

01_Playbooks

November 16, 2016

1 YAML Dateien

- Die erste Zeile eines Playbooks sollte mit “—” beginnen (drei Bindestriche). Diese zeigt den Beginn eines YAML-Dokumentes an.
- Listen in YAML werden mit einem Bindestrich “-” gefolgt von einem Leerraum dargestellt.
- Ein Playbook enthält eine Liste von Spielanweisungen; Sie werden mit “-” dargestellt. Jedes Spiel ist ein assoziatives Array, ein Dictionary oder eine Map in Form von Schlüssel/Wert-Paaren.
- Einrückungen sind wichtig. Alle Mitglieder einer Liste sollten gleich eingerückt sein.
- Jede Spielanweisung kann Schlüssel-Wert-Paare enthalten, getrennt durch “:”, um Hosts, Variablen, Rollen, Aufgaben und so weiter anzugeben.

1.1 Weblinks

- <https://de.wikipedia.org/wiki/YAML>
- <http://www.yaml.org/>
- <http://docs.ansible.com/ansible/YAMLSyntax.html>

1.2 Playbooks

simple_playbook.yml

```
---
- hosts: all
  remote_user: vagrant
  become: yes
  tasks:
    - group: name=devops state=present
    - name: create devops user with admin privileges
      user: name=devops comment="Devops User" uid=2001 group=devops
    - name: install http package
      apt: name=http state=present update_cache=yes

- hosts: www
  user: vagrant
  become: yes
  tasks:
```

- name: add official nginx repository
 - apt_repository: repo='ppa:nginx/stable'
- name: install nginx web server and ensure its at the latest version
 - apt: name=nginx state=latest
- name:
 - service: name=nginx state=started

Alternativ

- ```

```
- hosts: all
    - remote\_user: vagrant
    - become: yes
    - tasks:
      - group:
        - name: devops
        - state: present
      - name: create devops user with admin previleges
        - user:
          - name: devops
          - comment: "Devops User"
          - uid: 2001
          - group: devops
      - name: install htop package
        - apt:
          - name: htop
          - state: present
          - update\_cache: yes  
    - hosts: www
      - user: vagrant
      - become: yes
      - tasks:
        - name: add official nginx repository
          - apt\_repository:
            - repo: 'ppa:nginx/stable'
        - name: install nginx web server and ensure its at the latest version
          - apt:
            - name: nginx
            - state: latest
        - name:
          - service:
            - name: nginx
            - state: started

In [1]: cp /vagrant/simple\_playbook.yml ./

```
In [2]: eval `ssh-agent -s` > /dev/null
 ./ssh-add-passphrase.sh
```

```
Enter passphrase for /home/vagrant/.ssh/id_rsa:
Identity added: /home/vagrant/.ssh/id_rsa (/home/vagrant/.ssh/id_rsa)
```

```
In [3]: ansible-playbook simple_playbook.yml
```

```
PLAY [all] *****

TASK [setup] *****
ok: [192.168.60.21]
ok: [192.168.60.11]
ok: [192.168.60.2]
ok: [192.168.60.12]
ok: [192.168.60.22]
ok: [localhost]
ok: [192.168.60.13]

TASK [group] *****
changed: [192.168.60.21]
changed: [192.168.60.2]
changed: [192.168.60.12]
changed: [192.168.60.22]
changed: [192.168.60.11]
changed: [localhost]
changed: [192.168.60.13]

TASK [create devops user with admin previleges] *****
changed: [192.168.60.22]
changed: [192.168.60.11]
changed: [192.168.60.12]
changed: [192.168.60.2]
changed: [192.168.60.21]
changed: [192.168.60.13]
changed: [localhost]

TASK [install htop package] *****
ok: [192.168.60.22]
ok: [192.168.60.2]
ok: [192.168.60.11]
ok: [192.168.60.12]
ok: [192.168.60.21]
ok: [192.168.60.13]
ok: [localhost]
```

```

PLAY [www] *****

TASK [setup] *****
ok: [192.168.60.11]
ok: [192.168.60.13]
ok: [192.168.60.12]

TASK [add official nginx repository] *****
ok: [192.168.60.13]
ok: [192.168.60.12]
ok: [192.168.60.11]

TASK [install nginx web server and ensure its at the latest version] *****
changed: [192.168.60.11]
changed: [192.168.60.12]
changed: [192.168.60.13]

TASK [service] *****
ok: [192.168.60.11]
ok: [192.168.60.13]
ok: [192.168.60.12]

PLAY RECAP *****
192.168.60.11 : ok=8 changed=3 unreachable=0 failed=0
192.168.60.12 : ok=8 changed=3 unreachable=0 failed=0
192.168.60.13 : ok=8 changed=3 unreachable=0 failed=0
192.168.60.2 : ok=4 changed=2 unreachable=0 failed=0
192.168.60.21 : ok=4 changed=2 unreachable=0 failed=0
192.168.60.22 : ok=4 changed=2 unreachable=0 failed=0
localhost : ok=4 changed=2 unreachable=0 failed=0

