Tasks

- Tasks are the application of a module to perform a specific unit of work.

- **file**: A directory should exist
- **yum**: A package should be installed
- **service**: A service should be running
- **template**: Render a configuration file from a template
- **get_url**: Fetch an archive file from a URL
- **git**: Clone a source code repository

Example Tasks in a Play

```yaml
tasks:
- name: httpd package is present
  yum:
    name: httpd
    state: latest


- name: latest index.html file is present
  copy:
    src: files/index.html
    dest: /var/www/html/


- name: restart httpd
  service:
    name: httpd
    state: restarted
```

Playbook Example

```
---

- name: install and start apache

  hosts: webserver

  become: yes

  vars:

    http_port: 80


  tasks:

  - name: httpd package is present

    yum:

      name: httpd

      state: latest
```

Host Selector

Privilege Escalation

Play Variables

Human-Meaningful Naming

Task

Executing Playbooks

▶ Run playbook

ansible-playbook playbook.yml

▶ If you want to start executing your playbook at a particular task, you can do so with the --start-at-task option:

ansible-playbook playbook.yml --start-at-task="install packages"

▶ Playbooks can also be executed interactively with --step

ansible-playbook playbook.yml --step

This will cause ansible to stop on each task, and ask if it should execute that task. Say you had a task called "configure ssh", the playbook run will stop and ask:

Perform task: configure ssh (y/n/c):

Answering "y" will execute the task, answering "n" will skip the task, and answering "c" will continue executing all the remaining tasks without asking.

Without specifying Ansible's **name** module, Ansible will give human-readable names to each command

...

- hosts: all
 tasks:
 - yum:
    name=ntp
    state=installed
 - service:
    name=ntpd
    state=started
    enabled=yes

```
[root@ip-172-31-21-214 DevOps]# ansible-playbook 1.yml

PLAY [all] ***********************************************************

TASK [Gathering Facts] **********************************************
ok: [13.127.202.133]

TASK [Ensure NTP (for time synchronization) is installed.] *********
changed: [13.127.202.133]

TASK [Ensure NTP is running.] **************************************
changed: [13.127.202.133]

PLAY RECAP *********************************************************
13.127.202.133            : ok=3    changed=2    unreachable=0    failed=0

[root@ip-172-31-21-214 DevOps]# vim 1.yml
[root@ip-172-31-21-214 DevOps]# ansible-playbook 1.yml

PLAY [all] ***********************************************************

TASK [Gathering Facts] **********************************************
ok: [13.127.202.133]

TASK [Ensure NTP (for time synchronization) is installed.] *********
ok: [13.127.202.133]

TASK [Ensure NTP is running.] **************************************
ok: [13.127.202.133]

PLAY RECAP *********************************************************
13.127.202.133            : ok=3    changed=0    unreachable=0    failed=0
```

Loops

▶ Loops can do one task on multiple things, such as create a lot of users, install a lot of packages, or repeat a polling step until a certain result is reached.

- yum:

    name: **"{{ item }}"**

    state: latest

  with_items:

  - httpd

  - smtp

5.yml

 - name: Playbook with multiple items

   hosts: all

   tasks:

   - name: Install required packages

    yum:

     name={{ item }}

     state=installed

    with_items:

    - httpd

    - ntp

    - samba

```
[root@ip-172-31-29-57 DevOps]# ansible-playbook 4.yml

PLAY [Playbook with multiple items] ********************************************

TASK [Gathering Facts] *********************************************************
ok: [13.127.244.80]

TASK [Install required packages] ***********************************************
changed: [13.127.244.80] => (item=[u'httpd', u'mariadb', u'samba'])

PLAY RECAP *********************************************************************
13.127.244.80              : ok=2    changed=1    unreachable=0    failed=0
```

6.yml

- name: Playbook with multiple items

  hosts: all

  tasks:

  - name: 'Install required packages'
    yum:
      name={{ item }}
      state=installed
    with_items:
     - httpd
     - ntp
     - samba
     - targetcli

  - name: "Start required services"
    service:
      name={{ item }}
      state=started

    with_items:
     - httpd
     - ntpd
     - smb
     - target

```
[root@ip-172-31-29-57 DevOps]# ansible-playbook 5.yml

PLAY [Playbook with multiple items] ************************************

TASK [Gathering Facts] ************************************************
ok: [13.127.244.80]

TASK [Install required packages] **************************************
changed: [13.127.244.80] => (item=[u'httpd', u'ntp', u'samba', u'targetcli'])

TASK [Start required services] ****************************************
changed: [13.127.244.80] => (item=httpd)
changed: [13.127.244.80] => (item=ntpd)
changed: [13.127.244.80] => (item=smb)
ok: [13.127.244.80] => (item=target)

PLAY RECAP ************************************************************
13.127.244.80              : ok=3    changed=2    unreachable=0    failed=0
```