Basic Ansible Ad-hoc Commands

What do we mean by Ad-hoc commands in Ansible ?

Ad hoc commands are commands which can be run individually to perform quick functions. These commands need not be performed later.

For example, you have to reboot all your company servers. For this, you will run the Adhoc commands from '**/usr/bin/ansible**'.

These ad-hoc commands are not used for configuration management and deployment, because these commands are of one time usage.

ansible-playbook is used for configuration management and deployment.

Let's take a look at few of the Ad-hoc commands in Ansible

1. **ansible all –m ping**

The very basic command in Ansible to check whether all nodes are reachable from controller or not, here –m stands for module, in place of **all** we can specify hostname of node as well, please find below given output for above command

```
[student@ansible ~]$ ansible all -m ping
ansible.controller.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "ping": "pong"
}
ansible.node1.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "ping": "pong"
}
ansible.node2.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "ping": "pong"
}
```

```
[student@ansible ~]$ ansible ansible.node1.com -m ping
ansible.node1.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "ping": "pong"
}
[student@ansible ~]$ ansible ansible.node2.com -m ping
ansible.node2.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "ping": "pong"
}
```

If execution of above command is successful then we should response as "pong" for "ping"

2. ansible all –list

Let's consider we have no idea about which nodes are connected to our controller, using above given command we can get this information, please find below given output for the same.

```
[student@ansible ~]$ ansible all --list
  hosts (3):
    ansible.controller.com
    ansible.node1.com
    ansible.node2.com
[student@ansible ~]$ 
```

Command and Shell module

In next phase of examples we are going to see few Ad-hoc commands where at some places we will see '*command*' and few of the examples with '*shell*' module. One should ask question, What is difference between the 2?

Command module is designed for **"bash"** shell and **Shell** module is designed for all types of shell available in Linux Operating System. We can use either of module to trigger Ansible commands.

1. ansible all –m command –a 'date'

As we have already discussed –m stands for Module, then Why do we use –a?, -a stands for attribute. In the above command we are using attribute date from command module.

```
[student@ansible ~]$ ansible all -m command -a 'date'
ansible.node1.com | CHANGED | rc=0 >>
Sat Oct 23 22:46:12 PDT 2021
ansible.node2.com | CHANGED | rc=0 >>
Sat Oct 23 22:46:12 PDT 2021
ansible.controller.com | CHANGED | rc=0 >>
Sat Oct 23 22:46:13 PDT 2021
```

2. ansible all –m command –a 'hostname'

```
[student@ansible ~]$ ansible all -m command -a 'hostname'
ansible.node1.com | CHANGED | rc=0 >>
ansible.node1.com
ansible.node2.com | CHANGED | rc=0 >>
ansible.node2.com
ansible.controller.com | CHANGED | rc=0 >>
ansible.controller.com
```

3. ansible all –m command –a 'free –h'

```
[student@ansible ~]$ ansible all -m command -a 'free -h'
ansible.node1.com | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:          780Mi       516Mi        59Mi       2.0Mi       205Mi       137Mi
Swap:         2.0Gi       103Mi       1.9Gi
ansible.node2.com | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:          782Mi       501Mi        62Mi       4.0Mi       218Mi       157Mi
Swap:         2.0Gi        86Mi       1.9Gi
ansible.controller.com | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:          782Mi       512Mi       101Mi       1.0Mi       168Mi       148Mi
Swap:         2.0Gi       200Mi       1.8Gi
```

4. ansible all –m command –a 'df –h .'

```
[student@ansible ~]$ ansible all -m command -a 'df -h .'
ansible.node1.com | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3        18G  5.3G   13G  30% /
ansible.node2.com | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3        18G  5.7G   13G  32% /
ansible.controller.com | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3        18G  5.1G   13G  29% /
```

5. ansible ansible.node1.com –m command –a 'lscpu'

```
[student@ansible ~]$ ansible ansible.node1.com -m command -a 'lscpu'
ansible.node1.com | CHANGED | rc=0 >>
Architecture:        x86_64
CPU op-mode(s):      32-bit, 64-bit
Byte Order:          Little Endian
CPU(s):              1
On-line CPU(s) list: 0
Thread(s) per core:  1
Core(s) per socket:  1
Socket(s):           1
NUMA node(s):        1
Vendor ID:           GenuineIntel
CPU family:          6
Model:               42
Model name:          Intel(R) Core(TM) i3-2377M CPU @ 1.50GHz
Stepping:            7
CPU MHz:             1496.599
BogoMIPS:            2993.19
Hypervisor vendor:   VMware
Virtualization type: full
L1d cache:           32K
L1i cache:           32K
L2 cache:            256K
L3 cache:            3072K
NUMA node0 CPU(s):   0
Flags:               fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 syscall nx rdtscp lm constant_tsc arch_perfmon no
pl xtopology tsc_reliable nonstop_tsc cpuid pni pclmulqdq ssse3 cx16 pcid sse4_1 sse4_2 x2apic popcnt tsc_deadline_timer xsave avx hypervisor lahf_lm pti tsc_adjust ara
t
```

