Linux User Creation & Sudo Access Using Ansible

Author: Venkata Subrahmanyeswarao
Designation: Data Architect
Email: subrahmanyeswaraokarri@gmail.com

This document provides a complete, step-by-step implementation guide for creating a Linux user and granting sudo access across multiple servers using Ansible automation from a centralized Ansible control node.

1. Prerequisites

• Ansible installed on control server
• Python 3 available on all target hosts
• SSH connectivity to target servers
• touadmin user with sudo privileges
Verify Ansible:

```
ansible --version
```

2. Directory Structure

```
/home/venkata/ansible/
■■■ inventory.ini
■■■ add_user_venkata.yml
```

3. Inventory Configuration

The inventory file lists all target servers under the group **mongo_servers**.

```
[mongo_servers]
192.168.61.165
192.168.61.185
192.168.61.135
...
```

4. Connectivity Test

```
ansible -i inventory.ini mongo_servers -m ping -u touadmin --ask-pass
```

5. Password Hash Generation

```
python3 - <import crypt
print(crypt.crypt("xuv7ooax5", crypt.mksalt(crypt.METHOD_SHA512)))
EOF
```

6. Ansible Playbook

```
---
- name: Create Linux user venkata and grant sudo access
hosts: mongo_servers
become: yes

vars:
username: venkata
user_password: "$6$HASH_REDACTED"

tasks:
- name: Ensure venkata user exists
user:
name: "{{ username }}"
password: "{{ user_password }}"
shell: /bin/bash
create_home: yes
state: present

- name: Add venkata to sudo group
user:
name: "{{ username }}"
groups: sudo
append: yes
```

7. Playbook Execution

```
ansible-playbook -i inventory.ini add_user_venkata.yml -u touadmin --ask-pass --ask-become-pass
```

8. Verification

```
id venkata
groups venkata
```

9. Idempotency

The playbook is idempotent. Re-running it will not create duplicate users or reapply changes unnecessarily.

Regards,
Venkata Subrahmanyeswarao
Data Architect
Email: subrahmanyeswaraokarri@gmail.com