

## Fact Variable

1. In this lab you are going to learn about some more variable which is gathering fact.
2. So, whenever you run ansible playbook command it will always gather fact.

```
ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ ansible-playbook -i inventory demo_prec.yaml

PLAY [Understanding vars] ****
TASK [Gathering Facts] ****
ok: [demoweb02]
ok: [demoweb01]
ok: [demodb01]
```

3. Now if you just run this ad hoc command. You will see that you got a very long output which will be in JASON format. And if you read that output you will see that the whole output is a huge dictionary. In that you have keys like ansible facts and in the you have key values.
4. So, the below is just for one machine which is demoweb01. You can also try and run for other machines or for the whole setup.

```
ansible -m setup -i inventory demoweb01
ansible -m setup -i inventory all
```

```
ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ ansible -m setup -i inventory demoweb01
demoweb01 | SUCCESS => {
    "ansible_facts": {
        "ansible_all_ipv4_addresses": [
            "172.31.9.178"
        ],
        "ansible_all_ipv6_addresses": [
            "fe80::8d7:88ff:fe48:e767"
        ],
        "ansible_apparmor": {
            "status": "disabled"
        }
    }
}
```

5. Now we'll try to print these variables for that we need to create a new directory.

```
vim facts.yaml
```

```
---
- name: Print facts
  hosts: all
  tasks:
    - name: Print OS name
      debug:
        var: ansible_distribution
```

```
ubuntu@ip-172-31-12-170: ~/i  X  +  |  ~
---
- name: Print facts
  hosts: all
  #gather_facts: False
  tasks:
    - name: Print OS name
      debug:
        var: ansible_distribution
~
```

6. Here you can see the output.

```
ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ vim facts.yaml
ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ ansible-playbook -i inventory facts.yaml

PLAY [Print facts] ****
TASK [Gathering Facts] ****
ok: [demoweb02]
ok: [demodb01]
ok: [demoweb01]

TASK [Print OS name] ****
ok: [demoweb01] => {
    "ansible_distribution": "CentOS"
}
ok: [demoweb02] => {
    "ansible_distribution": "CentOS"
}
ok: [demodb01] => {
    "ansible_distribution": "CentOS"
}

PLAY RECAP ****
demodb01 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb01 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb02 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ |
```

7. Now if you will see we only have CentOS in here. So, you are going to create a new machine with Ubuntu OS and launch that with the same key pair and security as CentOS machines.
8. Once it started running copy its private IP and come back to inventory file to update it. Just copy the previous set and paste it change its name and IP then save and quit.

**vim inventory**

```
ubuntu@ip-172-31-12-170: ~/. + | ^

all:
  hosts:
    demoweb01:
      ansible_host: 172.31.9.178
      ansible_user: ec2-user
      ansible_ssh_private_key_file: demousrkey.pem
    demoweb02:
      ansible_host: 172.31.8.250
      ansible_user: ec2-user
      ansible_ssh_private_key_file: demousrkey.pem
    demoweb03:
      ansible_host: 172.31.11.210
      ansible_user: ec2-user
      ansible_ssh_private_key_file: demousrkey.pem
  demodb01:
    ansible_host: 172.31.0.93
    ansible_user: ec2-user
    ansible_ssh_private_key_file: demousrkey.pem

  children:
    webservers:
      hosts:
        demoweb01:
        demoweb02:
        demoweb03:
    dbservers:
      hosts:
        demodb01:
    demoall:
      children:
        webservers:
        dbservers:
```

9. Now if you will try to ping all the servers, you will see that our new server has failed because it is an Ubuntu OS.
10. So, such cases we should have their own separate variables. The host variables will take higher priority, so it will check first whether the host has those values or not. If it's not, then only it will go for global.
11. Now we need to define it in the code.

```

ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ ansible -m ping all -i inventory
demoweb03 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Warning: Permanently added '172.31.11.210' (ED25519) to the list of known hosts.\r\nnec2-user@172.31.11.210: Permission denied (publickey).",
    "unreachable": true
}
demoweb01 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
demodb01 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
demoweb02 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ |

```

12. Again, open your inventory. You need to change the user's name. Then just run the command again.

```

demoweb03:
  ansible_host: 172.31.11.210
  ansible_user: ubuntu
  ansible_ssh_private_key_file: demousrkey.pem

```

13. And you will see that this it has successfully executed.

```

ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ ansible -m ping all -i inventory
demodb01 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
demoweb01 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
demoweb02 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
demoweb03 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ |

```

14. Now if you will run the ansible playbook command again you will that our new machine has join the league and it is showing its OS.

