ANSIBLE

What is Ansible ?

- Ansible is an open-source automation tool, or platform, used for IT tasks such as configuration management, application deployment, intraservice orchestration, and provisioning.
- Configuration management tool.
- Can use this tool whether your servers are in on-premises or in the cloud.
- It turns your code into infrastructure i.e your computing environment has some of the same attributes as your application:

What is Configuration Management:

configuration: Each and every minure details of a system.

- It is a method through which we automate admin tasks.
- Configuration management tools turns your code into infrastructure.Your code would be Testable, Repeatable, Versionable.

Configuration Management have been performed in two ways:

1. Manual
2. Automation

Note: The Devops engineers managing the configuration in a automated manner using Devops tools.

IT infrastructure refers to composite of:

- Software
- Network
- People
- Process

Pain points: Earlier what kind of problems have before automation tools came ?

- Managing users& groups accounts.
- Dealing with packages
- Taking backup
- Deploy all kind of applications
- Configuring services.

Why configuration Management:

- Complete automation
- Increase up time
- Improve perfomance
- Prevent errors
- Reduce costcess

Benefits of Ansible:

- **Free :** Ansible is an open-source tool.
- **Powerful:** Ansible lets you model even highly complex IT workflows.
- **Flexible:** You can orchestrate the entire application environment no matter where it's deployed. You can also customize it based on your needs.
- **Agentless:** You don't need to install any other software or firewall ports on the client systems you want to automate. You also don't have to set up a separate management structure.
- Push mechanism

Other tools in the market can be really complicated :

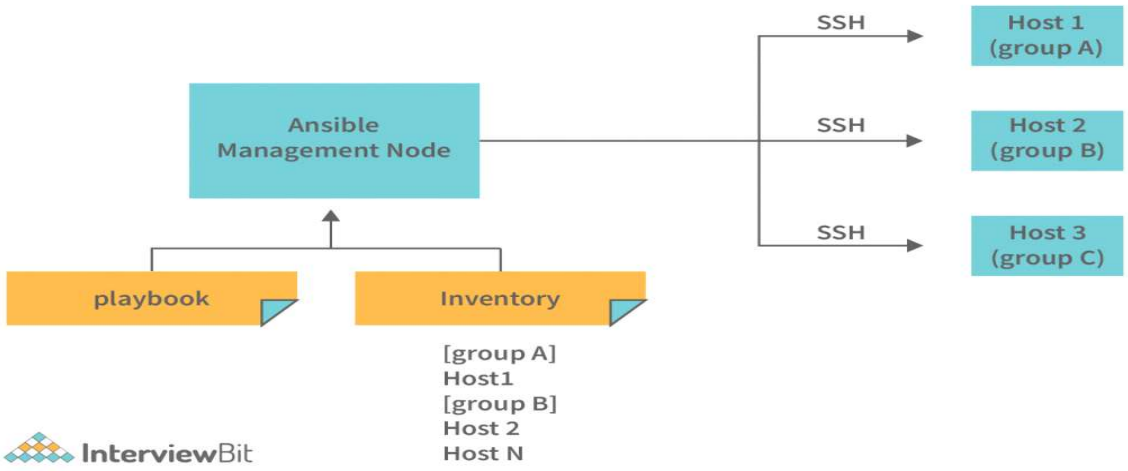
- huge overhead of Infrastructure setup
- complicated setup

- Pull mechanism
- Lot of learning required.

Ansible Features:

- Configuration Management
- Application Deployment
- Orchestration
- Security and Compliance
- Cloud Provisioning

Ansible Architecture:



- **Modules :**

1. Modules are like small programs that Ansible pushes out from a control machine to all the nodes or remote hosts.
2. The modules are executed using playbooks (see below), and they control things such as services, packages, and files.
3. Ansible executes all the modules for installing updates or whatever the required task is, and then removes them when finished. Ansible provides more than 450 modules for everyday tasks.

- **Plugins :**

1. As you probably already know from many other tools and platforms, plugins are extra pieces of code that augment functionality.
2. Ansible comes with a number of its plugins, but you can write your own as well. Action, cache, and callback plugins are three examples.

- **Inventories:**

1. All the machines you're using with Ansible (the control machine plus nodes) are listed in a single simple file, along with their IP addresses, databases, servers, and so on.
2. Once you register the inventory, you can assign variables to any of the hosts using a simple text file.

- **Playbooks :**

1. Ansible playbooks are like instruction manuals for tasks.
2. They are simple files written in YAML, which stands for YAML Ain't Markup Language, a human-readable data serialization language.
3. Playbooks are really at the heart of what makes Ansible so popular is because they describe the tasks to be done quickly and without the need for the user to know or remember any particular syntax.
4. Not only can they declare configurations, but they can orchestrate the steps of any manually ordered task, and can execute tasks at the same time or at different times.

Playbooks:

- Each playbook is composed of one or more 'modules' in a list
- Playbooks are divided into many sections like....

1. **Target Section** - Defines the hosts against which playbooks tasks has to be executed
2. **Variable Section** - Defines variables
3. **Tasks Section** - List of all modules that we need to run, in an order

- **APIs :**

1. Various APIs (application programming interfaces) are available so you can extend Ansible's connection types (meaning more than just SSH for transport), callbacks, and

more.

YAML(Yet Ain't Markup Language) Basics :

1. For Ansible, nearly every YAML file starts with a list.
2. Each item in the list is a list of key/value pairs, commonly called a "dictionary"

- All YAML files have to begin with "---" and end with "...
- All members of a list lines must begin with same indentation level

starting with "- "

--- # A list of tasty fruits

fruits:

- Apple

- Orange

- Strawberry

- Mango

...

