

## **ANSIBLE CONFIGURATION MANAGEMENT**

Configuration Management has evolved over the years and has been a revelation to solving critical problems in the tech industry that has given organisations the opportunity to have lots of servers in their thousands.

As DevOps Engineers, we are required to perform certain functions like installation, updates, upgrades ,patches ,security fixes and provisioning servers. These servers could be webservers, NFS server or database servers and when provisioned, these servers are required to be managed properly .There are various configuration tools in the market like Ansible, puppet, salt and chef but Ansible stands out of all of them because of the major functions and task carried out by the tool.

Advantages of Ansible as a configuration tool.

- 1)Easy and free to use and its a Python based which makes it the preferred choice to most DevOps Engineers
- 2)Ansible is Agentless and just by installing it on the control server, it can control other targeted servers to communicate with them through ssh and can target as much as 1000 servers through the control server
- 3)Ansible is pushed based and once pushed it can be automatically installed to all servers at once if they are all in the inventory.
- 4)Ansible uses YAML file which is easy to learn as well as It can be installed in Ubuntu and Linux repo and its readily available to be installed.

In this project we would be installing and configuring Ansible client to act as a Control server and creating a playbook to automate servers configuration

Pre-requisite for the projects is the following.

- 1) Fundamental Knowledge of Installing and downloading software
- 2) Basic Understanding of Linux Commands

- 3) AWS account login with EC2 instances
- 4) 3 Webserver Ubuntu
- 5) Database Server: On Ubuntu 22.04+ MySQL
- 6) Storage Server: Red Hat Enterprise Linux 9 +NFS Server
- 7) Load Balancer Server
- 8) Ansible Jenkins Server
- 9) Jenkins
- 10) Python Language
- 11) Code Repository
- 12) Internet connection
- 13) Visual Studio Code

### IMPLEMENTATION STEPS: Set up of all EC-2 instances.

- i) Ensure you login with your details to your AWS console via the <https://aws.amazon.com>
- ii) Click on the EC2 link and spin up 7 EC2 instances and make sure they are set up with the operating systems below.

1 NFS server (Red hat)

3 webservers. 1 Ansible-Jenkins Server

1 Load Balance Server 1 Database Server

You can see the instance state that shows all 7 servers are currently running.

We would be updating our Jenkins server and naming it Jenkins-Ansible-Server for clarity as we want to use this to run our playbooks.

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<input type="checkbox"/>	Jenkins-Ansible	i-0bce5b7199064a2e3	<span>Running</span> 	t2.micro	<span>2/2 checks passed</span> 	No alarms 	us-east-1b	ec2-52-2
<input type="checkbox"/>	Target server-...	i-0db50ef15367c4a5e	<span>Running</span> 	t2.micro	<span>2/2 checks passed</span> 	No alarms 	us-east-1a	ec2-44-2
<input type="checkbox"/>	Target server-...	i-0ecd47a52897b2b17	<span>Running</span> 	t2.micro	<span>2/2 checks passed</span> 	No alarms 	us-east-1a	ec2-54-
<input type="checkbox"/>	Target-server-...	i-05fac673cd05d6514	<span>Running</span> 	t2.micro	<span>2/2 checks passed</span> 	No alarms 	us-east-1a	ec2-54-8

We would navigate to github to create a new repository called ansible-config-mgt.

The screenshot shows a GitHub repository page for 'ansible-config-mgt'. At the top, it says 'Public'. Below that, there are buttons for 'main' (with a dropdown arrow), '1 branch' (with a dropdown arrow), and '0 tags'. To the right are buttons for 'Go to file', 'Add file', and 'Code' (with a dropdown arrow). A commit history section shows a single commit: 'Create README.md' by 'cb25b19' 4 hours ago, with 1 commit. Below the commit list is a message: 'Help people interested in this repository understand your project by adding a README.' followed by a green 'Add a README' button.

Then we proceed to install ansible and check its version as shown below.

```
ubuntu@ip-172-31-16-193:~$ sudo hostname jenkins-ansible-server
ubuntu@ip-172-31-16-193:~$ bash
ubuntu@jenkins-ansible-server:~$ ansible --version
ansible 2.9.6
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.8.10 (default, May 26 2023, 14:05:08) [GCC 9.4.0]
ubuntu@jenkins-ansible-server:~$ []
```

## JENKINS CONFIGURATION

Now we have to configure Jenkins job to save our repository every time we make changes. Please note that our Jenkins server was configured to use an elastic IP to be able to maintain a dedicated server for our process as you would see as we progress.

- 1) We access our Jenkins and create a new project “ansible new” as shown below .



## Welcome to Jenkins!

Username

Password

Keep me signed in

Sign in

Enter an item name

ansible new  
+ Required field

Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Fill in the description of your project and select git .

## Description

My second ansible creation for a project

[Plain text] [Preview](#)

- Discard old builds [?](#)
- GitHub project
- This project is parameterised [?](#)
- Throttle builds [?](#)
- Execute concurrent builds if necessary [?](#)

Advanced ▾

## Source Code Management

None

Git [?](#)

Copy and paste the git repo and ensure its on main branch.

<https://github.com/username/repo/ansible-config-mgt.git>

Credentials ?  
- none -

Add Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*main

Add Branch

</> Changes

My second ansible c

Workspace

▶ Build Now  


⚙ Configure

Delete Project

Rename

Permalinks

Click the build now manually to test it

Success > Console Output /

#1 24 Jun 2023, 00:32

Atom feed for all Atom feed for failures

↑ ↑ ↓

Proceed to configure webhook and set webhook to trigger ansible build as shown below.

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules Beta

Actions

Webhooks

Webhooks

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

⚠ http://3.87.244.167:8080/github-w... (push)

Edit Delete

Insert the Payload URL as well as the content type and proceed to add the webhook.

Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

Payload URL \*

Content type

Secret

Which events would you like to trigger this webhook?

Just the push event.