6. ansible all –m shell –a 'date'

```
[student@ansible ~]$ ansible all -m shell -a 'date'
ansible.node2.com | CHANGED | rc=0 >>
Sun Oct 24 00:47:34 PDT 2021
ansible.controller.com | CHANGED | rc=0 >>
Sun Oct 24 00:47:37 PDT 2021
ansible.node1.com | CHANGED | rc=0 >>
Sun Oct 24 00:47:39 PDT 2021
```

Above output shows, irrespective of whether we are using 'command' module or shell 'module' we are getting same output for the same.

7. ansible all –m shell –a 'hostname'

```
[student@ansible ~]$ ansible all -m shell -a 'hostname'
ansible.node1.com | CHANGED | rc=0 >>
ansible.node1.com
ansible.node2.com | CHANGED | rc=0 >>
ansible.node2.com
ansible.controller.com | CHANGED | rc=0 >>
ansible.controller.com
```

8. ansible all –m shell –a 'free –h'

```
[student@ansible ~]$ ansible all -m shell -a 'free -h'
ansible.node1.com | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:          780Mi       510Mi        87Mi       3.0Mi       183Mi       142Mi
Swap:         2.0Gi       110Mi       1.9Gi
ansible.node2.com | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:          782Mi       473Mi        86Mi       3.0Mi       222Mi       185Mi
Swap:         2.0Gi       119Mi       1.9Gi
ansible.controller.com | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:          782Mi       503Mi       128Mi       1.0Mi       151Mi       157Mi
Swap:         2.0Gi       265Mi       1.7Gi
```

9. ansible all –m shell –a 'df –h .'

```
[student@ansible ~]$ ansible all -m shell -a 'df -h .'
ansible.node1.com | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3        18G  5.3G   13G  30% /
ansible.node2.com | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3        18G  5.7G   13G  32% /
ansible.controller.com | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3        18G  5.1G   13G  29% /
```

10. ansible all –m shell –a 'lscpu'

```
[student@ansible ~]$ ansible all -m shell -a 'lscpu'
ansible.node1.com | CHANGED | rc=0 >>
Architecture:        x86_64
CPU op-mode(s):      32-bit, 64-bit
Byte Order:          Little Endian
CPU(s):              1
On-line CPU(s) list: 0
Thread(s) per core:  1
Core(s) per socket:  1
Socket(s):           1
NUMA node(s):        1
Vendor ID:           GenuineIntel
CPU family:          6
Model:               42
Model name:          Intel(R) Core(TM) i3-2377M CPU @ 1.50GHz
Stepping:            7
CPU MHz:             1496.599
BogoMIPS:            2993.19
Hypervisor vendor:   VMware
Virtualization type: full
L1d cache:           32K
L1i cache:           32K
L2 cache:            256K
L3 cache:            3072K
NUMA node0 CPU(s):   0
Flags:               fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 syscall nx rdtscp lm constant_tsc arch_perfmon no
pl xtopology tsc_reliable nonstop_tsc cpuid pni pclmulqdq ssse3 cx16 pcid sse4_1 sse4_2 x2apic popcnt tsc_deadline_timer xsave avx hypervisor lahf_lm pti tsc_adjust ara
t
ansible.node2.com | CHANGED | rc=0 >>
Architecture:        x86_64
CPU op-mode(s):      32-bit, 64-bit
Byte Order:          Little Endian
CPU(s):              1
On-line CPU(s) list: 0
Thread(s) per core:  1
Core(s) per socket:  1
```

11. ansible all –m shell –a 'cal;date;'

```
[student@ansible ~]$ ansible all -m shell -a 'cal;date'
ansible.node1.com | CHANGED | rc=0 >>
    October 2021
Su Mo Tu We Th Fr Sa
                1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
31
Sun Oct 24 01:07:36 PDT 2021
ansible.node2.com | CHANGED | rc=0 >>
    October 2021
Su Mo Tu We Th Fr Sa
                1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
31
Sun Oct 24 01:07:36 PDT 2021
ansible.controller.com | CHANGED | rc=0 >>
    October 2021
Su Mo Tu We Th Fr Sa
                1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
31
Sun Oct 24 01:07:36 PDT 2021
```

File module

Like Command and Shell module we have few more modules in Ansible, one of them is file module. Basically this module is being used for creating / removing files and directories and performing all other operations on files and directories. Let's have a look at few commands for file module.

