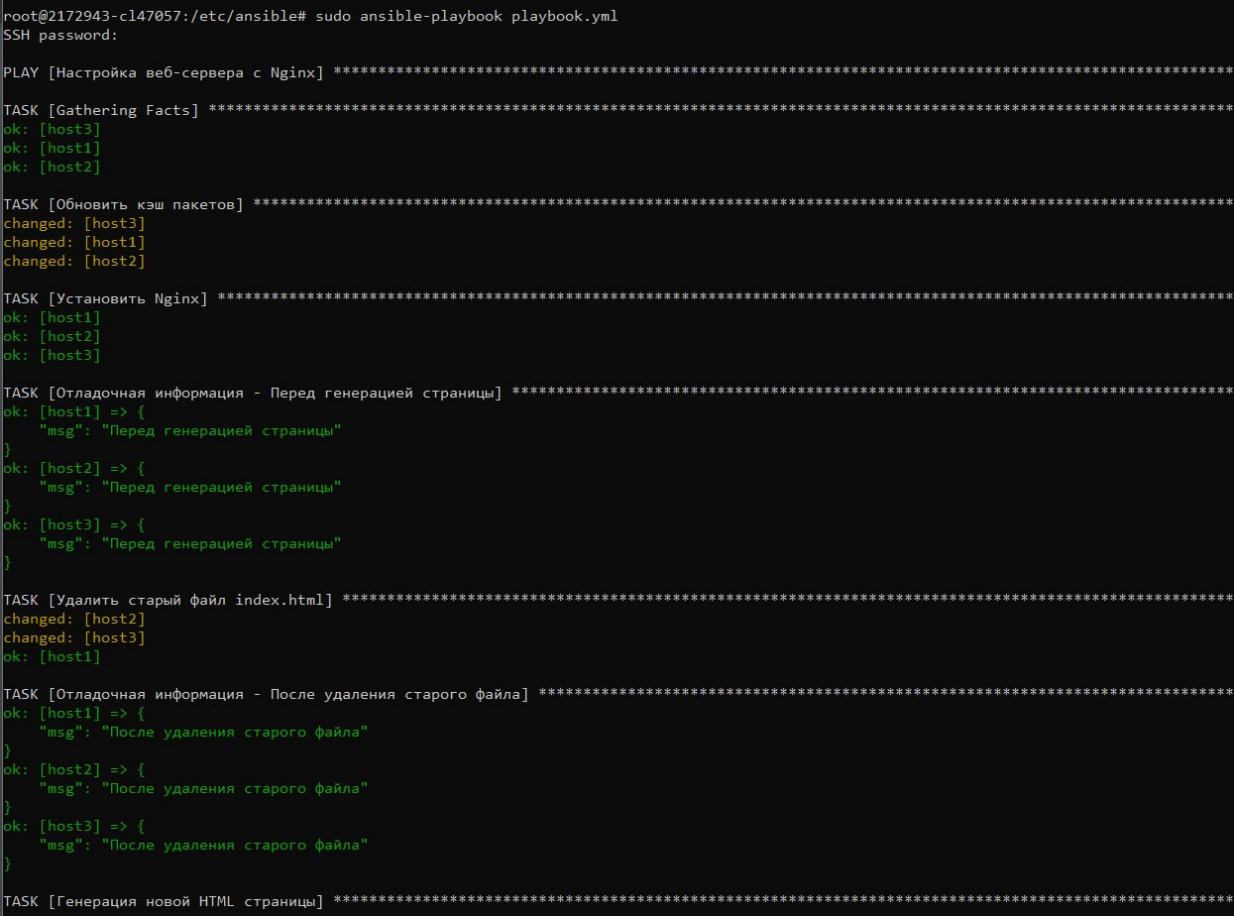


Рисунок 9 – Результат запуска «playbook.yml»

Разобьем «playbook» для запуска веб-сервера на две простые роли: «upgrade\_packages» для обновления всех пакетов и «setup\_web\_servers» для установки и запуска веб-сервера. Создадим папку «roles» в основной папке проекта и перейдем в неё. Для этого выполним следующие команды:

|  |
| --- |
| sudo mkdir roles  cd roles/  sudo ansible-galaxy init upgrade\_packages  sudo ansible-galaxy init setup\_web\_servers |

Результат выполнения команд представлен на рисунке 10.