- A dictionary is represented in a simple key: value form (the colon must be followed by a space)

--- # An employee record

Employee:

name: Hari

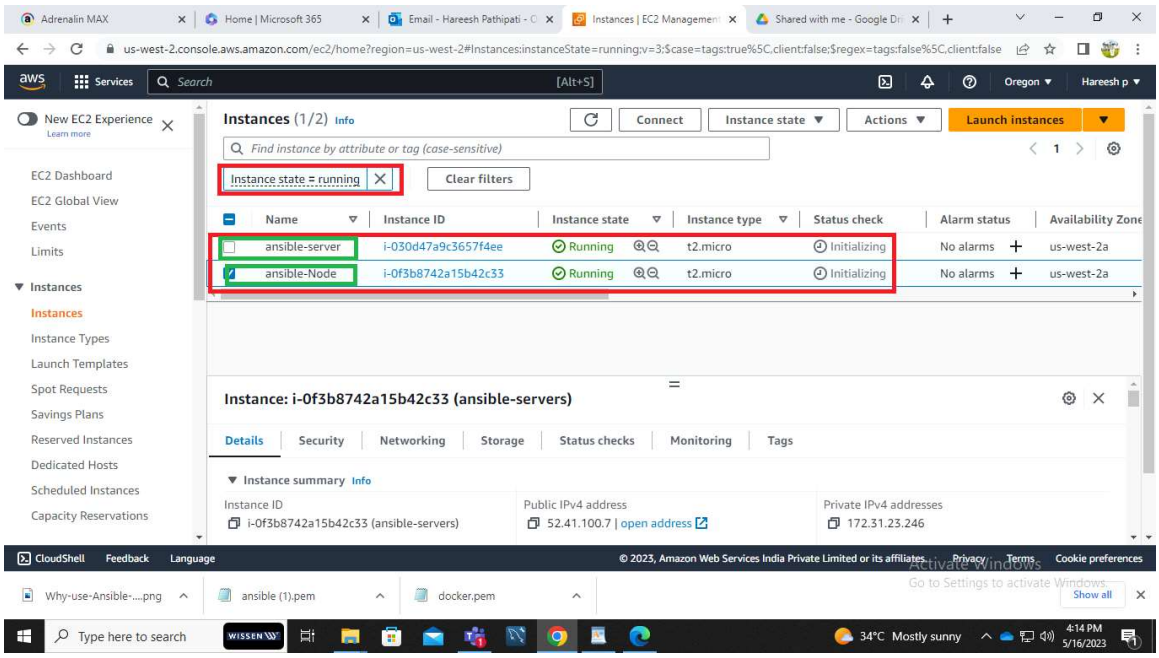
job: AWS Engineer

skill: cloud computing

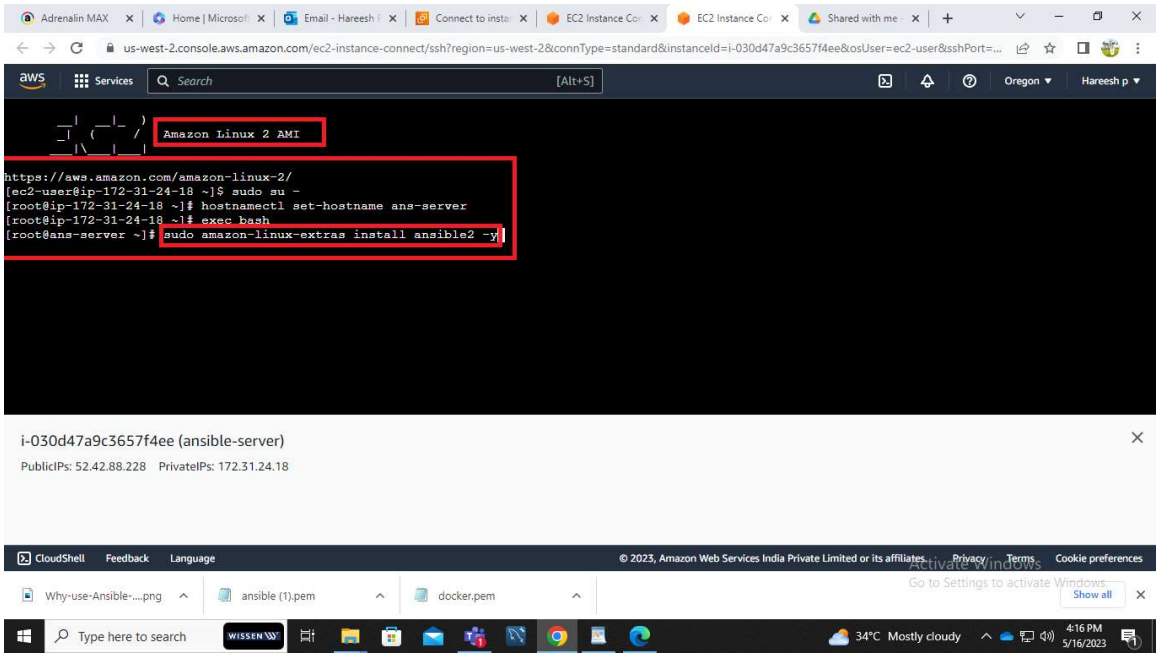
...

practicals:

