**Ansible and SNOW integration**

**Ansible Installation**:

#1. Run the below command from root user:

sudo -i

rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

#2. Install Ansible: yum install -y ansible

yum install -y ansible

#Check the ansible version

echo" Output:

ansible --version

#Ansible installed at the location --> /etc/Ansible

**Python3 installation in red hat**:

<https://developers.redhat.com/blog/2018/08/13/install-python3-rhel/>

pynow module is required to connect the SNOW instance link and execute the below steps.

**pynow module installation:**

pip install --upgrade pip

yum install pip

pip install pysnow

python –version

alias python="/usr/bin/python3.4" # to change the default version of python.

**Alternative method to install python3 and PIP if the above method is not working**:

**sudo yum install python-pip**

**rpm -ivh** [**https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm**](https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm)

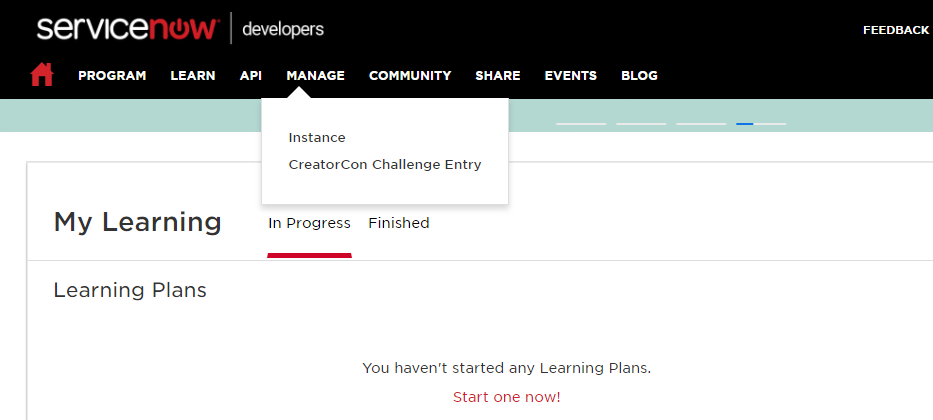
**yum install -y python34.x86\_64**

**Creating SNOW instance**: Follow the below procedures.

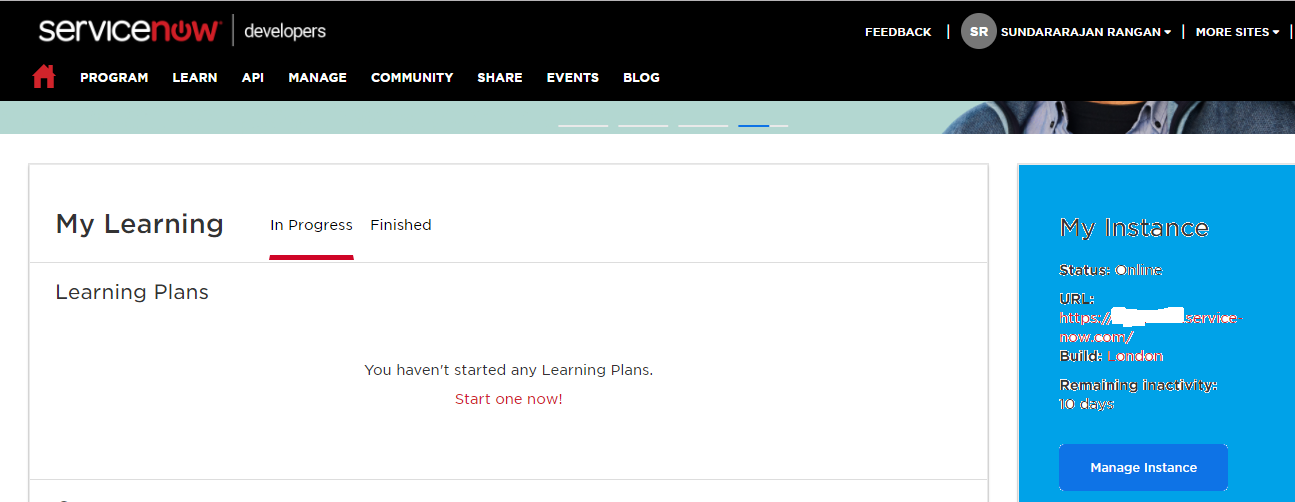
<https://developer.servicenow.com>

first time login users can see below picture to create a snow instance:

