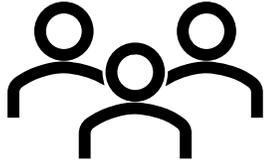


Linux RedHat Certified Engineer (RHCE - EX294)

Automate Linux Administration Tasks

User Account Management



- The playbook will
 - Create a user **george** on remote clients
 - The user **george** will have a home directory as **/home/George**
 - The shell environment for user **george** will be **/bin/bash**

```
# vim adduser.yml
```

```
---
```

```
- name: Playbook for creating users
```

→ Description of the playbook

```
hosts: all
```

→ Run on all clients

```
tasks:
```

→ Run the following task

```
- name: Create users
```

→ Name of the task

```
user:
```

→ Run **user** module for account management

```
name: george
```

→ Username

```
home: /home/george
```

→ User home directory

```
shell: /bin/bash
```

→ User shell

Run the playbook

```
# ansible-playbook adduser.yml
```

Add or Update User Password

- The playbook will
 - Add/update a password for user **george**



Please note: Ansible does not allow us to pass a cleartext password through the user **module**

```
# vim changepass.yml
---
- name: Add or update user password
  hosts: all

  tasks:
  - name: Change "george" password
    user:
      name: george
      update_password: always
      password: "{{ newpassword|password_hash('sha512') }}"
```

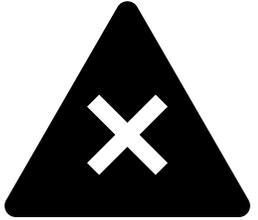
- To run this playbook, run the command as below. This will input the **newpassword** variable that will be used by our playbook

Run the playbook

```
# ansible-playbook changepass.yml --extra-vars newpassword=abc123
```

Kill a Running Process

- The playbook `killprocess.yml` will
 - Find a running process by process name
 - Ignore any errors
 - Hold the result in registry variable
 - Use `shell` module and run `kill` command to kill the registered variable



```
# vim killprocess.yml

---
- name: Find a process and kill it
  hosts: 10.253.1.115

  tasks:
    - name: Get running processes from remote host
      ignore_errors: yes
      shell: "ps -few | grep top | awk '{print $2}'"
      register: running_process

    - name: Kill running processes
      ignore_errors: yes
      shell: "kill {{ item }}"
      with_items: "{{ running_process.stdout_lines }}"
```

Run the playbook

```
# ansible-playbook killprocess.yml
```