step-1: Login to aws management console and launch the 2 servers

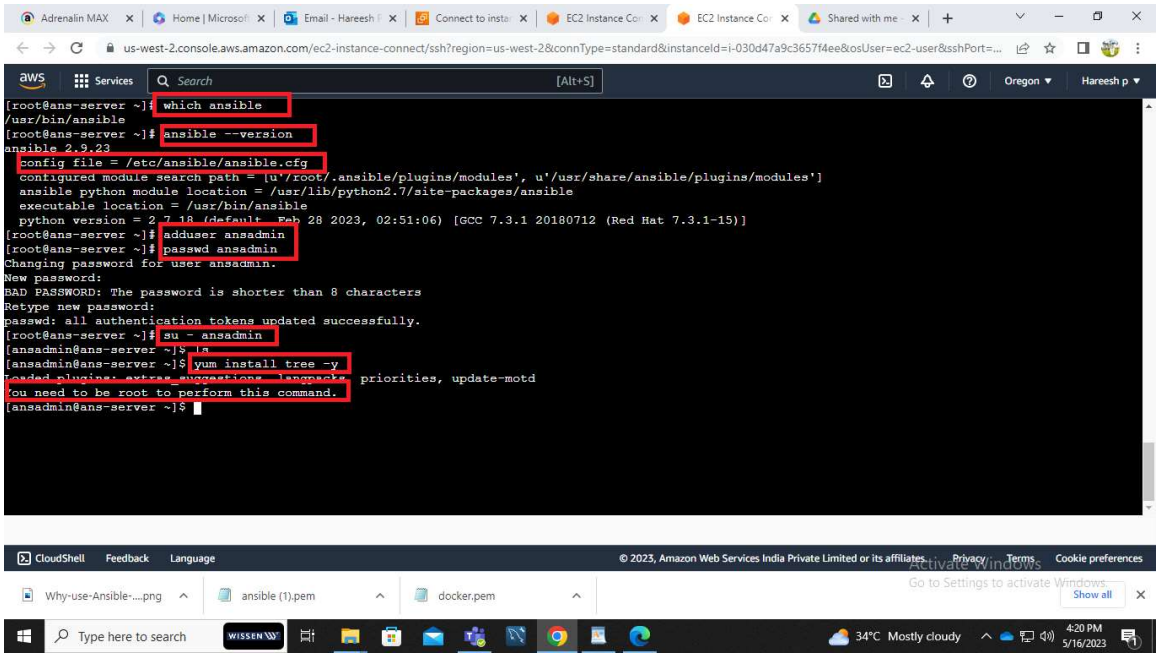


- Install Ansible in main ansible-server

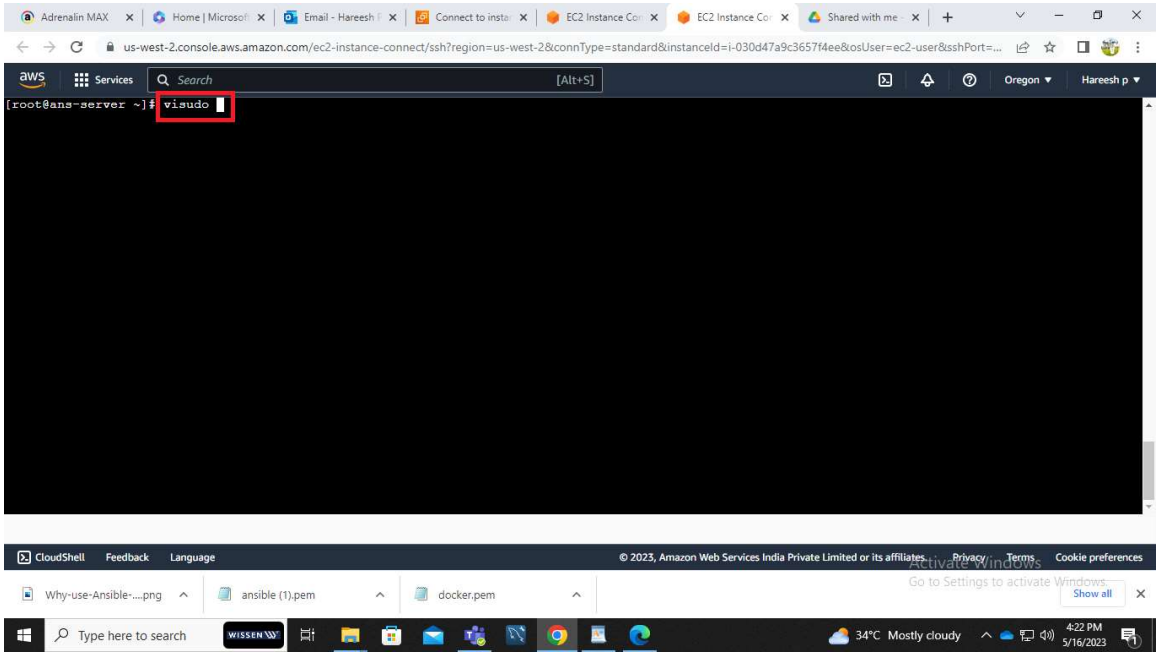


- check the vesrion of ansible and ansible location
- add the user and set password to user in both server and node machines.

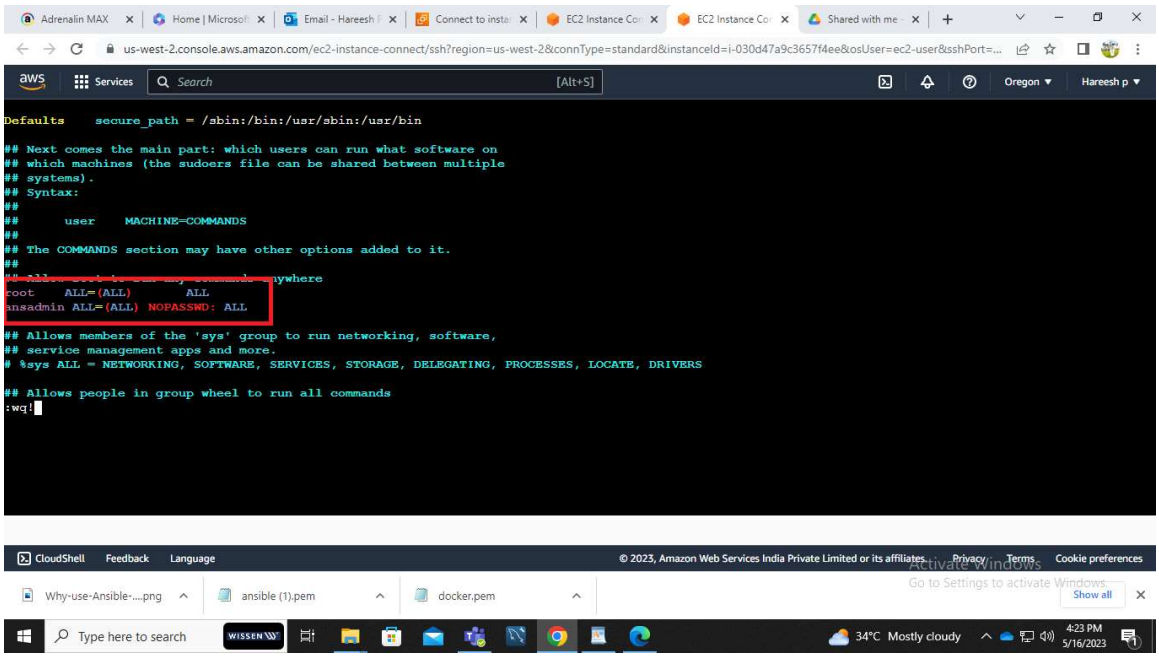
- when we install a package being an normal user it will thrown error



