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.

1. 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.

2. 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
```

3. 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
```

4. 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