ANSIBLE ROLES AND VARIABLES

Pre-setup:

Create your own directory (i.e we have created ansible dir in this practice) and copy the ansible.cfg, hosts, roles from default /etc/ansible/ (which is automatically created when installing ansible package).

In ansible.cfg mention the path of role and host

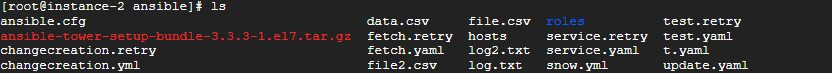
Inventory = /your\_ansible\_directory/hosts

roles\_path = /your\_ansible\_directory/roles

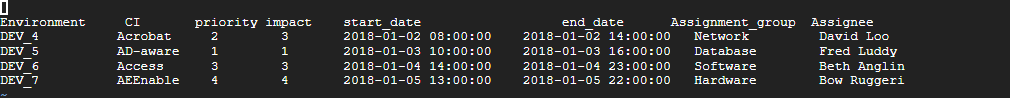
you can create any directory anywhere in the machine, but copy all the files under /etc/ansible to your directory and do the above changes in your ansible.cfg.

**Hosts file – is an inventory file which has the target servers going to affect in running playbook.**

**In this case we have only localhost where our playbook is dealing with API.**



Keep your file.csv in ansible directory with below format (headers).



You can neither use single playbook or roles to complete the tasks. When its going to production we have to use roles for more security, reusable and non-complexity. Here we have created a role called servicenow.

Roles:

The first step in creating a role is creating its directory structure. To create the base directory structure, going to use a tool bundled with Ansible called ansible-galaxy.

Go to roles directory and create role “servicenow”.



ansible-galaxy init servicenow

That command will give servicenow directory with the following structure.

├── README.md

├── defaults

│ └── main.yml

├── files

├── handlers

│ └── main.yml

├── meta

│ └── main.yml

├── tasks

│ └── main.yml

├── templates

└── vars

\_\_main.yml

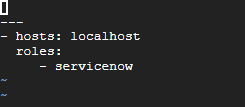
Variables:

So place your playbook under tasks and place your sensitive variables to variables directory.

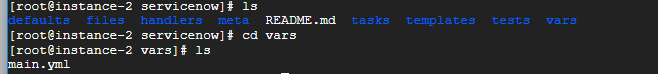
It can be re-usable and can separated from the playbook.

1. Go to the your ansible home directory and create yaml file which calls the role.





2. And then go to variable section and open main.yml for adding variables



Open main.yml under variable section and then add your snow credentials and instance details

**---**

**snow\_username : admin**

**Snow\_password : XXXXX**

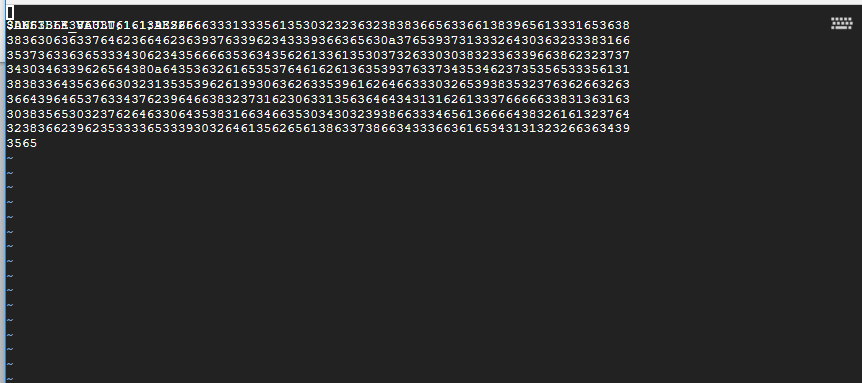
**Snow\_instance : dev71301**

Using Ansible-Valut Encrypting unencrypted files:

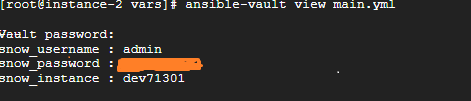
Please use the below command to encrypt under var directory which has main.yml

ansible-vault encrypt main.yml

Run the above command and set the vault password as per your request. Use the same passwd to view, edit, decrypt the same file.Now you can see it is encrypted below and the credentials wont be visible.