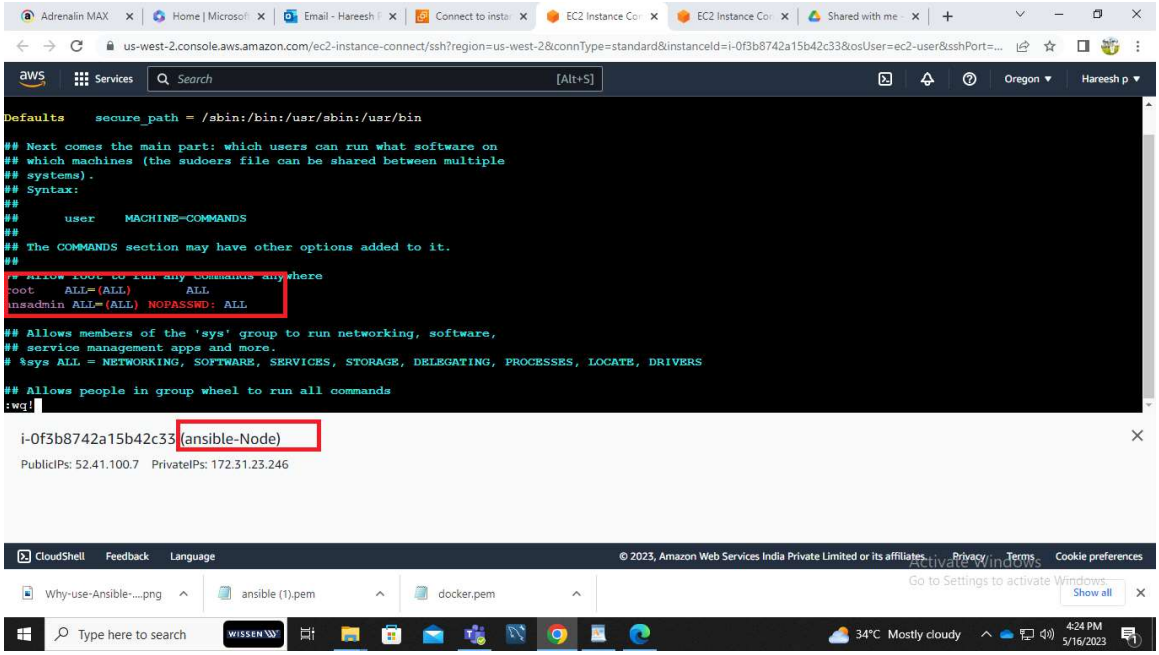
- For that error we need to change sudo file so we will get sudo priviliges