Click the manage menu and select the instance below: It will give you a developer snow instance for you to manage the portal:



if the snow instance is already created you can see the below picture:



We will get the unique server instance to access the SNOW.

Ansible playbook:

**Step 1**: Before you create a Ansible playbook check the python version.

**Step 2**: SNOW module will work only in python version 3 and above.

**Step 3**: Provide the alias name for the latest installed python like below.

alias python="/usr/bin/python3.4"

**Step 4**: Run python in the command VM command prompt and check if the latest version is being displayed.

Command prompt: python

**Step 5**: Create a playbook like below. [ex: vi create\_change.yml] The file extension should be .yml ( YAML file)

---

- hosts: localhost

gather\_facts: False

tasks:

- name: Create a ticket

snow\_record:

username: admin

password: <<your\_password>>

instance: <<your\_instance\_name\_only and not full link>>>>

state: present

table: change\_request

data:

short\_description: "This is the test ticket from ansible"

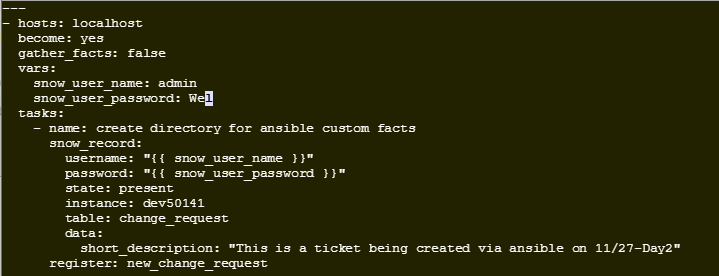
register: new\_change\_request

**Step 5**: Execute the playbook.

ansible-playbook create\_change.yml

**Step 6**: Check the developer SNOW instance the change would be created at the top.

Passing variable in our playbook and see the below example:



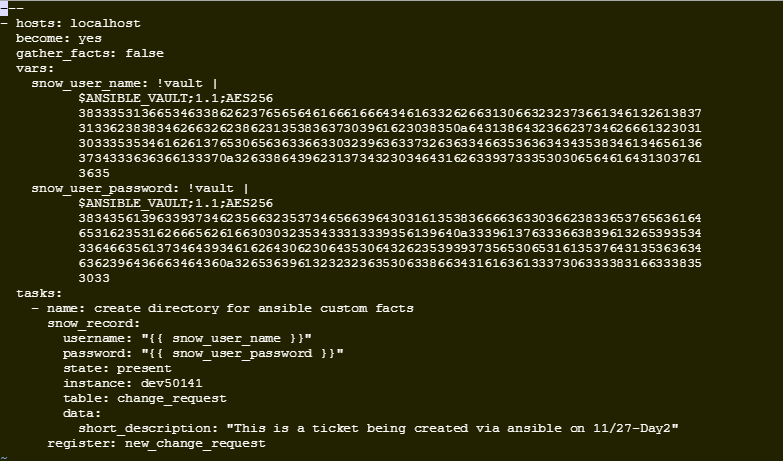
Using ansible-vault to encrypt the sensitive details in the playbook:

Follow is the example to encrypt the particular variable of string:

ansible-vault encrypt\_string admin --ask-vault-pass

ansible-vault encrypt\_string password --ask-vault-pass

Once you encrypted the fields and you have to copy the contents into your playbook as below:



Run the playbook using –ask-vault-pass as below:

ansible-playbook test\_var.yml --ask-vault-pass