Send me everything.

Let me select individual events.

Active

We will deliver event details when this hook is triggered.

Add webhook

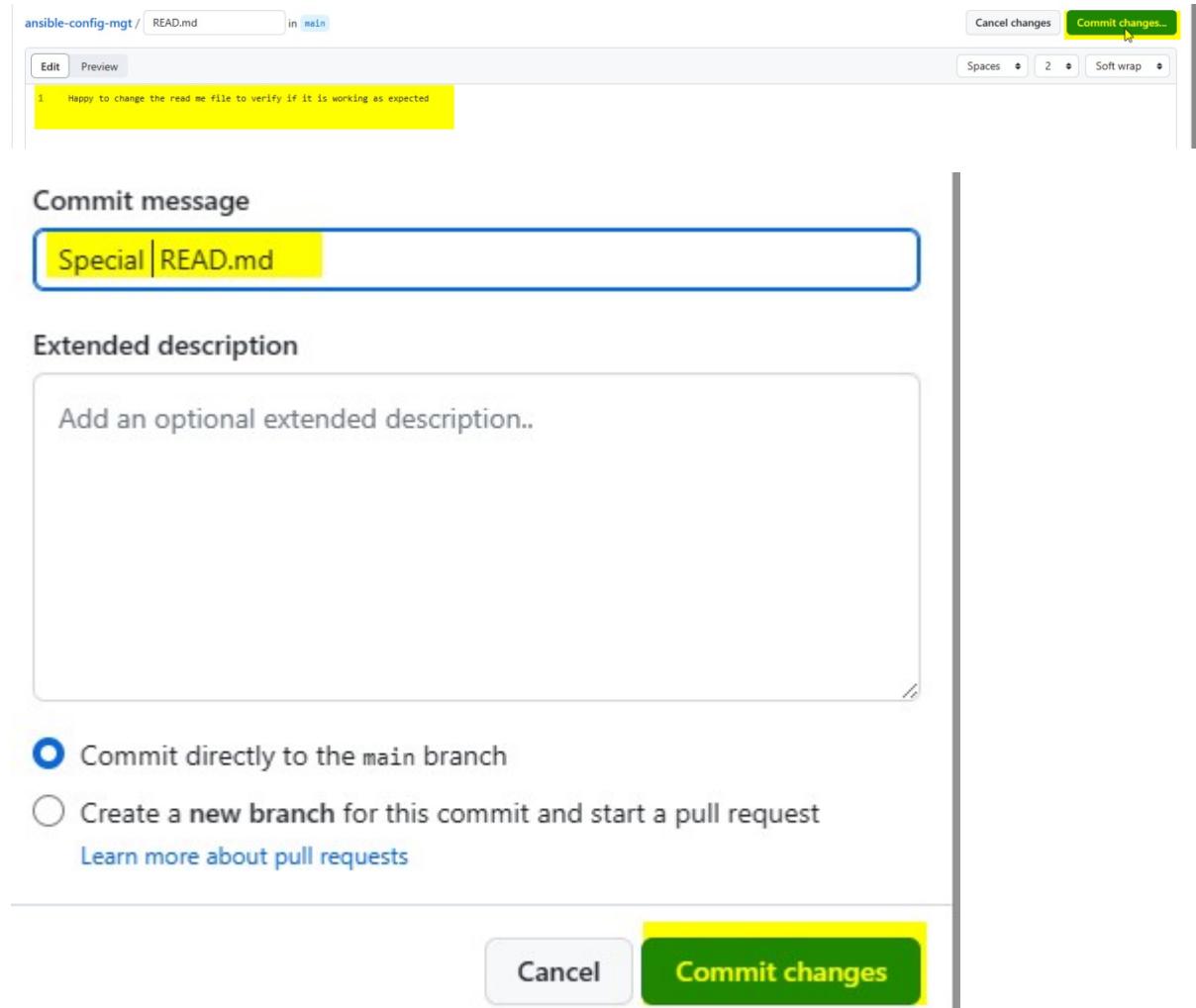
## Webhooks

[Add webhook](#)

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

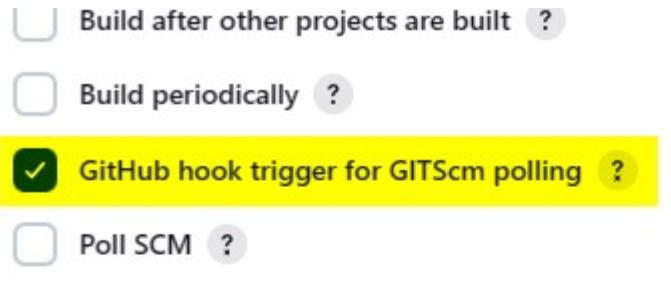
<a data-bbox="223 377 562 406" href="http://3.87.244.167:8080//github-w... (push)">http://3.87.244.167:8080//github-w... (push)</a>	<a href="#">Edit</a>	<a href="#">Delete</a>
<a data-bbox="223 440 562 469" href="http://18.206.234.83:8080//github-w... (push)">http://18.206.234.83:8080//github-w... (push)</a>	<a href="#">Edit</a>	<a href="#">Delete</a>