- In this visudo file we add an user and some priviliges

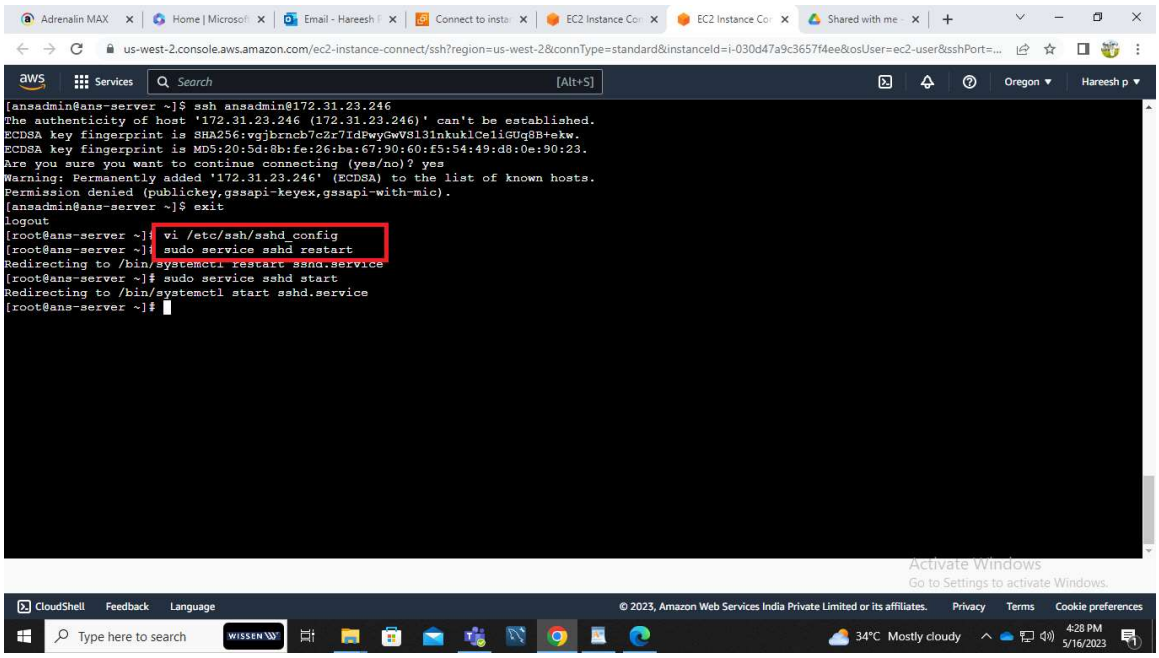


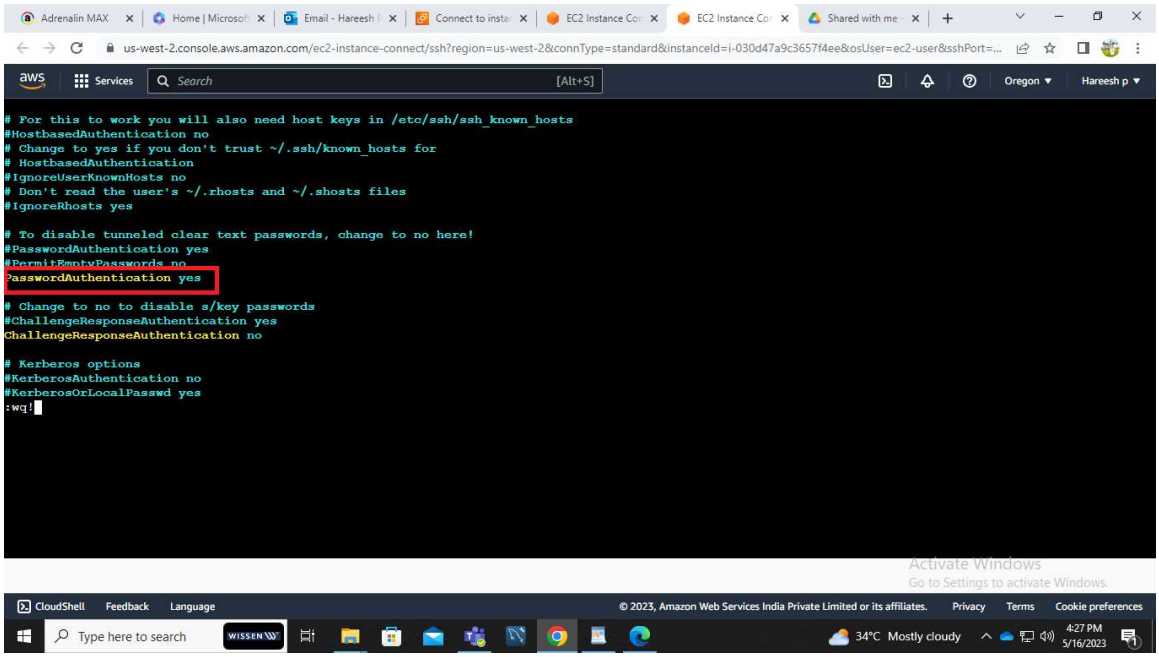
- In node machine also we have to add user



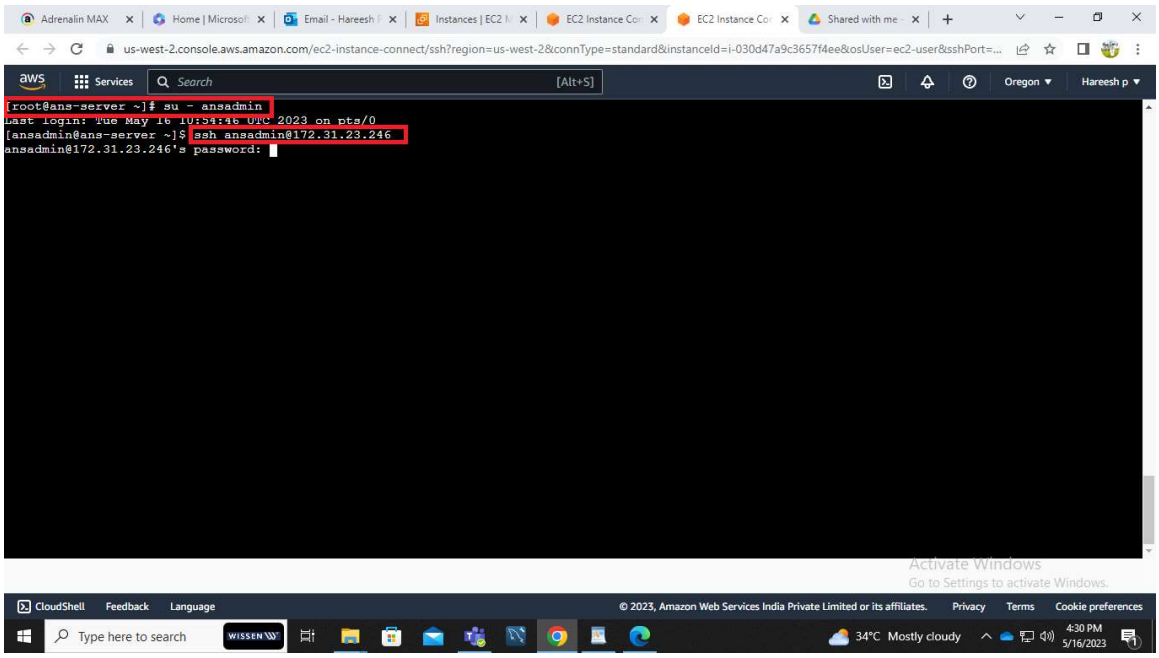
1. **with password authentication ssh establishment from Ansible server to node**

- For password authentication we need to change in con figuration file in both server and node machines.

