Red Hat
Training and
Certification

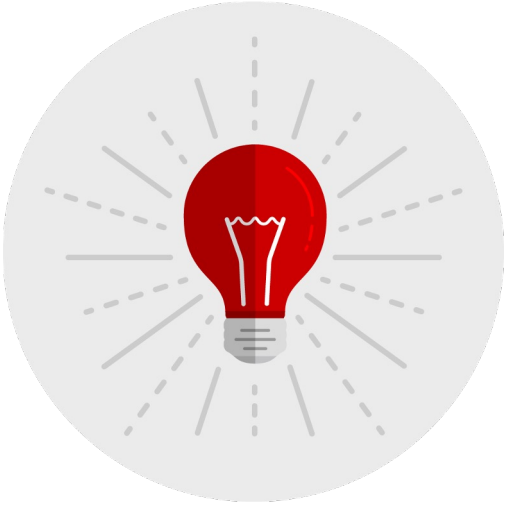
RED HAT
OPEN INNOVATION LABS

DevOps Culture and Practice Enablement

Ansible Overview

Red Hat

Topics

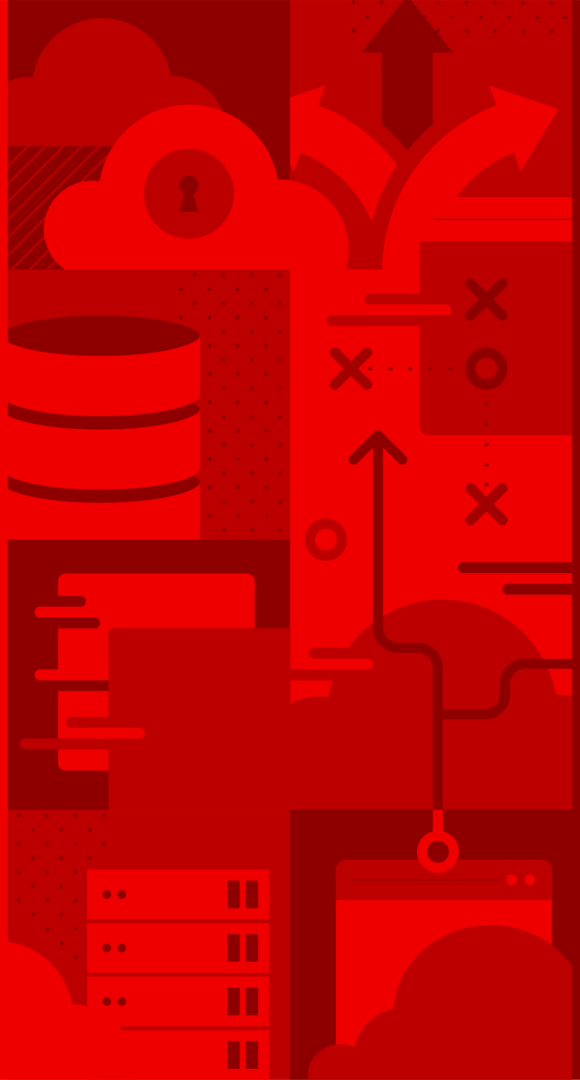


Ansible
What is Ansible?

Ansible Playbooks
What are Ansible playbooks and how are they used?

Ansible Roles
What are Ansible roles and how are they used?

Ansible Collections
What are Ansible collections and how are they used?



Ansible

Red Hat

Ansible Introduction & Overview

Ansible – An automation language leveraging modules to be used in one or more tasks on managed systems. Most Ansible automation leverages and Ansible playbook which is a YAML formatted file containing Ansible directives.

Ansible modules – Components used by Ansible tasks and playbooks which are generally implemented and developed in Python. Ansible modules work with certain system utilities and are optimized to be leveraged as a declarative automation language and provide idempotency.

Ansible ad-hoc commands – A way of executing a single Ansible task quickly that relies on a single Ansible module to perform the tests/changes of the task.

Ansible Roles– Curated, independent components of Ansible allowing re-use of common tasks. Roles can consist of variables, tasks, handlers, files, and templates.