After its been successfully added, we proceed to commit new changes

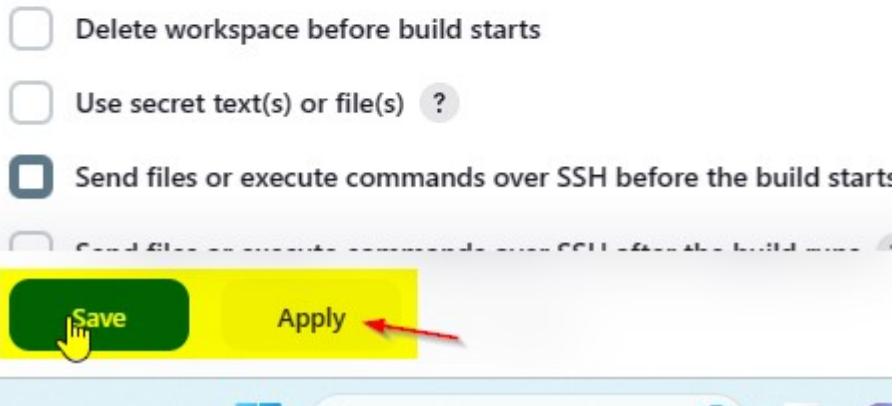


The screenshot shows the GitHub commit interface. At the top, there's a header with the repository name "ansible-config-mgt" and a file path "READ.md" under the "main" branch. Below the header, there are buttons for "Edit" and "Preview". The main area contains a commit message: "Happy to change the read me file to verify if it is working as expected". To the right of the message, there are buttons for "Cancel changes" and a prominent green "Commit changes" button. Below the commit message, there's a section titled "Commit message" with a text input field containing "Special | READ.md". Underneath, there's a section titled "Extended description" with a text area placeholder "Add an optional extended description..". At the bottom, there are two radio buttons for branching: "Commit directly to the main branch" (selected) and "Create a new branch for this commit and start a pull request". A link "Learn more about pull requests" is provided. Finally, at the very bottom, there are "Cancel" and "Commit changes" buttons.

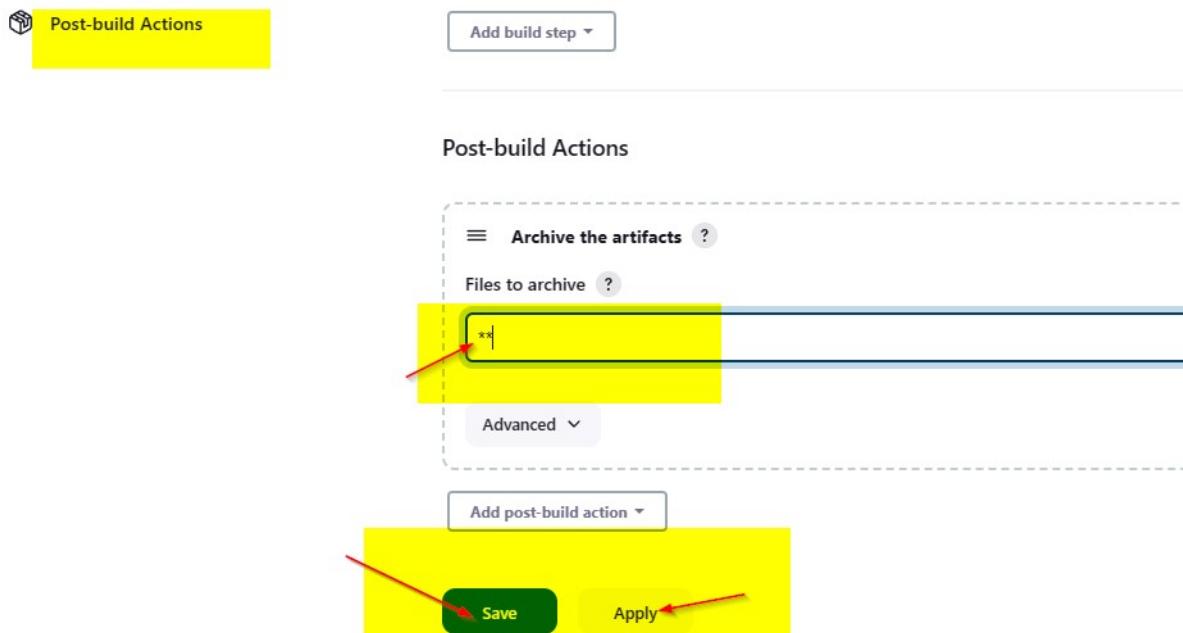
We navigate back to Jenkins to select the “GitHub hook trigger for GITScm polling. Then Apply and Save



## Build Environment



We then click on post build actions and archive the artifacts by adding a \*\* and clicking Apply and Save



Once all task is completed, we then proceed to commit more changes but this time it would automatically populate in the Jenkins server.

A screenshot of a code editor interface. At the top, there's a header with the project name "ansible-config-mgt / READ.md" and a branch indicator "in main". On the right side of the header are buttons for "Cancel changes" and "Commit changes...". Below the header is a toolbar with "Edit" and "Preview" buttons, and settings for "Spaces", "2", "Soft wrap". The main area shows a code editor with the following content:

```
1 Happy to change the read me file to verify if it is working as expected
2
3 thanks
4 We need ansible to run a command fr-or 1000 servers
5
6 lets do it |
```

## Commit message

1000 servers READ.md

### Extended description

Add an optional extended description..

- Commit directly to the `main` branch  
 Create a new branch for this commit and start a pull request  
[Learn more about pull requests](#)

Cancel

Commit changes

## </> Changes

My second ansible creation for a project

Workspace

Build Now

Configure

Delete Project

GitHub Hook Log

Rename



Build History

trend ▾

Filter builds...

/

✓ #6

27 Jun 2023, 02:56



Last Successful Artifacts

[READ.md](#) 182 B [view](#)

## Permalinks

- Last build (#6), 1 min 23 sec ago
- Last stable build (#6), 1 min 23 sec ago
- Last successful build (#6), 1 min 23 sec ago
- Last completed build (#6), 1 min 23 sec ago

Go to terminal and check any of the archived builds using the commands below

```
ubuntu@ip-172-31-16-193:~$ cd /var/lib/jenkins/jobs/ansible/builds/
ubuntu@ip-172-31-16-193:/var/lib/jenkins/jobs/ansible/builds$ cd 5/
ubuntu@ip-172-31-16-193:/var/lib/jenkins/jobs/ansible/builds/5$ ls
build.xml log
ubuntu@ip-172-31-16-193:/var/lib/jenkins/jobs/ansible/builds/5$ cat build.xml
<?xml version='1.1' encoding='UTF-8'?>
<build>
  <actions>
    <hudson.model.CauseAction>
```

Please note that anytime you try to stop and start your Jenkins-Ansible server. The ip changes and as I said earlier we switched to the elastic ip to ensure he ip is still available when instance is turned off .