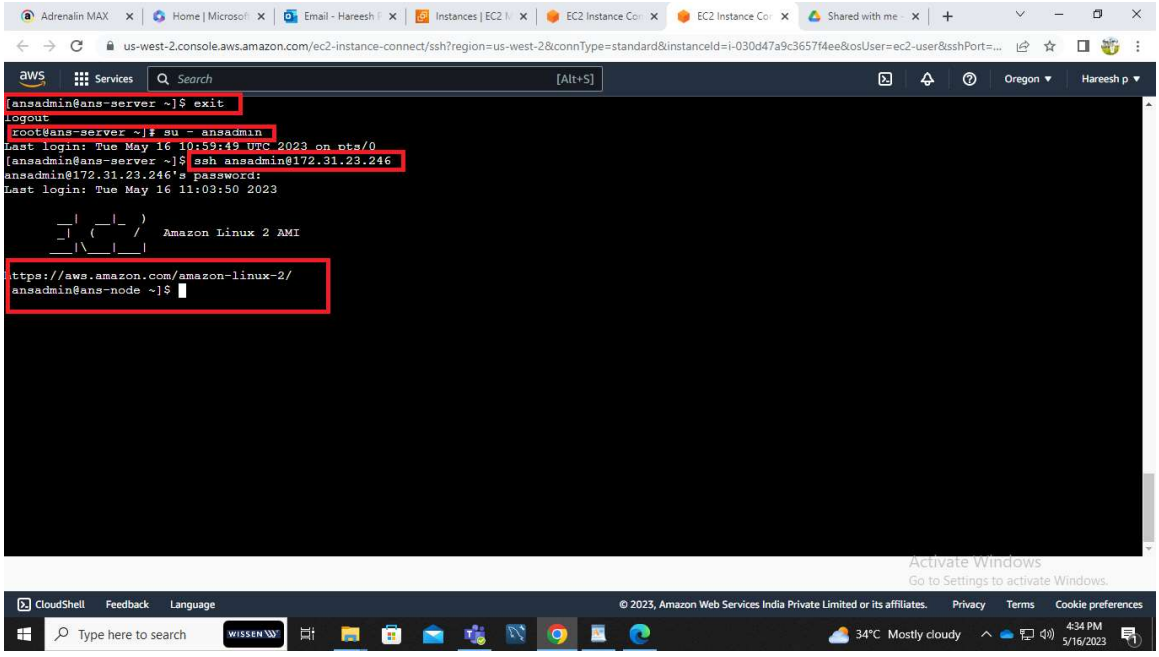




- For the last time it will ask permission login as a server to node machine.