Ansible Components & Commands

- **ansible.cfg** – Configuration file for running the **ansible** and **ansible-playbook** commands.
- **Inventory** – Inventory file used by the **ansible** and **ansible-playbook** commands identifying managed hosts/nodes and also contains Ansible inventory variables
- **ansible** – Command used to perform/execute Ansible ad-hoc commands against a managed node.
- **ansible-playbook** – Command used to execute and run Ansible playbooks
- **ansible-galaxy** – Command to create or utilize Ansible roles. Many of these roles are published on http://galaxy.ansible.com
- **Playbook** – Collection of Ansible tasks organized into one or more Ansible plays.
- **Task** – Collection/list of Ansible modules arranged into instructions. Each task utilizes an Ansible module to perform a given action.
- **Ansible Module** – Specific module (small program generally implemented in Python) which perform the commands and executes the program to get the desired state of a given task.

ansible.cfg File

- This file defines the configuration directives which apply directly to how the **ansible** and **ansible-playbook** command interact with the Ansible application and which configuration items are applied to a given Ansible session.
  - **./ansible.cfg** – When located in the current working directory (CWD) this file is the highest precedence.
  - **~/.ansible.cfg** – When located in the user's home directory, this file will have precedence if an *ansible.cfg* doesn't exist in the CWD.
  - **/etc/ansible/ansible.cfg** – This is the default configuration file and has the lowest precedence. This file is used when no other *ansible.cfg* file exists.

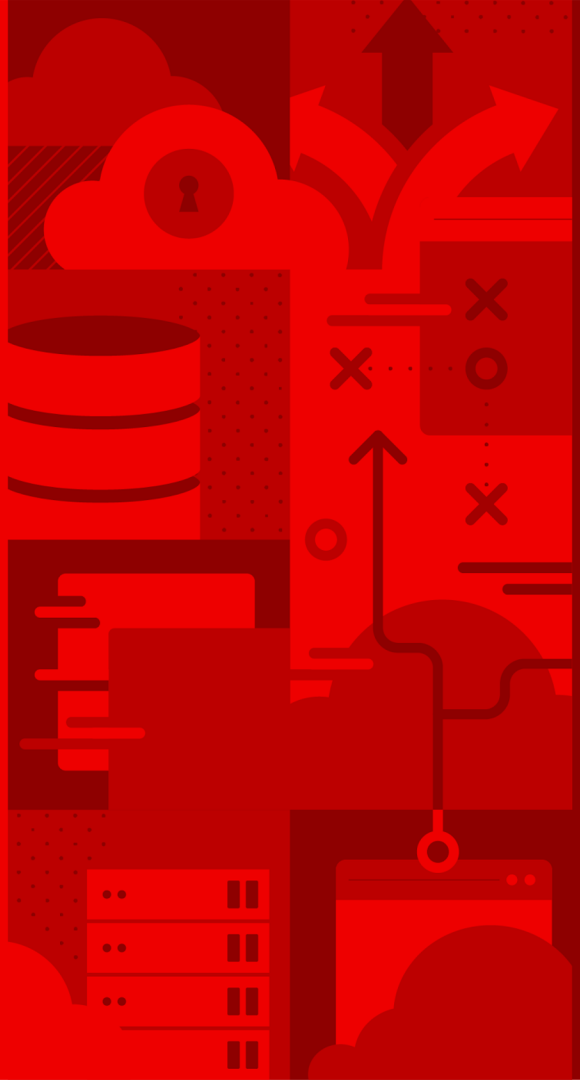
Important	It is also possible to define the **ansible.cfg** file with the environment variable **ANSIBLE_CONFIG**. If this variable is used, it will override all other configuration files.

Inventory File

The **inventory** file can contain both Ansible managed hosts/nodes as well as inventory variables to be used for the managed nodes.