## Prepare your environment using Visual Studio Code

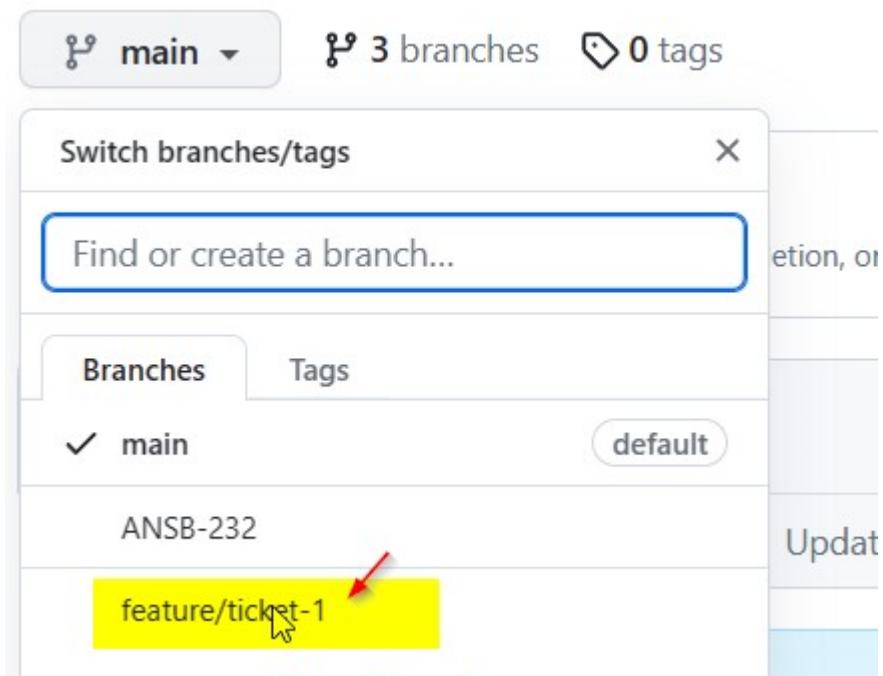
The next task would be to write some codes and we would need an IDE to perform such task hence visual studio code.

We would first clone down our ansible-config repository to our Jenkin instance. You check the content of the ansible -server to confirm it is there as shown below .

```
ubuntu@jenkins-ansible-server:~$ ls
ansible-config-mgt inventory
ubuntu@jenkins-ansible-server:~$ ls -ltr ansible-config-mgt/
total 12
-rw-rw-r-- 1 ubuntu ubuntu 145 Jun 25 18:11 READ.md
```

Then we navigate to the main branch to create a new branch called

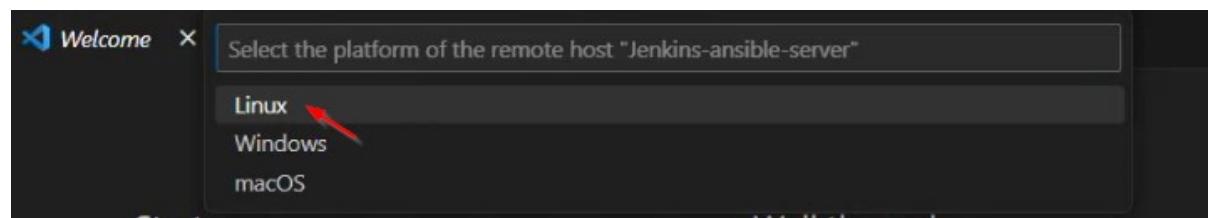
Feature/ticket-1

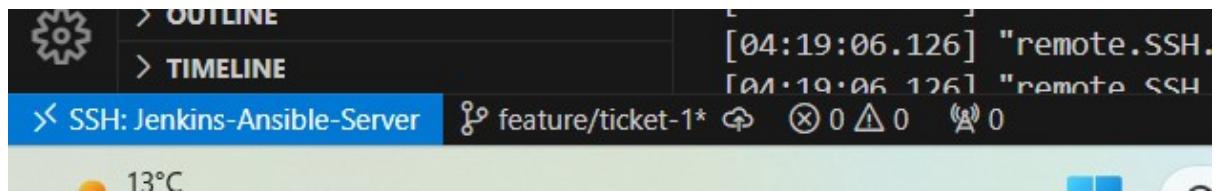
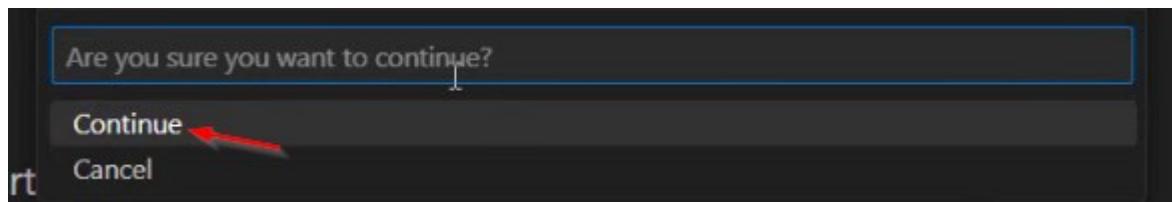


We are checking out the new branch to our local machine as shown below and we would start building codes and creating directory structure.

```
● ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ git branch -v
  a          87342c9 1000 servers READ.md
* feature/ticket-1 87342c9 1000 servers READ.md
  main       87342c9 1000 servers READ.md
  r          87342c9 1000 servers READ.md
○ ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ 
○ ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ 
○ ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ 
○ ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ 
○ ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ 
● ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ git branch
  a
* feature/ticket-1
  main
  r
```

