Manage secrets with Ansible- Vault

We will discuss the procedure/steps to secure your secrets(passwords/keys/certs files) of configuration/deployments by automating with ansible.

ansible-vault is command line tool we use in ansible to encrypt information

1. Create a file called secrets.yml

vim secrets.yml

2. Add the entries (ssh keys/password variables)in the secrets.yml file.

3. To show the entries in the file

cat secrets.yml

4. Encrypt existing file

ansible-vault encrypt secrets.yml

It will ask prompt for password first time

```
output:
New vault password:
Confirm New Vault Password:
```

to create a file with vault encrypted

```
ansible-vault create secrets.yml
```

To edit vault encrypted file — to change sensitive information

```
ansible-vault edit secrets.yml
```

To encrypt an existing file (test.yml)which contains sensitive information

```
ansible-vault encrypt test.yml
```

The entries inside the file will be encrypted as ANSCII format as shown in the below output

cat test.yml

```
Output
$ANSIBLE_VAULT;1.1;AES256
13030636134643235316439336133363531303838376235376635373430336333
303362333236643064346263636437316265366438333366383566613963643136663666231616261
3764656365313263620a383666383233626665376364323062393462373266663066
31643731343666353761633563633634326139396230313734333034653238303166
```

Read the password automatically by storing in a file and referring from ansible.cfg

first create a vault password file and add your password in the file.

```
cat ‘vaultpassword’ > .vault_pass
```

change the mode of the file to 755

```
chmod 755 .vault_pass
```

add the path of the .vault_pass file in the ansible.cfg file under the settings vault_password_file=/path/.vault_pass

Note: Make sure you should not commit the .vault_pass file to version control by adding to gitignore file

```
echo ‘.vault_pass’ >> .gitignore
```

Setting up secretes/sensitive information in a variable using vault

Organise your variables in a defined detailed structure.

In the example below, we store the password of the splunk forwarder in a variable stored inside a vault encrypted file . we refer to the variables in the playbook to get the password details as part of initial configuration.Follow the below steps as mentioned.

1. Please create directory splunk under the group_vars directoy

```
mkdir group_vars/splunk/
```

2. we create two files 1-variables.yml and 2- vault.yml file

add the non-sensitive information as variables in the normal file variables.yml

```
vim varibales.ymluser: adminport: 9997splunkpasswd: “{{ vault_splunkpasswd }}”
```

and for the sensitive information(splunk password)create a second file vault.yml using vault

```
ansible-vault create vault.yml
```

and in the vault.yml file add the entry as below

```
vault_splunkpasswd: password
```

In the vault file we prefix the variables with “vault_ “ and now in normal file we refer those variables of vault file with an another variable, which gives us the opportunity to encrypt our secrets and at the same time we can refer those variables in readable format.

Retrive the password using variables encrypted by vault in the playbook

```
--- - name: get the variables   include_vars: /group_vars/splunk/variables.yml - name: change the splunk user passwd   shell: /path/splun
```

you can also refer the variable in the templates.