- Inventory location is generally specified by the **ansible.cfg** file
  - **./inventory** – Common practice to leverage inventory files with playbooks and the **ansible.cfg** file in the current working directory
  - **/etc/ansible/hosts** – Default inventory file deployed with the Ansible package

Red Hat



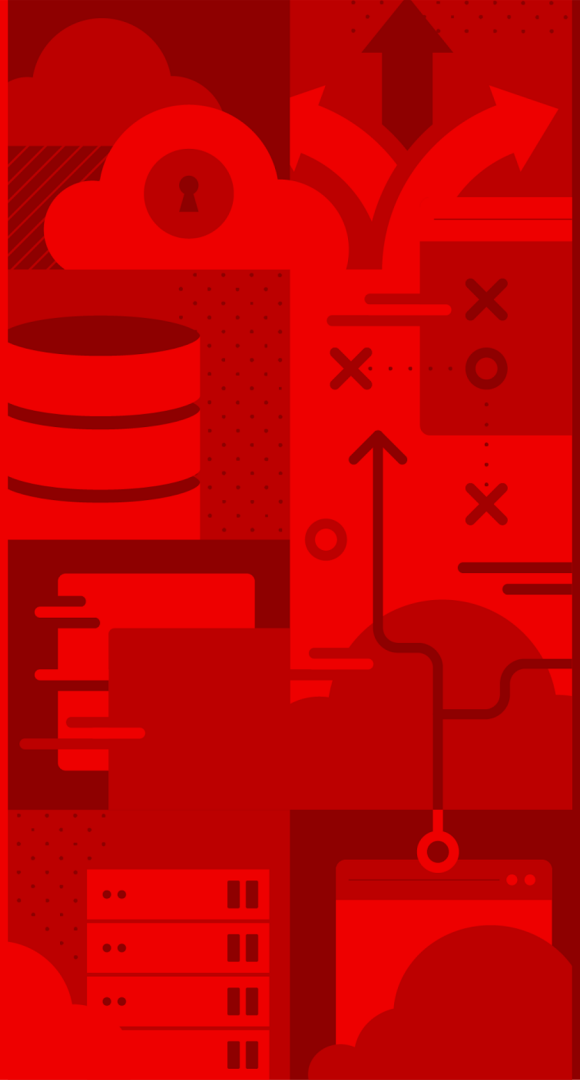
Ansible Playbooks

Playbook Structure

```
---
- name: Ansiblize Managed Hosts
  hosts: localhost
  vars_files:
    - variables.yml

  tasks:
    - name: Create Ansible User
      debug:
          msg: This will use the USER  module

  handlers:
    - name: Restart SSHD
      debug:
          msg: This sill use the SERVICE or SYSTEMD
```


Sample Playbook

```yaml
---
- name: Register Hosts to Satellite
  hosts: rhel83_test
  vars_files:
    - sat_vars.yml
  vars:
    activation_key: rhhi-vm
    lifecycle_env: Library
    Organization_Name: MichetteTech
    Sat_User_Name: sat_user_name
    Sat_User_PW: sat_password
    SatURL: https://sat6.michettetech.com

  tasks:
    - name: Prepare System for Satellite Registration - Get Cert
      get_url:
        url: "{{ SatURL }}/pub/katello-ca-consumer-latest.noarch.rpm"
        dest: /tmp/katello-latest.rpm
        validate_certs: no

    - name: Install Katello rpm
      yum:
        name: /tmp/katello-latest.rpm
        state: latest
        disable_gpg_check: yes
```