Navigating through the VS code we would see the blue icon that shows that the SSH for Jenkins-Ansible-Server was successful connected in the VS Code

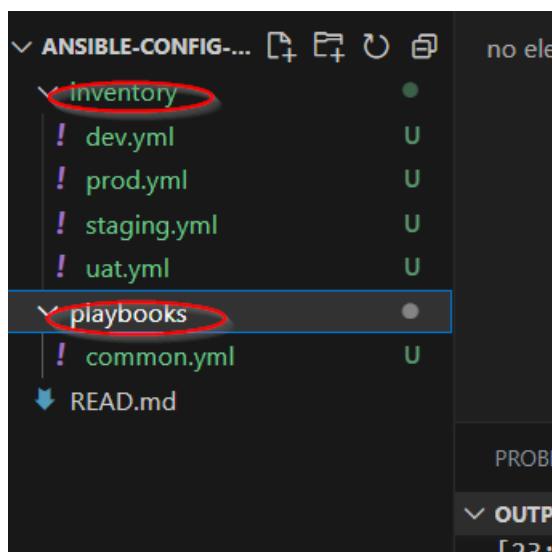




Then we create the folders and .yml files directly from the VS code rather than the normal convention of using linux command in the vs code terminal.

Inventory folder: contains dev.yml prod.yml staging.yml uat.yml files

Playbook folder: contains common.yml file



Once we have carried out the file and folder creation, we would need to set up an ansible inventory. This is where all the host private ip addresses are kept and executed all at once

#### TERMINAL

- `ubuntu@jenkins-ansible-server:~/ansible-config-mgt$ cd`
- `ubuntu@jenkins-ansible-server:~$ ls`  
`ansible-config-mgt inventory`
- `ubuntu@jenkins-ansible-server:~$`

The 3 webservers private ip are kept in this inventory as shown below.

```
● ubuntu@ip-172-31-16-193:~$ cat inventory  
172.31.83.227  
  
172.31.86.21  
  
172.31.86.172
```

With this set up we need to know that Ansible uses port :222 default and it needs to ssh target servers from the Jenkins-Ansible server using the SSH-AGENT concept.

Then we get into the Jenkins-Ansible server using the terminal. We try to perform the ansible operation to ping the target-server-A and target-server-B using the command below and it shows success

```
ubuntu@jenkins-ansible-server:~$ vi inventory
ubuntu@jenkins-ansible-server:~$ vi inventory
ubuntu@jenkins-ansible-server:~$ ansible -i inventory -m "ping" all
The authenticity of host '172.31.83.227 (172.31.83.227)' can't be established.
ECDSA key fingerprint is SHA256:iaG4bcIWD2F2DDxtnUe9jp0f+CjkpZhjuyRjb1yj0Fs.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
172.31.83.227 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

```
ubuntu@jenkins-ansible-server:~$ ansible -i inventory -m "ping" all
The authenticity of host '172.31.86.172 (172.31.86.172)' can't be established.
ECDSA key fingerprint is SHA256:P+yQm6qwfz3q3FEcbXPfLmB0s0m0cRpg4tse5rGh9P0.
The authenticity of host '172.31.86.21 (172.31.86.21)' can't be established.
ECDSA key fingerprint is SHA256:+/s8vz+3kk0FCTNDNN+0a3cP0fIDRCgl2scgi4UkM/8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? 172.31.83.227 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
^C [ERROR] User interrupted execution.
```

We can also use the ssh `ubuntu@ipaddress` to successfully connect as seen below.

```
ubuntu@jenkins-ansible-server:~$ ssh ubuntu@44.204.162.231
The authenticity of host '44.204.162.231 (44.204.162.231)' can't be established.
ECDSA key fingerprint is SHA256:iaG4bcIWD2F2DDxtnUe9jp0f+CjkpZhjuyRjb1yj0Fs.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '44.204.162.231' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Tue Jun 27 04:12:46 UTC 2023
```

```
ubuntu@jenkins-ansible-server:~$ ssh -A ubuntu@54.84.53.64
The authenticity of host '54.84.53.64 (54.84.53.64)' can't be established.
ECDSA key fingerprint is SHA256:P+yQm6qwfz3q3FEcbXPfLmB0s0m0cRpg4tse5rGh9P0.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.84.53.64' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)
```

Please note that we have also connect to the target 3 servers to ensure the public key file of the Jenkins- Ansible server is added in the authorized keys file `.ssh/` file as shown below

Firstly, we retrieve the public key from Jenkins- Ansible server as shown below.

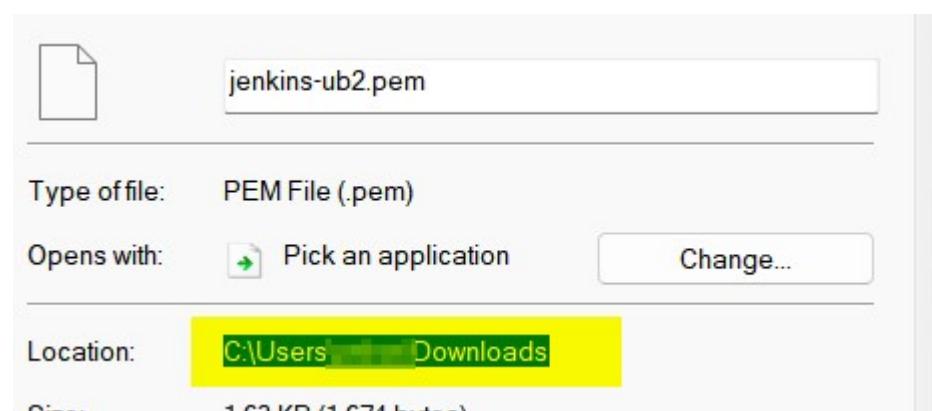
```
ubuntu@jenkins-ansible-server:~/ssh$ ls
authorized_keys  id_rsa  id_rsa.pub  known_hosts
ubuntu@jenkins-ansible-server:~/ssh$ cat id_rsa
id_rsa          id_rsa.pub
ubuntu@jenkins-ansible-server:~/ssh$ cat id_rsa.pub
ssh-rsa AAAQAB3T...= ubuntu@jenkins-ansible
ubuntu@jenkins-ansible-server:~/ssh$ ^C
```

Then add them into each server as seen below