**ansible-playbook -i inventory facts.yaml**

```

ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ ansible-playbook -i inventory facts.yaml

PLAY [Print facts] ****
TASK [Gathering Facts] ****
ok: [demoweb01]
ok: [demodb01]
ok: [demoweb02]
ok: [demoweb03]

TASK [Print OS name] ****
ok: [demoweb01] => {
    "ansible_distribution": "CentOS"
}
ok: [demoweb02] => {
    "ansible_distribution": "CentOS"
}
ok: [demoweb03] => {
    "ansible_distribution": "Ubuntu"
}
ok: [demodb01] => {
    "ansible_distribution": "CentOS"
}

PLAY RECAP ****
demodb01 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb01 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb02 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb03 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ |

```

15. Now go back to your facts YAML file and add some lines of code.

```

---
- name: Print facts
  hosts: all
  #gather_facts: False
  tasks:
    - name: Print OS name
      debug:
        var: ansible_distribution
    - name: Print selinux mode
      debug:
        var: ansible_selinux.mode
    - name: Print RAM memory
      debug:
        var: ansible_memory_mb.real.free

    - name: Print Processor name
      debug:
        var: ansible_processor[2]

```

```
ubuntu@ip-172-31-12-170: ~/l + | v

---
- name: Print facts
  hosts: all
  #gather_facts: False
  tasks:
    - name: Print OS name
      debug:
        var: ansible_distribution
    - name: Print selinux mode
      debug:
        var: ansible_selinux.mode
    - name: Print RAM memory
      debug:
        var: ansible_memory_mb.real.free
    - name: Print Processor name
      debug:
        var: ansible_processor[2]
```

16. Then run the ansible playbook command. Below you can see that it is showing you all the information.

```
ubuntu@ip-172-31-12-170:~/demodirectory/exc8$ ansible-playbook -i inventory facts.yaml

PLAY [Print facts] ****
TASK [Gathering Facts] ****
ok: [demoweb01]
ok: [demoweb02]
ok: [demodb01]
ok: [demoweb03]

TASK [Print OS name] ****
ok: [demoweb01] => {
    "ansible_distribution": "CentOS"
}
ok: [demoweb02] => {
    "ansible_distribution": "CentOS"
}
ok: [demoweb03] => {
    "ansible_distribution": "Ubuntu"
}
ok: [demodb01] => {
    "ansible_distribution": "CentOS"
}
```

```

TASK [Print selinux mode] ****
ok: [demoweb01] => {
    "ansible_selinux.mode": "enforcing"
}
ok: [demoweb02] => {
    "ansible_selinux.mode": "enforcing"
}
ok: [demoweb03] => {
    "ansible_selinux.mode": "VARIABLE IS NOT DEFINED!"
}
ok: [demodb01] => {
    "ansible_selinux.mode": "enforcing"
}

TASK [Print RAM memory] ****
ok: [demoweb01] => {
    "ansible_memory_mb.real.free": "410"
}
ok: [demoweb02] => {
    "ansible_memory_mb.real.free": "400"
}
ok: [demoweb03] => {
    "ansible_memory_mb.real.free": "382"
}
ok: [demodb01] => {
    "ansible_memory_mb.real.free": "274"
}

TASK [Print Processor name] ****
ok: [demoweb01] => {
    "ansible_processor[2)": "Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz"
}
ok: [demoweb02] => {
    "ansible_processor[2)": "Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz"
}
ok: [demoweb03] => {
    "ansible_processor[2)": "Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz"
}
ok: [demodb01] => {
    "ansible_processor[2)": "Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz"
}

PLAY RECAP ****
demodb01 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb01 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb02 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
demoweb03 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```