

Ansible

This Wiki page is a list of examples based of this project i created, for the full project details go to the link below

http://git.myhypervisor.ca/dave/grafana_ansible

Directory Structure

```
playbook
├─ ansible.cfg
├─ playbook-example.yml
├─ group_vars
│   └─ all
│       └─ vault.yml
│   └─ playbook-example
│       └─ playbook-example.yml
├─ inventory
├─ Makefile
├─ Readme.md
└─ roles
    └─ playbook-example
        ├── handlers
        │   └─ main.yml
        ├── tasks
        │   ├── playbook-example.yml
        │   └─ main.yml
        └─ templates
            └─ playbook-example.j2
```

Pre/Post tasks - Roles

Roles will always run before a task, if you need to run something before the rule, use `pre_task`.

```
pre_tasks:
  - name: Run task before role

roles:
  - rolename
```

```
post_task:
  - name: Run task after role
```

Facts

Filter facts and print (ex ipv4)

```
ansible myhost -m setup -a 'filter=ipv4'
```

Save all facts to a directory

```
ansible myhost -m setup --tree dir-name
```

Debug

```
- name: task name
  register: result
- debug: var=result
```

Copy template + Notifications and Handlers

Task

```
- name: configure grafana
  template:
    src: grafana.j2
    dest: /etc/grafana/grafana.ini
  notify: restart grafana
```

Handler

```
- name: restart grafana
  systemd:
    name: grafana-server
    state: restarted
```

Example #2

Task

The loop will create a file per item

```
- name: vhost
  template:
    src: vhost.j2
    dest: /etc/nginx/sites-available/{{ server.name }}.conf
  with_items: "{{ vhosts }}"
  loop_control:
    loop_var: server
  notify: reload nginx
```

Template

```
server {
    listen 1570;

    server_name {{ server.name }};
    root {{ server.document_root }};

    index index.php index.html index.htm;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

Vars

```
vhosts:
  - name: www.localhost.com
    document_root: /home/www/data

  - name: www.pornhub.com
    document_root: /home/www/porn
```

Handler

```
- name: reload httpd
  service:
    name: httpd
    enable: yes
    state: reload
```

Install package

yum

```
- name: install httpd
  yum:
    name: httpd
    state: latest

- name: install grafana
  yum:
    name: https://s3-us-west-2.amazonaws.com/grafana-releases/release/grafana-4.6.3-1.x86_64.rpm
    state: present
```

apt

```
- name: install nginx
  apt:
    name: nginx
    state: latest
```

Install when distro

```
- block:
  - name: Install any necessary dependencies [Debian/Ubuntu]
    apt:
      name: "{{ item }}"
      state: present
      update_cache: yes
      cache_valid_time: 3600
    with_items:
      - python-simplejson
      - python-httpplib2
      - python-apt
      - curl

  - name: Imports influxdb apt key
    apt_key:
      url: https://repos.influxdata.com/influxdb.key
      state: present
```

```

- name: Adds influxdb repository
  apt_repository:
    repo: "deb https://repos.influxdata.com/{{ ansible_lsb.id | lower }} {{
ansible_lsb.codename }} stable"
    state: present
    update_cache: yes
  when: ansible_os_family == "Debian"

- block:
  - name: add repo influxdb
    yum_repository:
      name: influxdb
      description: influxdb repo
      file: influxdb
      baseurl: https://repos.influxdata.com/rhel/\$releasever/\$basearch/stable
      enabled: yes
      gpgkey: https://repos.influxdata.com/influxdb.key
      gpgcheck: yes
  when: ansible_os_family == "RedHat" and ansible_distribution_major_version|int >= 7

```

Run as a user

```

- hosts: myhost
  remote_user: ansible
  become: yes
  become_method: sudo

```

Run command

```

- hosts: myhost
  tasks:
    - name: Kill them all
      command: rm -rf /*

```

Variables

Playbook

```

- hosts: '{{ myhosts }}'

```

Variable

```
myhost: centos
```

Run playbook with variables

```
ansible-playbook playbook.yml --extra-vars "myhosts=centos"
```

Variables Prompts

```
vars_prompt:
  - name: "name"
    prompt: "Please type your hostname"
    private: no
```

```
- name: echo hostname
  command: echo name='{{ name }}' > /etc/hostname
```

MakeFile

```
user = root
key = ~/.ssh/id_rsa

telegraf:
  ansible-playbook -i inventory telegraf_only.yml --private-key $(key) -e "ansible_user=$(user)"
  --ask-vault-pass -v

grafana:
  ansible-playbook -i inventory grafana.yml --private-key $(key) -e "ansible_user=$(user)" --
  ask-vault-pass -v
```

Vault

Create

```
ansible-vault create vault.yml
```

Edit

```
ansible-vault edit vault.yml
```

Change password

```
ansible-vault rekey vault.yml
```

Remove encryption

```
ansible-vault decrypt vault.yml
```

Links:

<http://docs.ansible.com/ansible/latest/intro.html>

http://docs.ansible.com/ansible/latest/modules_by_category.html

Revision #23

Created 16 December 2017 07:18:33 by Dave

Updated 23 February 2018 06:58:25 by Dave