1. ansible all –m file –a 'name=/tmp/unnati state=directory'

```
[student@ansible ~]$ ansible all -m file -a 'name=/tmp/unnati state=directory'
ansible.node2.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 1000,
    "group": "student",
    "mode": "0775",
    "owner": "student",
    "path": "/tmp/unnati",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 1000
}
ansible.node1.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 1000,
    "group": "student",
    "mode": "0775",
    "owner": "student",
    "path": "/tmp/unnati",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 1000
}
```

```
[student@ansible ~]$ ansible all -m file -a 'name=/tmp/unnati state=directory'
ansible.node2.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "gid": 1000,
    "group": "student",
    "mode": "0775",
    "owner": "student",
    "path": "/tmp/unnati",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 1000
}
ansible.controller.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "gid": 1000,
    "group": "student",
    "mode": "0775",
    "owner": "student",
    "path": "/tmp/unnati",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 1000
}
ansible.node1.com | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
```

As we can see, both above given output snapshots shows success message the why both outputs have different colors, it is because first output shows change has been performed and execution is successful, whereas in second output even if execution is successful no changes were made, hence color of output is green.

2. ansible all –m file –a 'name=/tmp/user1 state=directory mode=0777'

```
[student@ansible ~]$ ansible all -m file -a 'name=/tmp/user1 state=directory mode=
ansible.node2.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 1000,
    "group": "student",
    "mode": "0777",
    "owner": "student",
    "path": "/tmp/user1",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 1000
}
ansible.node1.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 1000,
    "group": "student",
    "mode": "0777",
    "owner": "student",
    "path": "/tmp/user1",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 1000
}
```

In this example **mode** refers to file permissions (0777, where 0 is sticky bit and 777 are file permissions),

```
[student@ansible ~]$ ansible all -m shell -a 'ls -lrt /tmp|tail -3'
ansible.node1.com | CHANGED | rc=0 >>
drwxrwxr-x. 2 student student    6 Oct 24 01:22 unnati
drwxrwxrwx. 2 student student    6 Oct 24 01:29 user1
drwx------. 2 student student   41 Oct 24 05:48 ansible_command_payload_9500ip5u
ansible.node2.com | CHANGED | rc=0 >>
drwxrwxr-x. 2 student student    6 Oct 24 01:22 unnati
drwxrwxrwx. 2 student student    6 Oct 24 01:29 user1
drwx------. 2 student student   41 Oct 24 05:48 ansible_command_payload_ewck1052
ansible.controller.com | CHANGED | rc=0 >>
drwxrwxr-x. 2 student student    6 Oct 24 01:22 unnati
drwxrwxrwx. 2 student student    6 Oct 24 01:29 user1
drwx------. 2 student student   41 Oct 24 05:48 ansible_command_payload_s78g4gcy
```

Let say , if we just want to change permissions of user1 directory then we should use, below given command.

3. Ansible all –m file –a 'name=/tmp/unnati mode=0700'

```
[student@ansible ~]$ ansible all -m file -a 'name=/tmp/unnati mode=0700'
ansible.node1.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 1000,
    "group": "student",
    "mode": "0700",
    "owner": "student",
    "path": "/tmp/unnati",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 1000
}
ansible.node2.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 1000,
    "group": "student",
    "mode": "0700",
    "owner": "student",
```

```
[student@ansible ~]$ ansible all -m shell -a 'ls -lrt /tmp|tail -3'
ansible.node1.com | CHANGED | rc=0 >>
drwx------. 2 student student    6 Oct 24 01:22 unnati
drwxrwxrwx. 2 student student    6 Oct 24 01:29 user1
drwx------. 2 student student   41 Oct 24 05:53 ansible_command_payload_7gy0sbwn
ansible.node2.com | CHANGED | rc=0 >>
drwx------. 2 student student    6 Oct 24 01:22 unnati
drwxrwxrwx. 2 student student    6 Oct 24 01:29 user1
drwx------. 2 student student   41 Oct 24 05:53 ansible_command_payload_og1qb0s0
ansible.controller.com | CHANGED | rc=0 >>
drwx------. 2 student student    6 Oct 24 01:22 unnati
drwxrwxrwx. 2 student student    6 Oct 24 01:29 user1
drwx------. 2 student student   41 Oct 24 05:53 ansible_command_payload_cmg95ym8
```