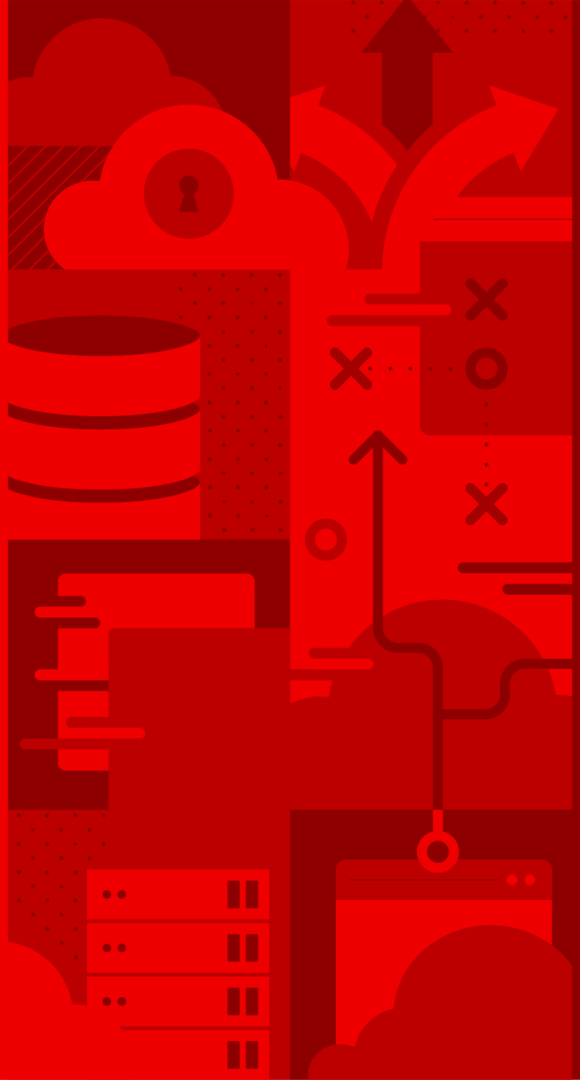
Ansible Roles

Ansible ROLE Structure

```
[student@workstation ROLES]$ tree My_Role/
My_Role/
├── defaults
│   └── main.yml
├── files
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── README.md
├── tasks
│   └── main.yml
├── templates
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    └── main.yml

8 directories, 8 files
```


Ansible ROLE Structure

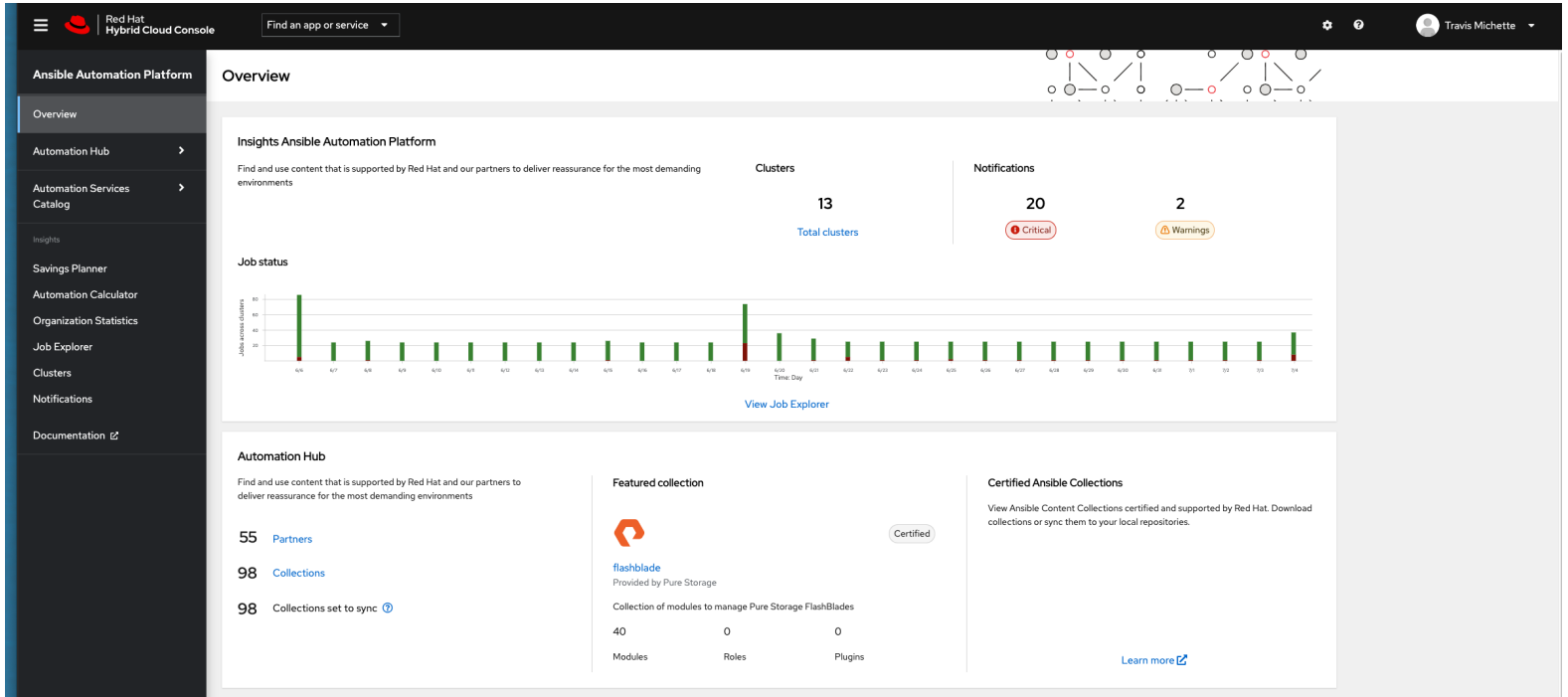
Subdirectory	Function of Directory
defaults	The **main.yml** file contains default variable values that are used by the role. These have the lowest precedence and priority.
files	Contains files that are used by the tasks in the role.
handlers	The **main.yml** file contains handlers that are executed by the role.
meta	The **main.yml** file contains information about the role. At a minimum you should modify the ***author***, ***license***, ***platforms***, and ***dependencies***.
tasks	The **main.yml** file contains the tasks being used by the Role.
templates	Contains Jinja2 templates used by tasks in the role.
tests	Contains an inventory and **test.yml** playbook that can be used to test the role.
vars	The **main.yml** file contains role variables and values. These are high precedence and not intended to be changed when used in a playbook.