```
drwx----- 2 ubuntu ubuntu 4096 Jun 23 18:29 .ssh/
-rw-r--r-- 1 ubuntu ubuntu     0 Jun 27 04:52 .sudo_as_admin_successful
ubuntu@Target-server-B:~$ cd .ssh
ubuntu@Target-server-B:~/ssh$ ls
authorized_keys
ubuntu@Target-server-B:~/ssh$ vi authorized_keys
ubuntu@Target-server-B:~/ssh$ 
drwx----- 2 ubuntu ubuntu 4096 Jun 25 18:46 .ssh/
-rw-r--r-- 1 ubuntu ubuntu     0 Jun 25 18:36 .sudo_as_admin_successful
-rw----- 1 ubuntu ubuntu 1126 Jun 25 18:46 viminfo
ubuntu@target-server-A:~$ cd .ssh
ubuntu@target-server-A:~/ssh$ ls
authorized_keys
ubuntu@target-server-A:~/ssh$ 
ubuntu@target-server-A:~/ssh$ ls
authorized_keys
drwx----- 2 ubuntu ubuntu 4096 Jun 23 18:29 .ssh/
-rw-r--r-- 1 ubuntu ubuntu     0 Jun 27 05:00 .sudo_as_admin_successful
ubuntu@target-server-c:~$ cd .ssh
ubuntu@target-server-c:~/ssh$ ls
authorized_keys
ubuntu@target-server-c:~/ssh$ vi authorized_keys
ubuntu@target-server-c:~/ssh$
```

Here we create the agent id, then confirm the path where the .pem file key is then perform ssh-add operations and can see the identity is added and then listed .

```
ubuntu@jenkins-ansible-server:~$ eval `ssh-agent -s` ←
Agent pid 3896
ubuntu@jenkins-ansible-server:~$ ssh-add /home/ubuntu/.ssh/id_rsa ←
Identity added: /home/ubuntu/.ssh/id_rsa (ubuntu@jenkins-ansible)
ubuntu@jenkins-ansible-server:~$ ssh-add -l ←
3072 SHA256:rR05vMsQ6qQTDAATSgh3qf/9fJLpCpr05wGd5b3vaE8 ubuntu@jenkins-ansible (RSA)
```



Here you would see that from the Jenkins-Ansible Server, they ssh into the target servers.

```
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status
```

```
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update
```

```
Last login: Tue Jun 27 04:59:46 2023 from 81.152.239.86  
ubuntu@target-server-c:~$ exit  
logout  
Connection to 54.84.53.64 closed.
```

```
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status
```

```
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update
```

```
Last login: Tue Jun 27 04:51:48 2023 from 81.152.239.86  
ubuntu@Target-server-B:~$ exit  
logout  
Connection to 54.159.153.188 closed.
```

```
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status
```

```
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update
```

```
Last login: Tue Jun 27 04:12:46 2023 from 18.204.66.90  
ubuntu@target-server-A:~$ ssh-add -l  
3072 SHA256:rR05vMsQ6qQTDAATSgh3qf/9fJLpCpr05wGd5b3vaE8 ubuntu@jenkins-ansible (RSA)  
ubuntu@target-server-A:~$ ^C  
ubuntu@target-server-A:~$ exit  
logout  
Connection to 44.204.162.231 closed.  
ubuntu@jenkins-ansible-server:~$ vi inventory
```

Next would be updating the Inventory/dev.yml file also noting that.

RHEL based servers are called ec-2 users and Load Balance users are called ubuntu.

```
! dev.yml M X ! common.yml M
inventory > ! dev.yml
1 [nfs]
2 172.31.85.81 ansible_ssh_user='ec2-user'
3
4 [webserver1]
5 172.31.83.227 ansible_ssh_user='ubuntu'
6
7 [webserver2]
8 172.31.86.21 ansible_ssh_user='ubuntu'
9
10 [webserver3]
11 172.31.86.172 ansible_ssh_user='ubuntu'
12 [
13 [database]
14 172.31.81.224 ansible_ssh_user='ubuntu'
15 [
16 [loadbalancer]
17 172.31.80.120 ansible_ssh_user='ubuntu'
18 ]
```

Once the dev.yml file is updated we would need to change directory to the playbooks folder and update the common.yml file as shown below. It can be seen that all servers should be executed from the playbooks

```

EXPLORER          ...
ANSIBLE-CONFIG-MGT [SSH: JENKIN...  ...
inventory
  ! dev.yml
  ! prod.yml
  ! staging.yml
  ! uat.yml
playbooks
  ! common.yml
READ.md

! dev.yml M | ! common.yml M X
playbooks > ! common.yml

1 |
2 |
3 |   - name: update nfs
4 |     hosts: all
5 |     remote_user: ec2-user
6 |     become: yes
7 |     become_user: root
8 |     tasks:
9 |       - name: ensure wireshark is at the latest version
10 |         yum:
11 |           name: wireshark
12 |           state: latest
13 |
14 |   - name: update LB, webserver1,webserver2, webserver3 and db servers
15 |     hosts: all
16 |     remote_user: ubuntu
17 |     become: yes
18 |     become_user: root
19 |     tasks:
20 |       - name: Update apt repo
21 |         apt:
22 |           update_cache: yes
23 |
24 |       - name: ensure wireshark is at the latest version

```

We are now suppose to run some shell script command by using the ansible command to ping the dev.yml file as shown below and success depicts all is working as expected

