

Nornir

Nornir är ett Python-bibliotek som är gjort för att utföra förändringar på många nätverksenheter snabbt.

Går bra att använda när Ansible blir för långsamt.

- [Ansible-vault och Nornir](#)

Ansible-vault och Nornir

Det går att kombinera hemliga variabler med hjälp av Ansible vault i ett Nornir-jobb.

De synliga, icke hemliga variablarna placerar man som vanligt i `inventory/hosts.yaml` och `inventory/groups.yaml`. Hemliga variabler kan man sedan placera i `inventory/secrets.yaml`.

Man kommer sedan att kombinera dessa när jobbet körs, så det är viktigt att hierarkin är exakt lika i samtliga filer.

Har man exempelvis följande i `hosts.yaml`:

```
nxos:
  data:
    routed_underlay:
      ISIS_NET: 49.1337.0000.0000.0001.00
```

Och ett lösenord ska användas för ISIS som anropas av Python-jobbet på samma ställe ska då `secrets.yaml` se ut såhär:

```
nxos:
  data:
    routed_underlay:
      ISIS_PASSWORD: s0000pers3333cret
```

För att sedan använda sig av ansible vault behövs följande:

Kryptera filen med `ansible-vault encrypt secrets.yaml`. Redigering med `edit` eller `decrypt` (glöm inte `encrypt` efteråt).

Skapa en lokal fil på servern där krypteringslösenordet finns, använd `chmod 400` på den.

Följande moduler behövs:

```
from ansible.parsing.vault import VaultLib, VaultSecret
from ansible.constants import DEFAULT_VAULT_ID_MATCH
from ansible.parsing.dataloader import DataLoader
```

Anropa sedan i jobbet:

```
def load_vault_password(path=~/.ansible_losen"):
    path = Path(path).expanduser()
    return path.read_bytes().strip()
```

I `def main()` behövs följande för att låsa upp filen och använda den:

```
vault_password = load_vault_password()

loader = DataLoader()
vault = VaultLib([(DEFAULT_VAULT_ID_MATCH, VaultSecret(vault_password))])
loader.set_vault_secrets(vault.secrets)

data = loader.load_from_file("inventory/secrets.yaml")

for host, host_data in data.items():
    if host in nr.inventory.hosts:
        nr.inventory.hosts[host].data.update(host_data)
```