**Ensure that when productionizing automation code we use secure methods for storing credentials.**

### Defining the client’s input: Credentials should always store in a secure location and best practice while using Ansible Vault is to encrypt only the sensitive data and leave other non-sensitive-data in plain text inventory files.The vault file contains the settings, which will be used on the OneView appliance connection, like hostname, username, and password. Here's an example:

### Assuming vault.yml is encrypted the file can be defined as follows

### "oneview\_ip": "172.25.105.12"

### "oneview\_username": "Administrator"

### "oneview\_password": "secret123"

### "api\_version": 200

### The bigger concern is that the examples don’t show how to use OneView modules with vaulted configuration which is simple as follow:

### name: Create a server profile template

### oneview\_server\_profile\_template:

### hostname: "{{ oneview\_ip }}"

### username: "{{ oneview\_username }}"

### password: "{{ oneview\_password }}"

### api\_version: "{{ api\_version }}"

### state: present

### data:

### name: "{{ server\_profile\_template\_name }}"

### serverProfileName: "{{ server\_profile\_name }}" # Optional - Server Profile to base this SPT on

### serverHardwareTypeName: "{{ server\_hardware\_type\_name }}"

### enclosureGroupName: "{{ enclosure\_group\_name }}"

### params:

### force: "True" # Supprted only for API version >= 600

### delegate\_to: localhost

### Design: We can use Ansible Vault feature provide by Ansible.

### Ansible Vault is a feature of ansible that allows you to keep sensitive data such as password or keys in encrypted files, rather than as plaintext in playbooks or roles.

### Algorithm Used to Encrypt Files: (AES256) identifies the cipher algorithm used to encrypt the data. Currently, the only supported cipher is ‘AES256’. [vault format 1.0 used ‘AES’, but current code always uses ‘AES256’] We have commands to encrypt, view and decrypt our data in the files.

### Create: Assume you want to create a new file and you want to encrypt the data. Then you can use ansible vault create.

### View: command is to view data in encrypt file. Ansible vault view.

### Edit: command is to edit data in encrypt file. Ansible Vault Edit.

### Encrypt: Encrypts any existing unencrypted file.

### Decrypt: Convert an encrypted format file to normal file.

### If we want to use this encrypted format file while running our playbook, we need to pass the password. To get the prompt you will add "--ask-vault-pass" to get the password prompt.

### If you want to pass the password through a file, we have "--vault-password-file"

### Implementation: Below we will go through the steps to implement Ansible Vault in our existing OneView Ansible SDK Setup.

### Create a OneView credential file while will have OneView credentials in encrypted format

$ ansible-vault create ov\_credential.yml

New Vault Password:

* File will get encrypted while you save it.

1. To view the encrypted file.

$ ansible-vault view ov\_credential.yml

Vault Password:

*hostname: “10.50.8.80”*

*username: “Administrator”*

*password: “Password”*

*api\_version: 1200*

1. Setting your ansible playbook to use the credentials from encrypted ov\_credential.yml file.

$ cat oneview\_ethernet\_network\_facts.yml

----

* hosts: all

vars\_files:

- ov\_credential.yml

tasks:

-name: Gather facts about all ethernet networks

oneview\_ethernet\_network\_facts:

hostname: “{{ *hostname* }}”

username: “{{ *username* }}”

password: “{{ *password* }}”

api\_version: “{{ *api\_version* }}”

-delegate\_to: localhost

1. To run the OneView ansible playbook

$ ansible-playbook oneview\_ethernet\_network\_facts.yml –ask-vault-pass -vvv

Vault Password:

Resources: <https://docs.ansible.com/ansible/latest/user_guide/vault.html>