```

ubuntu@Jenkins-Ansible-Server:/ansible-config-mgt/inventory$ ansible all -m ping -i dev.yml
The authenticity of host '172.31.86.21 (172.31.86.21)' can't be established.
EDSA key fingerprint is SHA256:+s8vz+3k0FCTNDNN+0a3cP0fIDRGU2scg14UKM/8.
The authenticity of host '172.31.86.172 (172.31.86.172)' can't be established.
EDSA key fingerprint is SHA256:PyQm6qwfz3q3FccXpfLmB0sOm0cRpg4tse5rGh9P0.
Are you sure you want to continue connecting (yes/no/[fingerprint])? [DEPRECATION WARNING]: Distribution rhel 9.2 on host 172.31.85.81 should use /usr/libexec/pla
-with prior Ansible releases. A future Ansible release will default to using the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature will be removed in version 2.12. Deprecati
-warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
172.31.85.81 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
172.31.83.227 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
172.31.81.224 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
172.31.80.120 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}


```

Using the ansible -playbook command we get to install the wireshark from the Jenkins -ansible server to all other target

servers and it can be seen that each individual servers received those changes .

```
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ ansible-playbook -i inventory/dev.yml playbooks/common.yml
PLAY [update nfs] ****
[ASK [Gathering Facts]] ****
The authenticity of host '172.31.86.21 (172.31.86.21)' can't be established.
EDSA key fingerprint is SHA256:+/8vz+3kk0FCTNDNN+0a3cP0fIDRCg1zscg14UkM/8.
The authenticity of host '172.31.86.172 (172.31.86.172)' can't be established.
EDSA key fingerprint is SHA256:P+YQm6gwTzq3FEcbAPtLmb9stUmecKpg4tse5rhP0.
Are you sure you want to continue connecting (yes/no)? ok: [172.31.83.227]
[DEPRECATION WARNING]: Distribution rhel 9.0 on host 172.31.85.81 should use /usr/libexec/platform-python, but is using /usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using the discovered platform python for this host. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
ok: [172.31.85.81]
ok: [172.31.81.224]
ok: [172.31.80.120]
```

```
TASK [Update apt repo] ****
[WARNING]: Updating cache and auto-installing missing dependency: python3-apt
fatal: [172.31.85.81]: FAILED! => {"changed": false, "cmd": "apt-get update", "msg": "[Errno 2] No such file or directory: b'apt-get'", "rc": 2}
changed: [172.31.83.227]
changed: [172.31.81.224]
changed: [172.31.80.120]

TASK [ensure wireshark is at the latest version] ****
ok: [172.31.83.227]
ok: [172.31.80.120]
ok: [172.31.81.224]

PLAY RECAP ****
172.31.80.120          : ok=5    changed=2    unreachable=0    failed=0    skipped=0
d=0    rescued=0    ignored=0
172.31.81.224          : ok=5    changed=2    unreachable=0    failed=0    skipped=0
d=0    rescued=0    ignored=0
172.31.83.227          : ok=5    changed=2    unreachable=0    failed=0    skipped=0
d=0    rescued=0    ignored=0
172.31.85.81           : ok=3    changed=1    unreachable=0    failed=1    skipped=0
d=0    rescued=0    ignored=0
172.31.86.172          : ok=0    changed=0    unreachable=1    failed=0    skipped=0
```

Now lets check each server to be sure wireshark is installed on them as shown below

```
ubuntu@nginx-lb:~$ which wireshark
ubuntu@nginx-lb:~$ which wireshark
/usr/bin/wireshark
ubuntu@nginx-lb:~$
```

```
ubuntu@Database-Server:~$ which wireshark
ubuntu@Database-Server:~$ which wireshark
/usr/bin/wireshark
ubuntu@Database-Server:~$
```

```
ubuntu@target-server-A:~$ which wireshark
ubuntu@target-server-A:~$ which wireshark
/usr/bin/wireshark
ubuntu@target-server-A:~$ █
[ec2-user@nfs-server ~]$ which wireshark
/usr/bin/wireshark
[ec2-user@nfs-server ~]$ █
```

Now that we confirmed this ,we would be pushing our commit into git hub and creating a pull request .Once that happens we should expect our Jenkins server to be triggered automatically and builds and artifacts generated .

We use the git branch to know the branch that we are currently using as shown below and check the status to know the untracked files we have .They are shown below

```
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git branch
  a
* feature/ticket-1
  main   I
  r
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git status
On branch feature/ticket-1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    inventory/
    playbooks/
nothing added to commit but untracked files present (use "git add" to track)
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ █
```

We proceed to add inventory and check to status to confirm the new files has been added .Please note we need to add playbooks as it still shows untracked files

```
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git add inventory/
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git status
On branch feature/ticket-1
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  inventory/dev.yml
    new file:  inventory/prod.yml
    new file:  inventory/staging.yml
    new file:  inventory/uat.yml

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    playbooks/

```

Then we add playbooks

```
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git add playbooks/
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git status
On branch feature/ticket-1
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  inventory/dev.yml
    new file:  inventory/prod.yml
    new file:  inventory/staging.yml
    new file:  inventory/uat.yml
    new file:  playbooks/common.yml

ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ 
```

We run the git commit with a message tag and as it shows we need to set out git account to be accepted on the terminal and this would require the use of classic token from git as password would not be allowed

```
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git commit -m "added new inventory and playbooks yaml files"

*** Please tell me who you are.

Run

git config --global user.email "you@example.com"
git config --global user.name "Your Name"
  I
to set your account's default identity.
Omit --global to set the identity only in this repository.

