Date: 30.07.2025

Salt-API Integration for Automating state application with Jenkins

Contents

1. Launch the Jenkins Instance via Ansible

Here is the Git repo containing the ansible-playbook: aryan-madhavi/ansible-saltstack
You may only require the `hosts` and `launch-jenkins-instance.yml` files.

The task of EC2 provisioning can be automated with `ansible-playbook`. Ensure you have `amazon.aws` collection installed with `ansible-galaxy collection list`. If not, run `ansible-galaxy collection install amazon.aws`.

Also ensure you have `aws cli` preconfigured before running the playbooks. If not, create access key from AWS console and run `aws configure` on the terminal.

Launch instance by running the following command. Use the `-v` flag if required:

```
$  ansible-playbook -i hosts launch-jenkins-instance.yml
```

2. Git Repository for the Jenkins Automation Tasks

This repository contains all the salt state files used for salt operations: aryan-madhavi/saltstack
These files are to be stored in `/srv/salt`.

You may require to configure `/etc/salt/master` to include `fileserver.conf` if you wish to not clone the git repo into `/srv/salt`

/etc/salt/master.d/fileserver.conf

```
fileserver_backend:
  - git
gitfs_remotes:
  - https://github.com/aryan-madhavi/saltstack-demo.git
gitfs_base: master
```

3. Ensure salt-api is running

If the salt-master was installed with the provided playbook within aforementioned git repository, then the salt-api is already pre-installed and configured for use.

If not, add the following `.conf` files into `/etc/salt/master.d/`

/etc/salt/master.d/auth.conf

```
external_auth:
  pam:
    saltuser:
      - .*
      - '@wheel'
      - '@runner'
      - '@jobs'
```

/etc/salt/master.d/api.conf

```
rest_cherrypy:
  port: 8080
  host: 0.0.0.0
  debug: true
  disable_ssl: true
```

/etc/salt/master.d/logs.conf

```
log_level: info
```

/etc/salt/master.d/clients.conf

```
netapi_enable_clients:
  - local
  - local_async
  - local_batch
  - local_subset
  - runner
  - runner_async
```

Also create a system user `saltuser` and set password as `passwd`

```
  $  useradd -m -s /bin/bash saltuser
  $  echo "saltuser:passwd" | chpasswd
```

Also ensure `salt-master` is running as root by commenting/removing `user: salt` from `/etc/salt/master`. Otherwise PAM Authentication with `salt-api` will fail.

Once all configuration is done restart salt-api and salt-master with:

```
  $  systemctl restart salt-api salt-master
```

4. Test salt-api locally

Using salt

```
  $  salt -a pam ‘*’ test.ping
```

Using curl

```
  $  curl localhost:8080/login \
        -c ~/cookies.txt \
        -H ‘Accept: application/json’ \
        -H ‘Content-Type: application/json’ \
        -d ‘{
          “username”: “saltuser”,
          “password”: “passwd”,
          “eauth”: “pam”,
        }’ | jq
```

Using curl test a function

```
  $  curl localhost:8080 \
        -b ~/cookies.txt \
        -H ‘Accept: application/json’ \
        -H ‘Content-Type: application/json’ \
        -d ‘{
          “client”: “local”,
          “tgt”: “*”,
          “fun”: “test.ping”,
        }’ | jq
```

5. Set up Jenkins to handle Salt requests

Open the Jenkins Dashboard, login with initialAdminPassword if first time login.
Install all suggested plugins. Then create a jenkins admin user.

Now, go to Jenkins Dashboard → Manage Jenkins → Install Plugins.
Search for SaltStack and install that plugin. Once installed restart the jenkins service.

Re-login with the previously created admin user.

Go to Jenkins Dashboard → Manage Jenkins → Credentials → Global Credentials
Create a credential for salt-api PAM eauth with id as `saltuser-creds`.
Set the username as `saltuser` and password as `passwd`.

Now, create a multibranch pipeline that uses the aforementioned aryan-madhavi/saltstack repository as Git Source. This repository also contains the Jenkinsfile for the build steps. This Jenkinsfile applies the `nginx` state to all minions. Also ensure that your credentialsId and servername are correct.

Once the build is successful, the build log will contain the output.