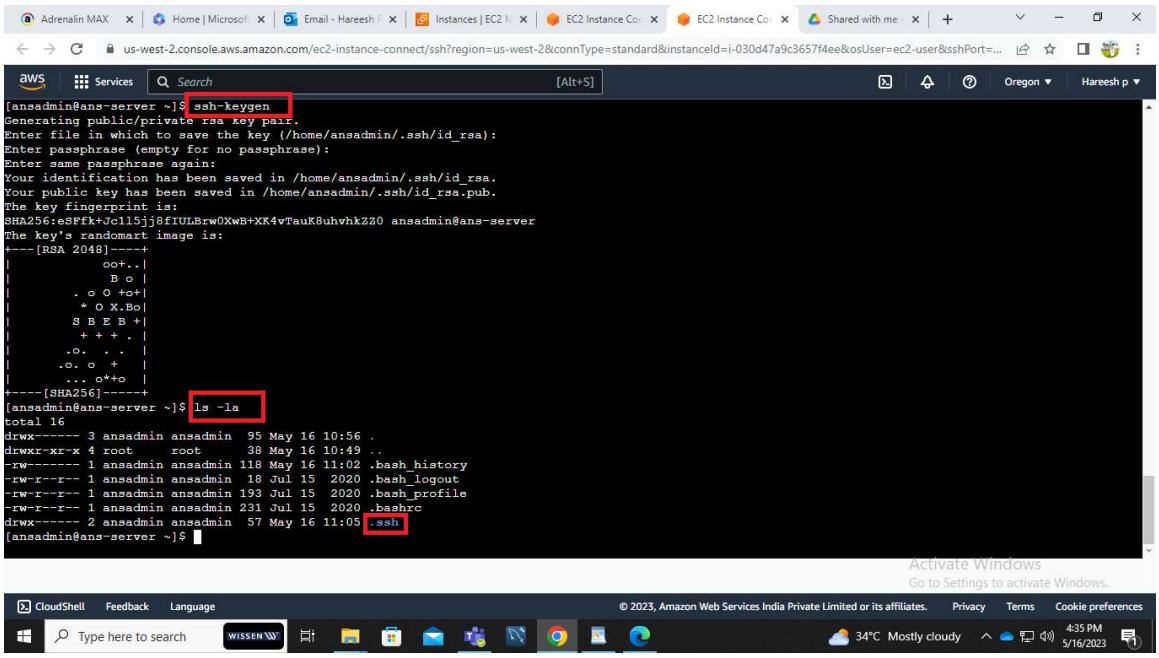


- copy the ssh and username @ private IP of node machine execute the command

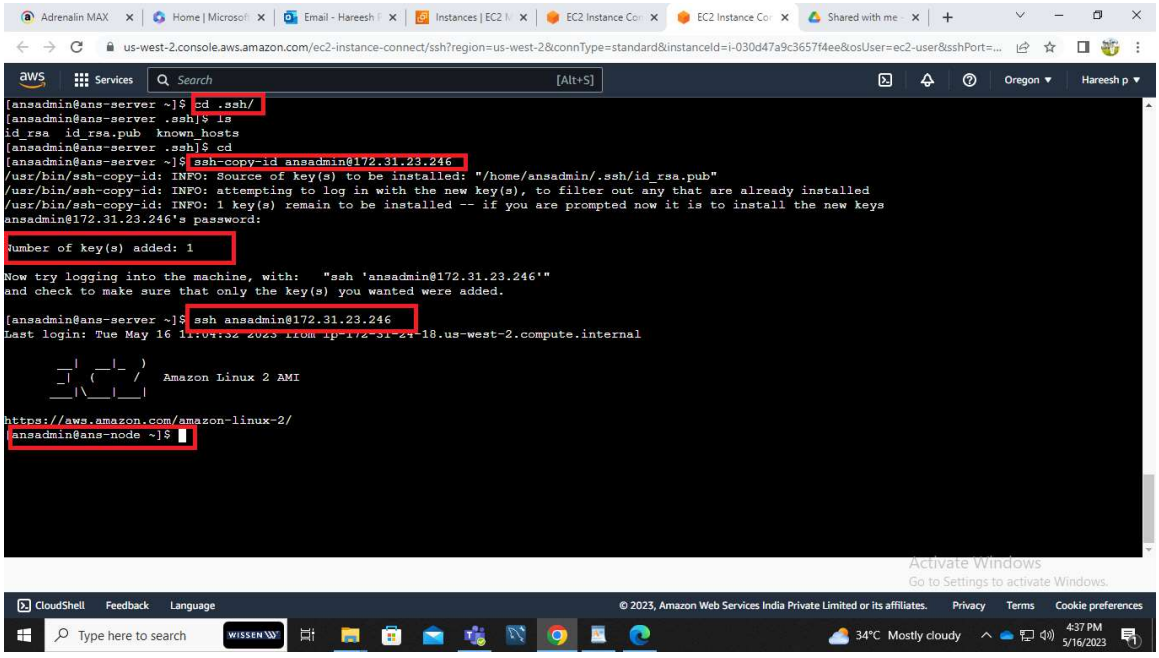


2. Password less ssh connection establishment from server to node :

- Creat ssh keys



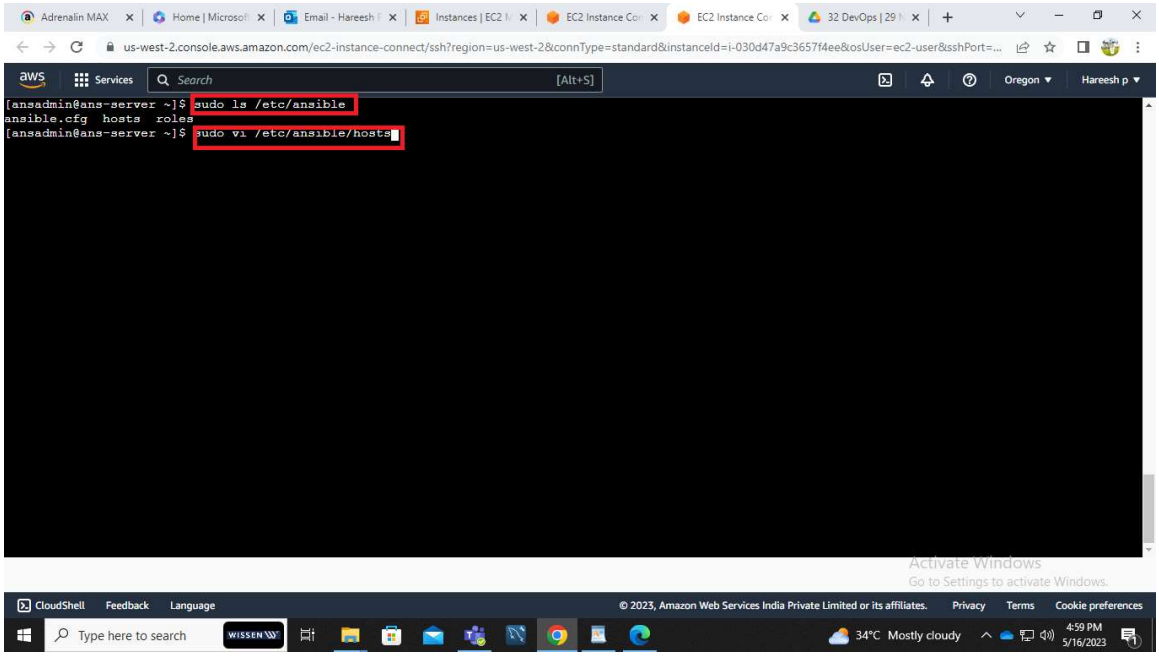
- copy the public key in node machine
- here we copy the username and private IP of node
- Being present in Main server we can control the node machine tasks using password less ssh connection



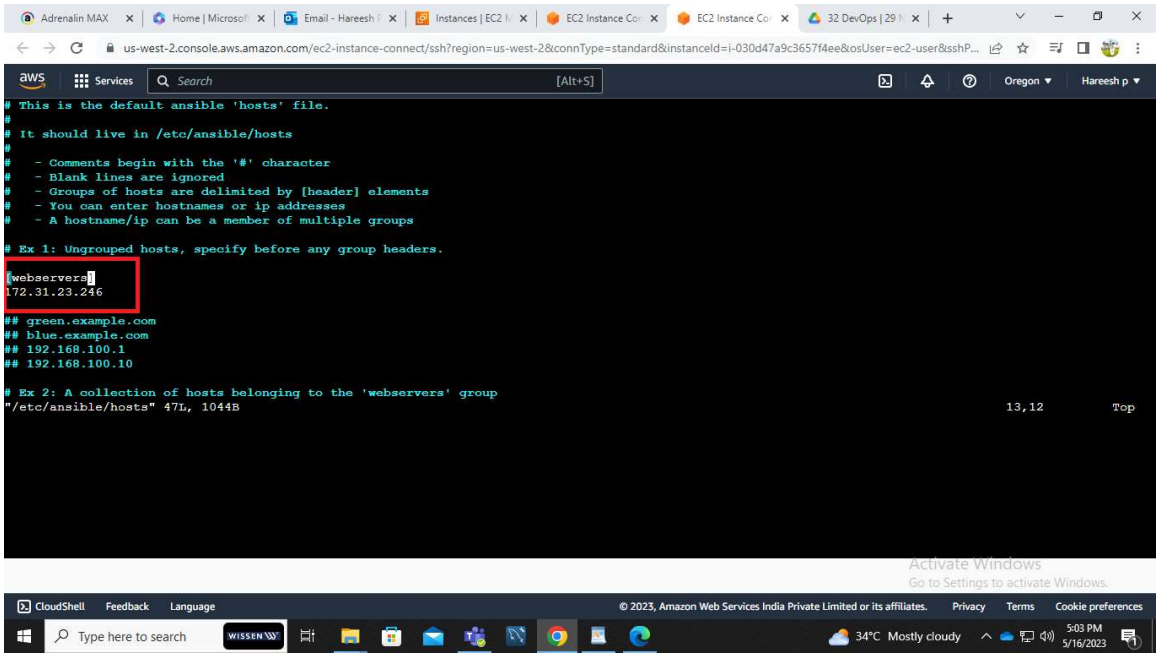
- In ansible we have two files

1. ansible.cfg - this the main ansible configuration file.