If we want to create a file then value of **"state"** parameter should be **touch**

4. ansible all -m file -a 'name=/tmp/unnati/test.txt state=touch'

```
[student@ansible ~]$ ansible all -m file -a 'name=/tmp/unnati/test.txt state=touch'
ansible.node1.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "dest": "/tmp/unnati/test.txt",
    "gid": 1000,
    "group": "student",
    "mode": "0664",
    "owner": "student",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 0,
    "state": "file",
    "uid": 1000
}
ansible.node2.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "dest": "/tmp/unnati/test.txt",
    "gid": 1000,
    "group": "student",
    "mode": "0664",
    "owner": "student",
    "secontext": "unconfined_u:object_r:user_tmp_t:s0",
    "size": 0,
    "state": "file",
    "uid": 1000
}
```

```
[student@ansible ~]$ ansible all -m shell -a 'ls -lrt /tmp/unnati/'
ansible.node1.com | CHANGED | rc=0 >>
total 0
-rw-rw-r--. 1 student student 0 Oct 24 06:02 test.txt
ansible.node2.com | CHANGED | rc=0 >>
total 0
-rw-rw-r--. 1 student student 0 Oct 24 06:02 test.txt
ansible.controller.com | CHANGED | rc=0 >>
total 0
-rw-rw-r--. 1 student student 0 Oct 24 06:02 test.txt
```

To remove / delete file / directory, we need to pass "absent" value to state parameter

5. ansible all -m file -a 'name=/tmp/unnati/test.txt state=absent'

```
[student@ansible ~]$ ansible all -m file -a 'name=/tmp/unnati/test.txt state=absent'
ansible.node1.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "path": "/tmp/unnati/test.txt",
    "state": "absent"
}
ansible.node2.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "path": "/tmp/unnati/test.txt",
    "state": "absent"
}
ansible.controller.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "path": "/tmp/unnati/test.txt",
    "state": "absent"
}
```

```
[student@ansible ~]$ ansible all -m shell -a 'ls -lrt /tmp/unnati/'
ansible.node1.com | CHANGED | rc=0 >>
total 0
ansible.node2.com | CHANGED | rc=0 >>
total 0
ansible.controller.com | CHANGED | rc=0 >>
total 0
```

Let's try to create file or directory at / location

6. ansible all -m file -a 'name=/unnati state=directory'

```
[student@ansible ~]$ ansible all -m file -a 'name=/unnati state=directory'
ansible.node1.com | FAILED! => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "msg": "There was an issue creating /unnati as requested: [Errno 13] Permission denied: b'/unnati'",
    "path": "/unnati"
}
ansible.node2.com | FAILED! => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "msg": "There was an issue creating /unnati as requested: [Errno 13] Permission denied: b'/unnati'",
    "path": "/unnati"
```

It has got failed, as to create file or directory at / location, one should have admin or root access.
Use "--become" keyword to achieve this task.

ansible all -m file -a 'name=/unnati state=directory' --become

```
[student@ansible ~]$ ansible all -m file -a 'name=/unnati state=directory' --become
ansible.node2.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 0,
    "group": "root",
    "mode": "0755",
    "owner": "root",
    "path": "/unnati",
    "secontext": "unconfined_u:object_r:default_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 0
}
ansible.node1.com | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": true,
    "gid": 0,
    "group": "root",
    "mode": "0755",
    "owner": "root",
```

```
[student@ansible ~]$ ansible all -m shell -a 'ls -lrt /|tail -3'
ansible.node1.com | CHANGED | rc=0 >>
drwxr-xr-x.  42 root   root  1220 Oct 23 22:15 run
drwxr-xr-x.   2 root   root     6 Oct 24 06:15 unnati
drwxrwxrwt.  25 root   root  4096 Oct 24 06:17 tmp
ansible.node2.com | CHANGED | rc=0 >>
drwxr-xr-x.  42 root   root  1220 Oct 23 22:15 run
drwxr-xr-x.   2 root   root     6 Oct 24 06:15 unnati
drwxrwxrwt.  20 root   root  4096 Oct 24 06:17 tmp
ansible.controller.com | CHANGED | rc=0 >>
drwxr-xr-x.  42 root   root  1220 Oct 23 22:14 run
drwxr-xr-x.   2 root   root     6 Oct 24 06:15 unnati
drwxrwxrwt.  27 root   root  4096 Oct 24 06:17 tmp
[student@ansible ~]$ date
Sun Oct 24 06:17:57 PDT 2021
```

In place of –become , one can use –b option as well