(/rol/app/)

Search

Red Hat Enterprise Linux Automation with Ansible

📢 **FEEDBACK**   **TRANSLATIONS ▾**   **CERTIFICATE OF ATTENDANCE**

▤   📖   ⚡   ★

VIDEO CLASSROOM

Lab: Deploying Files to Managed Hosts ☆

Performance Checklist

In this lab, you will run a playbook that creates a customized file on your managed hosts by using a Jinja2 template.

Outcomes

You should be able to:

- Build a template file.
- Use the template file in a playbook.

  Log in to `workstation` as `student` using `student` as the password.

  On `workstation`, run the `lab file-review start` command. This ensures that Ansible is installed on `workstation`, creates the `/home/student/file-review` directory, and downloads the `ansible.cfg` file into that directory. It also downloads the `motd.yml`, `motd.j2`, `issue`, and `inventory` files into the `/home/student/file-review/files` directory.

```
[student@workstation ~]$ lab file-review start
```

> **NOTE**
>
> All files used in this exercise are available on `workstation` in the `/home/student/file-review/files` directory.

Procedure 5.3. Instructions

1. Review the `inventory` file in the `/home/student/file-review` directory. This inventory file defines the `servers` group, which has the `serverb.lab.example.com` managed host associated with it.

   1.1. On `workstation`, change to the `/home/student/file-review` directory.

   ```
   [student@workstation ~]$ cd ~/file-review/
   ```

   1.2. Display the content of the `inventory` file.

   ```
   [servers]
   serverb.lab.example.com
   ```

   HIDE SOLUTION

2. Identify the facts on `serverb.lab.example.com` that show the total amount of system memory, and the number of processors.

Use the `setup` module to get a list of all the facts for the `serverb.lab.example.com` managed host. The `ansible_processor_count` and `ansible_memtotal_mb` facts provide information about the resource limits of the managed host.

```
[student@workstation file-review]$ ansible serverb.lab.example.com -m setup
serverb.lab.example.com | SUCCESS => {
    "ansible_facts": {
...output omitted...
        "ansible_processor_count": 1,
...output omitted...
        "ansible_memtotal_mb": 821,
...output omitted...
    },
    "changed": false
}
```

HIDE SOLUTION

3. Create a template for the Message of the Day, named `motd.j2`, in the current directory. When the `devops` user logs in to `serverb.lab.example.com`, a message should display that shows the system's total memory and processor count. Use the `ansible_facts['memtotal_mb']` and `ansible_facts['processor_count']` facts to provide the system resource information for the message.

```
System total memory: {{ ansible_facts['memtotal_mb'] }} MiB.
System processor count: {{ ansible_facts['processor_count'] }}
```

HIDE SOLUTION

4. Create a new playbook file called `motd.yml` in the current directory. Using the `template` module, configure the `motd.j2` Jinja2 template file previously created to map to the file `/etc/motd` on the managed hosts. This file has the `root` user as owner and group, and its permissions are 0644. Using the `stat` and `debug` modules, create tasks to verify that `/etc/motd` exists on the managed hosts and displays the file information for `/etc/motd`. Use the `copy` module to place `files/issue` into the `/etc/` directory on the managed host, use the same ownership and permissions as `/etc/motd`. Use the `file` module to ensure that `/etc/issue.net` is a symbolic link to `/etc/issue` on the managed host. Configure the playbook so that it uses the `devops` user, and sets the `become` parameter to `true`.

```
---
- name: Configure system
  hosts: all
  remote_user: devops
  become: true
  tasks:
    - name: Configure a custom /etc/motd
      template:
        src: motd.j2
        dest: /etc/motd
        owner: root
        group: root
        mode: 0644

    - name: Check file exists
      stat:
        path: /etc/motd
      register: motd

    - name: Display stat results
      debug:
        var: motd

    - name: Copy custom /etc/issue file
      copy:
        src: files/issue
        dest: /etc/issue
        owner: root
        group: root
        mode: 0644

    - name: Ensure /etc/issue.net is a symlink to /etc/issue
      file:
        src: /etc/issue
        dest: /etc/issue.net
        state: link
        owner: root
        group: root
        force: yes
```

HIDE SOLUTION

5. Run the playbook included in the `motd.yml` file.

5.1. Before you run the playbook, use the `ansible-playbook --syntax-check` command to verify its syntax. If it
     reports any errors, correct them before moving to the next step. You should see output similar to the following:

```
[student@workstation file-review]$ ansible-playbook --syntax-check motd.yml

playbook: motd.yml
```

5.2. Run the playbook included in the `motd.yml` file.

```
[student@workstation file-review]$ ansible-playbook motd.yml

PLAY [Configure system] *********************************************

TASK [Gathering Facts] *********************************************
ok: [serverb.lab.example.com]

TASK [Configure a custom /etc/motd] *********************************
changed: [serverb.lab.example.com]

TASK [Check file exists] *******************************************
ok: [serverb.lab.example.com]

TASK [Display stat results] ****************************************
ok: [serverb.lab.example.com] => {
    "motd": {
        "changed": false,
        "failed": false,
...output omitted...

TASK [Copy custom /etc/issue file] *********************************
changed: [serverb.lab.example.com]

TASK [Ensure /etc/issue.net is a symlink to /etc/issue] ************
changed: [serverb.lab.example.com]

PLAY RECAP *********************************************************
serverb.lab.example.com    : ok=6    changed=3    unreachable=0    failed=0
```

<div style="border:1px solid green;display:inline-block;padding:8px">**HIDE SOLUTION**</div>

6. Check that the playbook included in the `motd.yml` file has been executed correctly.

Log in to `serverb.lab.example.com` as the `devops` user, and verify that the `/etc/motd` and `/etc/issue` contents are displayed when logging in. Log off when you have finished.

```
[student@workstation file-review]$ ssh devops@serverb.lab.example.com
*----------------------------- PRIVATE SYSTEM -----------------------------*
*   Access to this computer system is restricted to authorised users only.  *
*                                                                           *
*     Customer information is confidential and must not be disclosed.       *
*---------------------------------------------------------------------------*
System total memory: 821 MiB.
System processor count: 1
Activate the web console with: systemctl enable --now cockpit.socket

This system is not registered to Red Hat Insights. See https://cloud.redhat.com/
To register this system, run: insights-client --register

Last login: Thu Apr 25 22:09:33 2019 from 172.25.250.9
[devops@serverb ~]$ logout
```

<div style="border:1px solid green;display:inline-block;padding:8px">**HIDE SOLUTION**</div>

Evaluation

On `workstation`, run the `lab file-review grade` script to confirm success on this exercise.

```
[student@workstation ~]$ lab file-review grade
```

Finish

On `workstation`, run the `lab file-review finish` script to clean up after the lab.

```
[student@workstation ~]$ lab file-review finish
```

This concludes the guided exercise.

Privacy Policy (http://s.bl-1.com/h/cZrgWbQn?url=https://www.redhat.com/en/about/privacy-policy?extIdCarryOver=true&sc_cid=701f2000001D8QoAAK)

Red Hat Training Policies (http://s.bl-1.com/h/cZrb2DXG?url=https://www.redhat.com/en/about/red-hat-training-policies)

Terms of Use (https://www.redhat.com/en/about/terms-use)

All policies and guidelines (https://www.redhat.com/en/about/all-policies-guidelines)

Release Notes (https://learn.redhat.com/t5/Red-Hat-Learning-Subscription/Red-Hat-Learning-Subscription-Release-Notes/ba-p/22952)

Cookie Preferences