2. Hosts or inventory file - In this file we add node IP address then only ansible server recognise those nodes.

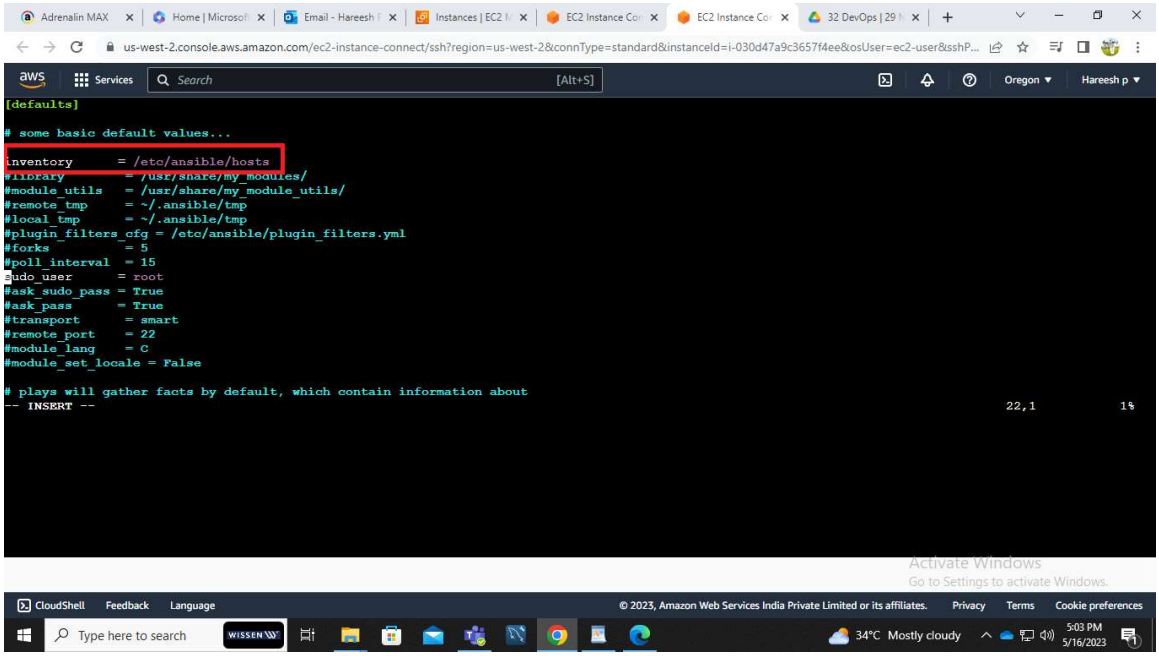


- open the hosts or inventory file and add groups under the group add Nodes IP address



- Open the ansible.cfg file and remove # comment in inventory file then only we can get

information form hosts file to Main file.



- **Host patterns : targeting hosts and groups**

1. An Ansible pattern can refer to a single host, an IP address, an inventory group, a set of groups, or all hosts in your inventory.

2. Patterns are highly flexible - you can exclude or require subsets of hosts, use wildcards or regular expressions, and more. Ansible executes on all inventory hosts included in the pattern.

- Follow the below commands to se numer of hosts list

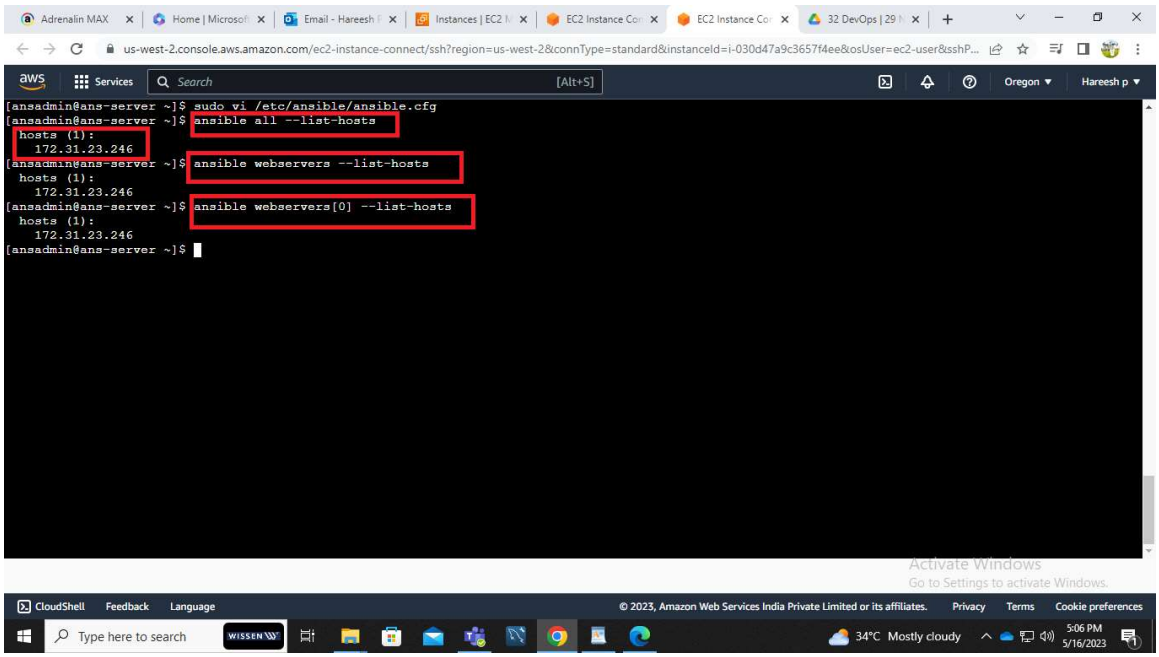
- Here ansible commands start with - ansible

- Here keyword - all (listed all hosts address)

1. Example we want to see all hosts in hosts file use command is
   - anasible all --list-hosts

2. we want to see particular group of hosts in hosts file command is
   - ansible webservers --list-hosts

3. we want to see list of first host in particular group use below command
   - ansible webservers[0] --list-hosts



- **For ansible information follow the below official website link**

https://docs.ansible.com/