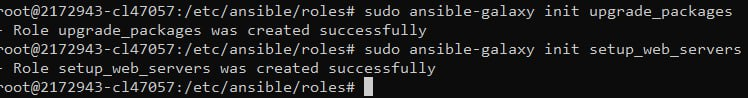


Рисунок 10 – Создание директорий для ролей

Перед запуском «playbook» необходимо внести в него следующие изменения:

|  |
| --- |
| ---  - name: Web server  hosts: all  become: yes  roles:  - upgrade\_packages  - setup\_web\_servers |

На рисунке 11 представлен результат запуска «playbook» с использованием ролей «upgrade\_packages» и «setup\_web\_servers».

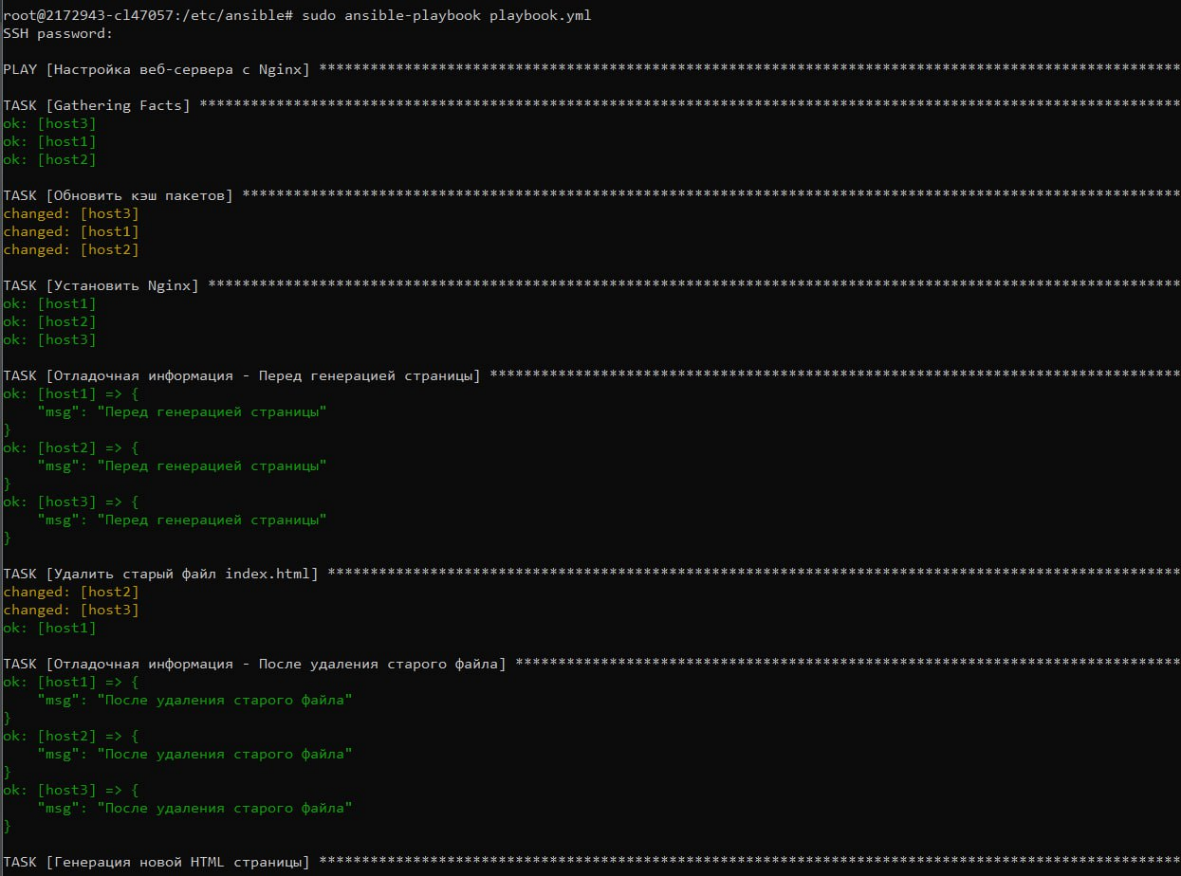


Рисунок 11 – Запуск «playbook» с ролями

Измененный «playbook» запустился без ошибок. Применение ролей в «Ansible» позволяет логически разделить задачи по настройке системы на отдельные блоки для повышения читаемости и сопровождаемости кода. Роли «upgrade\_packages» и «setup\_web\_servers» выполнили необходимые операции по установке и настройке веб-серверов на удаленных хостах.

В результате выполнения лабораторной работы были приобретены практические навыки по использованию «Ansible» для автоматизации настройки и управления конфигурациями серверов.