sdnog users creation using Ansible

Ansible Playbook: Sudo Users creation with SSH Keys

> This page was written by Manhal Mohamed, sdnog team, on 11 August 2024.

This Ansible playbook configures users with sudo privileges, sets up SSH keys, and requires users to change their password upon first login.

Overview

The playbook performs the following tasks:

1. Installs necessary packages based on the operating system (Debian/Ubuntu or RedHat/CentOS).
2. Checks if users already exist.
3. Generates passwords for new users.
4. Creates new users with these passwords and assigns them to the sudo group.
5. Sets passwords to expire upon the user's first login.
6. Deploys SSH public keys for users.
7. Sends the password to users via email.

Variables

- **users**: A list of users to be created, each with the following attributes:
  - **username**: The username for the new account.
  - **ssh_key**: The SSH public key to be deployed for the user.
  - **email**: The email address where the password will be sent.

Tasks

Task 1. Install Required Packages

For Debian/Ubuntu

```
  - name: needed packages are installed (Debian/Ubuntu)
    apt:
      name: "{{ item }}"
      state: latest
    loop: ["sudo", "openssh-server", "mailutils"]
```

```
  when: ansible_facts['os_family'] == "Debian"
```

For RedHat/CentOS

```
- name: needed packages are installed (RedHat/CentOS)
  yum:
    name: "{{ item }}"
    state: latest
  loop: ["sudo", "openssh-server", "mailx"]
  when: ansible_facts['os_family'] == "RedHat"
```

Task 2. Check if Users Exist and Set Facts for New Users

```
- name: Check if users exist and set fact for new users
  command: "getent passwd {{ item.username }}"
  register: user_check
  loop: "{{ users }}"
  changed_when: false
  failed_when: false
```

Task 3. Generate Passwords for New Users

```
- name: Generate passwords for new users
  set_fact:
    user_passwords: "{{ user_passwords | default({}) | combine({item.item.username: lookup('pa
  loop: "{{ user_check.results }}"
  when: item.stdout == ""
```

Task 4. Create New Users with Plain-Text Passwords

```
- name: Create new users with plain-text passwords if they do not exist
  user:
    name: "{{ item.username }}"
    password: "{{ user_passwords[item.username] | default('') | password_hash('sha512') }}"
    groups: sudo
    append: yes
    create_home: yes
    shell: /bin/bash
    update_password: on_create
    expires: -1
  loop: "{{ users }}"
  when: item.username in user_passwords
```

Task 5. Set Password to Expire Upon First Login

```
- name: Set password to expire upon first login for newly created users
  command: chage -d 0 "{{ item.username }}"
  loop: "{{ users }}"
```

```
    when: item.username in user_passwords
```

Task 6. Deploy SSH Public Keys for the Users

```
  - name: Deploy SSH public keys for the users
    authorized_key:
      user: "{{ item.username }}"
      state: present
      key: "{{ item.ssh_key }}"
    loop: "{{ users }}"
```

Task 7. Send Password to Users via Email

```
  - name: Send password to users via email
    mail:
      host: relay.example.com
      port: 25
      to: "{{ item.email }}"
      subject: "Your new account password"
      body: |
        Dear {{ item.username }},

        Your new account has been created on the following host: {{ ansible_host }}.

        Username: {{ item.username }}
        Password: {{ user_passwords[item.username] }}

        Please change your password upon first login.
        **Note:** This is an automated message generated by Ansible. Please do not reply to this



        Best regards,
        Sdnog Team

      from: sdnog-ansible-at-email.com
    loop: "{{ users }}"
    when: item.username in user_passwords
```

Appendix : The Full Code

```
  - name: Configure sudo users with SSH keys and require password change on first login
    hosts: host-ip-address
    become: true
    vars:
      users:
        - username: sdnog-user
          ssh_key: "ssh-ed25519 some SSH KEY here eddsa-key-20240807"
          email: "email-at-example.com"
```

```
tasks:
  - name: needed packages are installed (Debian/Ubuntu)
    apt:
      name: "{{ item }}"
      state: latest
    loop: ["sudo", "openssh-server", "mailutils"]
    when: ansible_facts['os_family'] == "Debian"

  - name: needed packages are installed (RedHat/CentOS)
    yum:
      name: "{{ item }}"
      state: latest
    loop: ["sudo", "openssh-server", "mailx"]
    when: ansible_facts['os_family'] == "RedHat"

  - name: Check if users exist and set fact for new users
    command: "getent passwd {{ item.username }}"
    register: user_check
    loop: "{{ users }}"
    changed_when: false
    failed_when: false

  - name: Generate passwords for new users
    set_fact:
      user_passwords: "{{ user_passwords | default({}) | combine({item.item.username: lookup
    loop: "{{ user_check.results }}"
    when: item.stdout == ""

  - name: Create new users with plain-text passwords if they do not exist
    user:
      name: "{{ item.username }}"
      password: "{{ user_passwords[item.username] | default('') | password_hash('sha512') }}
      groups: sudo
      append: yes
      create_home: yes
      shell: /bin/bash
      update_password: on_create
      expires: -1
    loop: "{{ users }}"
    when: item.username in user_passwords

  - name: Set password to expire upon first login for newly created users
    command: chage -d 0 "{{ item.username }}"
    loop: "{{ users }}"
    when: item.username in user_passwords

  - name: Deploy SSH public keys for the users
    authorized_key:
      user: "{{ item.username }}"
      state: present
      key: "{{ item.ssh_key }}"
    loop: "{{ users }}"
```

```
    - name: Send password to users via email
      mail:
        host: relay.example.com
        port: 25
        to: "{{ item.email }}"
        subject: "Your new account password"
        body: |
          Dear {{ item.username }},

          Your new account has been created on the following host: {{ ansible_host }}.
          Username: {{ item.username }}
          Password: {{ user_passwords[item.username] }}

          Please change your password upon first login.

          **Note:** This is an automated message generated by Ansible. Please do not reply to

          Best regards,
          Sdnog Team

        from: sdnog-ansible-at-example.com
      loop: "{{ users }}"
      when: item.username in user_passwords
```

Revision #4
Created 30 November 2024 10:44:11 by sara
Updated 17 May 2025 17:12:59 by sara