Adhoc

modules

playbook

Adhoc :- these are simple linux commad that we run on linux server.

When we want simple work like create file, shutdown for multiple server we use this command.

We dont use it for heavy task like configuration management.

Adhoc use binary like /*usr/*bin/ansible, to automate a single task.

If we can run linux command like that why use module ??

so answer is adhoc command is **not idempotent**. Means if you run “touch file.txt” two time it will overwrite it.

So Adhoc command are not smart as module.

Module --> run single command both written in YAML

Playbook --> run more that one command or use multiple module

like if we want only to install httpd, we can done that using module.

But we want more that one task like:- Installing, httpd service start, coping html file , then we have to use three module. This we can mention in playbook.

**Adhoc Commands:-**  
  
 $ ansible all -a “<linux command>” => ansible all -a “ls”

here "commad" module is default.

=> ‘-a’ this is for argument

$ ansible web -a “touch file1.txt”

$ ansible web[0] -a “touch file1.txt”

$ ansible web -a “sudo yum install httpd -y”

or

$ ansible web -**b**a “yum install httpd -y”

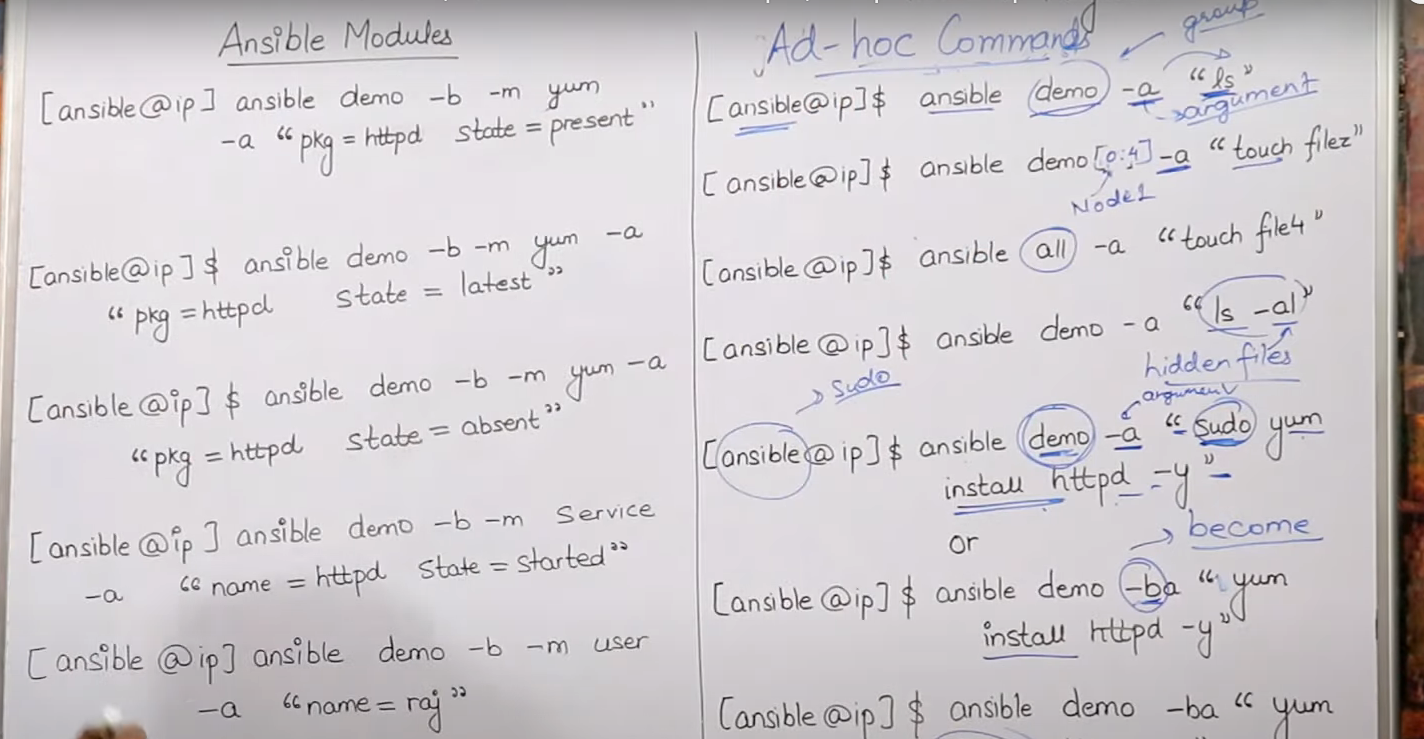
here ‘b’ means “become”/”sudo” so no need to mention sudo afterward.



Module:- this is single unit to represent one type of command. In backend it call linux command but in smatterway.

Default localtion to store module is :- /*etc*/ansible/hosts => same file for inventory.

Difference in module and adhoc command:-



$ ansible web -b -m yum -a “pkg=httpd state=present”

-b ==> become ie sudo previledge

-m ==> module ie yum

-a ==> argument ie argument for yum

yum to install pakage (pkg) , state=present (means you want to install package ).

this “state=present” is default as well, you can even skip.

But is you want to remove then you have type “state=absent”

so

install =====> present

uninstall ====> absent

update =====> latest

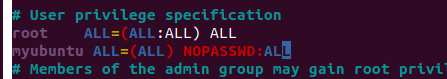
$ ansible web -b -m service -a “name=httpd state=started”

hint :- “rc” in ansible means return code.

**-k, --ask-pass**: ask for connection password

**-K, --ask-become-pass**: ask for privilege escalation password

or in node $sudo visudo



to give sudo priviledge.

tester ALL=(ALL:ALL) NOPASSWD:ALL # good

The meaning of these additional fields is: First “ALL” indicates that the user can run commands as all users. The second “ALL” indicates that the user can run commands as all groups.

$ ansible -K web -bm copy -a “src=/art/ dest=/tmp”

$ ansible web -m setup ==> this module store state for machine.

$ ansible web -m setup -a “filter=\*ipv4\*” to get only ip address.