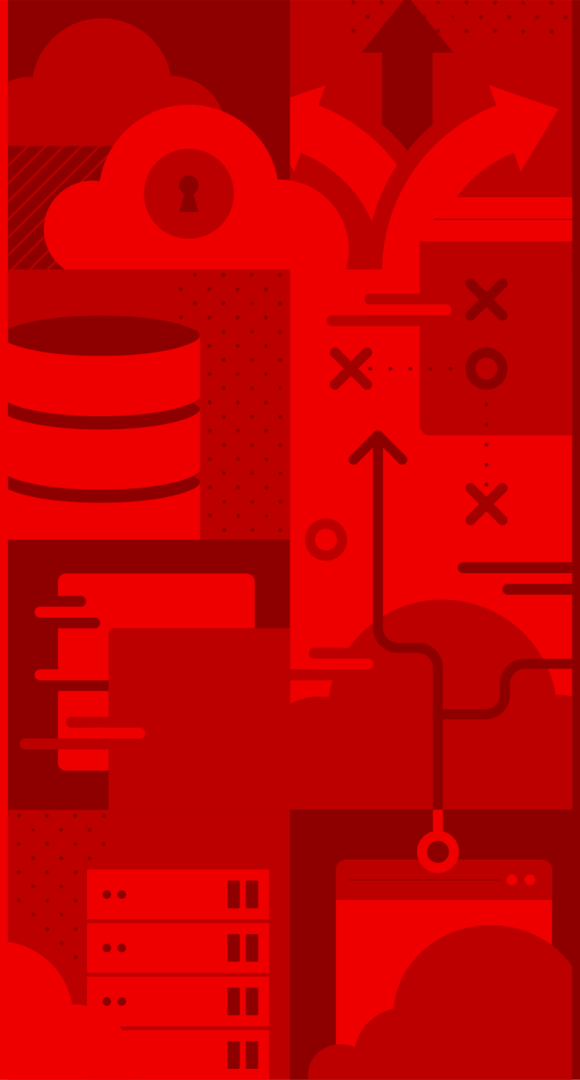
Ansible Collections

Ansible Collections – The Future

Ansible Collections – A new method and standard of delivering and distributing Ansible automation components. Collections can consist of Ansible modules, playbooks, roles, and plug-ins.



Red Hat



References
&
More Information

Additional Resources

Ansible Automation Platform – https://console.redhat.com/ansible/ansible-dashboard

Ansible Documentation – https://docs.ansible.com/ansible/latest/index.html

Github with Sample Playbooks – https://github.com/tmichett/AnsiblePlaybooks

Ansible Roles Creation (LUG Presentation)
Github - https://github.com/tmichett/LUG/tree/main/Ansible_Roles
YouTube - https://www.youtube.com/watch?v=UoY5QZ0dzbo&t=31s

Red Hat Training Courses: RH294, DO417 (Windows), DO447, DO457 (Networking)

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

https://www.facebook.com/RedHatServices

https://twitter.com/RedHat_Services

E-Mail: tmichett@redhat.com

Red Hat