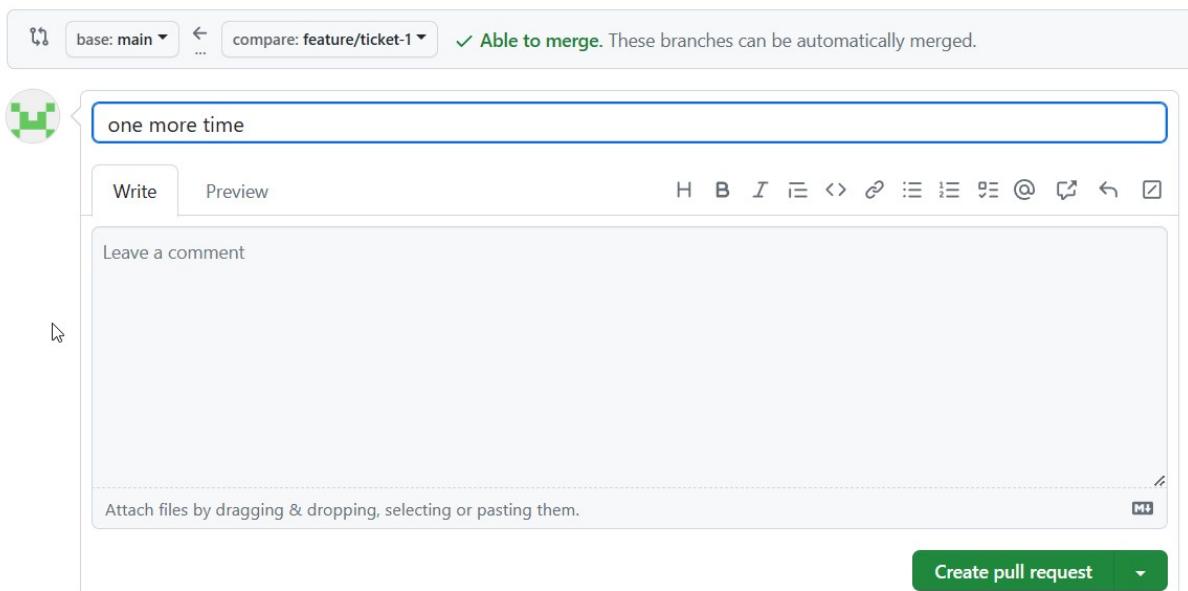
fatal: unable to auto-detect email address (got 'ubuntu@Jenkins-Ansible-Server.(none)')
```

```
'')
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git config --global user.name "eyewande2022"
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git config --global user.email "eyewande2022@gmail.com"
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git commit -m "one more time"
[feature/ticket-1 bceddb1] one more time
 5 files changed, 42 insertions(+)
   create mode 100644 inventory/dev.yml
   create mode 100644 inventory/prod.yml
   create mode 100644 inventory/staging.yml
   create mode 100644 inventory/uat.yml
   create mode 100644 playbooks/common.yml
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ 
```

Set the Url origin and check the remote

```
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git remote set-url origin https://github.com/eyewande2022/ansible-config-mgt
ubuntu@Jenkins-Ansible-Server:~/ansible-config-mgt$ git remote -v
origin https://github.com/eyewande2022/ansible-config-mgt (fetch)
origin https://github.com/eyewande2022/ansible-config-mgt (push)
```

We navigate back to git hub and create a pull request and merge the pull request to main .



**Open** eyewande2022 wants to merge 1 commit into `main` from `feature/ticket-1`

Conversation 0 Commits 1 Checks 0 Files changed 5

eyewande2022 commented now

No description provided.

one more time

bceddb1

Add more commits by pushing to the `feature/ticket-1` branch on [eyewande2022/ansible-config-mgt](#).

**Require approval from specific reviewers before merging**  
Branch protection rules ensure specific people approve pull requests before they're merged. [Add rule](#)

**Continuous integration has not been set up**  
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

**This branch has no conflicts with the base branch**  
Merging can be performed automatically.

**Merge pull request** You can also open this in GitHub Desktop or view command line instructions.

eyewande2022 merged commit `0c2e87a` into `main` now [Revert](#)

**Pull request successfully merged and closed**  
You're all set—the `feature/ticket-1` branch can be safely deleted. [Delete branch](#)

eyewande2022 Merge pull request #1 from eyewande2022/feature/ticket-1 ... 0c2e87a 1 minute ago 8 commits

inventory	one more time	20 minutes ago
playbooks	one more time	20 minutes ago
READ.md	Update READ.md	last week

As the pull request is successfully merged ,please note the Jenkins build would be automatically triggered .We proceed to view out Jenkins .During the course of the project the 2 jobs were created and had the same webhook connected ans they both got triggered .

S	Build	Time Since	Status
✓	ansible #12	4 min 19 sec	stable
✓	ansible new #7	4 min 20 sec	stable
✓	ansible new #6	6 days 19 hr	stable

Status    **Build #12 (3 Jul 2023, 22:13:54)**

</> Changes

Console Output

Success > Console Output /

#12 3 Jul 2023, 22:13  
#11 27 Jun 2023, 02:56  
#10 24 Jun 2023, 00:49

We can now proceed to check our server to see exactly where our build is located as shown below

```
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins$ cd jobs/
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins/jobs$ cd ansible
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins/jobs/ansible$ cd builds/
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins/jobs/ansible/builds$ ls
1 10 11 12 2 3 4 5 6 7 8 9 legacyIds permalinks
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins/jobs/ansible/builds$ cd 12/
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins/jobs/ansible/builds/12$ ls
build.xml changelog.xml log polling.log
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins/jobs/ansible/builds/12$ cat build.xml
```

```
ubuntu@Jenkins-Ansible-Server:/var/lib/jenkins/jobs/ansible/builds/12$ cat build.xml
<?xml version='1.1' encoding='UTF-8'?>
<build>
  <actions>
    <hudson.model.CauseAction>
      <causeBag class="linked-hash-map">
        <entry>
          <com.cloudbees.jenkins.GitHubPushCause plugin="github@1.37.1">
            <pushedBy>eyewande2022</pushedBy>
          </com.cloudbees.jenkins.GitHubPushCause>
        </entry>
      </causeBag>
    </actions>
  </build>
```

We just successfully automated our routine tasks by implementing Ansible