

# Ansible Tutorial | NIC.CZ

Vypracoval: Matěj Kevin Nechodom

---

## Basics

### ClusterSSH

- napsané v PERL
- nejde verzovat
- 4 terminály = 4 servery; příkazy se píší do jednoho terminálu a paralelně se odesílají na ostatní servery

### Vlastní balíčky

- IsMedia posinst

### CFEngine

- zastaralé
- napsané v C
- low-level

### Puppet, Chef, Salt

- pull/push model
- server obsahuje konfiguraci, kterou si klienti stahují
- podle intervalu (defaultně 30 min) se konfigurace refreshuje
- Puppet a Chef jsou napsané v Ruby, což znamená, že každý klient potřebuje Ruby interpret

### Ansible

- YAML => Python (2.7/3.5+) => SSH => Python (2.4)

## Instalace

```
apt install python3-venv python3-pip
python3 -m venv .muj_venv
source .muj_venv/bin/activate
pip3 install ansible
```

```
ansible-inventory --list
```

- vypíše seznam všech strojů a jejich konfiguraci

```
ansible 1-ubuntu -m 'command'
```

- posílá příkaz na různé stroje

```
ansible "vm:!1_ubuntu" -m 'command'
```

- posílá příkaz na různé stroje

## Playbooky

```
ansible-playbook play-1.yml
```

- spustí playbook play-1.yml

```
ansible hosts -m setup > /dev/null
```

- vypíše všechny informace o všech strojích (lze použít i pro jeden stroj)

```
ansible-playbook play-3.yml --check
```

- pouze ověřuje konfiguraci bez aplikace změn

```
ansible-playbook play-3.yml --check --limit server
```

- kontrola konfigurace na vybraných serverech

```
ansible-playbook play-3.yml --check --diff
```

- zobrazí rozdíly v konfiguraci

```
check_mode: no
```

- vypne `check_mode`

**Poznámka:** Použití shellu není ideální, protože neumožňuje ověřit úspěšné dokončení operace. Raději používej moduly Ansible.

## Handlery

- Handler se spustí pouze v případě změny

```
- name: Install webserver
  hosts: 1-ubuntu
  handlers:
    - name: Reload apache
      systemd:
        name: "apache2"
        state: "reloaded"
```

- Volání handleru se provádí pomocí `notify: nazev`

## Tagy

- Možnost přeskakovat a filtrovat kroky v playbooku

```
- name: Set hostname on homepage 2
  template:
    src: ./index.html.j2
    dest: /var/www/html/index8.html
  notify: Reload apache
  tags:
    - configure
    - index8
```

## Role

- Strukturovaný proces, lepší přehlednost souborů
- `ansible-galaxy role init nazev` vytvoří novou roli

```
roles/
  webserver_apache
    handlers
      main.yml
    tasks
      main.yml
    templates
      index.html.j2
```

- Playbook volající roli `webserver_apache`:

```
- name: Install webserver - Apache2
  hosts: 1-ubuntu
  roles:
```

```
- webserver_apache
#- monitoring_apache
```

- **Ansible Galaxy** nabízí komunitní role s dokumentací: Ansible Galaxy

## Kolekce

- V roce 2020 došlo k rozdělení Ansiblu na komunitní a nekomunitní verze
- Komunitní kolekce jsou dostupné např. zde: Panos Collection

## Pluginy

- Rozšiřují funkcionalitu Ansiblu
- Typy pluginů:
  - **Callback plugin** – zavolán po dokončení playbooku (např. výstup do e-mailu)
  - **Connection plugin** – způsob připojení (`ssh`, `local`, `chroot`, `docker`)
  - **Inventory plugin** – dynamické inventáře (`aws`, `nmap`, `virtualbox`)
  - **Shell plugin** – umožňuje spouštění shell příkazů

## Příbalené nástroje

- **Vault** – bezpečné uchování hesel
- **Galaxy** – sdílení rolí
- **Lint** – statická analýza kódu
- **Inventory** – parser inventáře

## Příkazy

- Spuštění pouze sekce s konkrétním tagem:

```
ansible-playbook playbook.yml --tags nazev_tagu
```

- Přeskočení sekce s tagem:

```
ansible-playbook playbook.yml --skip-tags nazev_tagu
```

- Vytvoření nové role:

```
ansible-galaxy role init nazev
```

- Zobrazení dokumentace:

```
ansible doc
```

- Generování konfigurace:

```
ansible config
```

- Komunitní verze pluginů:

```
ansible-community
```

- SSH příkaz na všechny stroje:

```
ansible-console-all
```

- Šifrování hesel pomocí AES256:

```
ansible-vault edit host_vars/1-ubuntu.yml
```

## Externí nástroje

- **Molecule** – testování ansible playbooků

## Poznámky

- Modul `raw` používá pouze SSH – není nutné mít Python na cílovém serveru
- Některé služby (např. nginx) mají vlastní validátory:

```
validate: nginx -t
```

- **Vagrant** umožňuje testování playbooků ve virtualizovaném prostředí
- Multiline v YAML
- Výpis všech serverů a jejich dat
- ShellCheck – kontrola shell skriptů