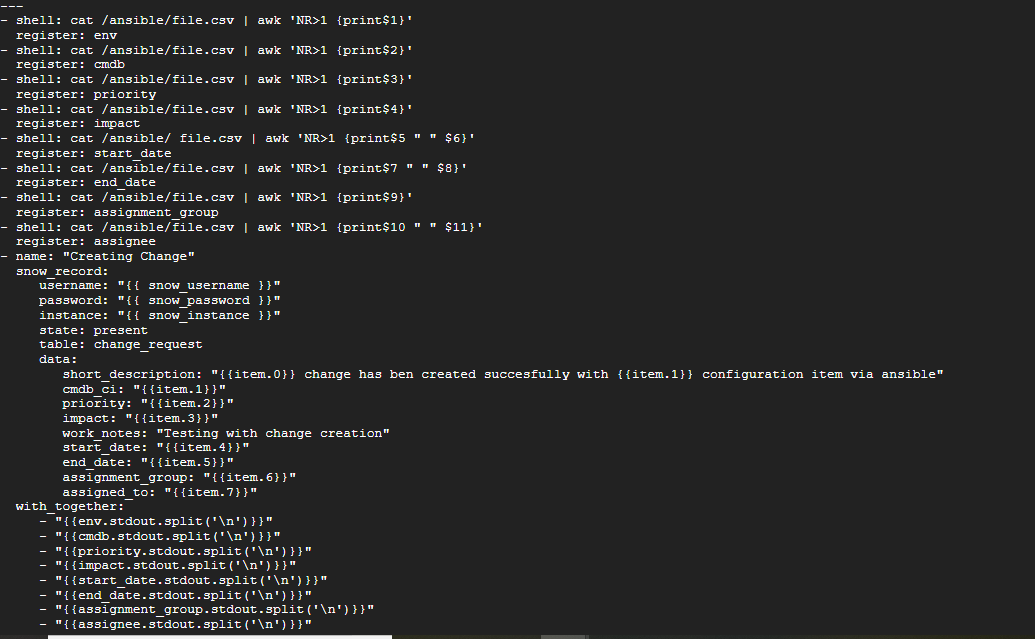


If you want to view the sensitive variable which mention in vault, please use below command and enter the passwd which is created above.

****

And then go to tasks directory and write our playbook in main.yml





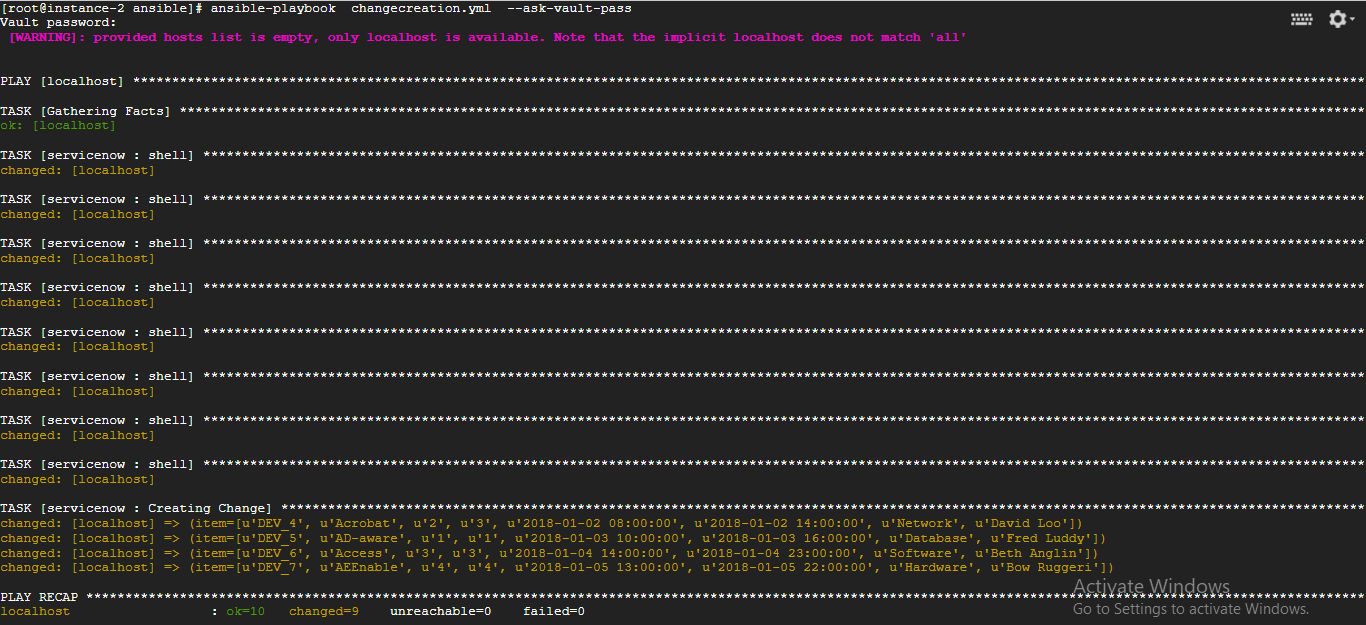
Run the playbook:

**ansible-playbook changecreation.yml --ask-vault-pass**

It will ask the password, please provide the below password

Password: ansible99

**Ex ouput:**



**4 changes created in snow**