```

### 1.3 Playbooks

Wichtige Abschnitte der obigen Playbooks sind:

1. Wer soll wie konfiguriert werden (hosts).
2. Was soll ablaufen (tasks).

### 1.4 Pattern für hosts

Im vorherigen Playbook bestimmen die folgenden Zeilen, welche Hosts für einen Play/Spiel ausgewählt werden sollen. Eine bestimmte Spielanweisung für:

- hosts: all
- hosts: www

Der erste Block wird mit allen Hosts ausgeführt. Der zweite Abschnitt/Play wird mit der www-Gruppe durchgeführt.

Die Pattern können eine der folgenden Liste, oder ihre Kombinationen, sein:

|                    |                                      |
|--------------------|--------------------------------------|
| Pattern            | Beispiele                            |
| Gruppenname        | Name_der_Rechner (ansible inventory) |
| Spiel alle         | all oder *                           |
| Range              | Name_des_Rechner[0:100]              |
| Hostnamen globs    | *.example.com, host01.example.com    |
| Ausnahmen          | Name_der_Rechner:!diesen_nicht       |
| Überschneidung     | Name_der_Rechner:&weitere_Namen      |
| Reguläre Ausdrücke | ~(nn zk).*\.example\.org             |

## 1.5 Der Block Tasks

Die Aufgaben für eine Gruppe (hosts). Aufgaben sind eine Folge von Aktionen, die gegen eine Gruppe von Hosts ausgeführt werden. Jedes Play enthält in der Regel mehrere Tasks, die seriell auf jeder Maschine ausgeführt werden, die dem Muster entspricht.

Jede Aktion in einer Aufgabenliste kann deklariert werden, indem Folgendes angegeben wird:

- Der Name des Moduls
- Optional der Zustand der verwalteten Systemkomponente
- Die optionalen Parameter

## 1.6 Module

Module sind die gekapselten Prozeduren, die spezifischen Systemkomponenten für bestimmten Plattformen verwalten z. B. apt, user oder service.

[http://docs.ansible.com/ansible/list\\_of\\_all\\_modules.html](http://docs.ansible.com/ansible/list_of_all_modules.html)

Nichts gefunden? Dann selber schreiben:

[http://docs.ansible.com/ansible/developing\\_modules.html](http://docs.ansible.com/ansible/developing_modules.html)

## 1.7 Module und Idempotence

Aus der Wikipedia (10/2016) "Analog dazu wird in der Informatik ein Stück Programmcode, das mehrfach hintereinander ausgeführt das gleiche Ergebnis wie bei einer einzigen Ausführung liefert, als idempotent bezeichnet."

```
In [4]: ansible-playbook simple_playbook.yml
```

```
PLAY [all] *****

TASK [setup] *****
ok: [192.168.60.12]
ok: [192.168.60.11]
ok: [192.168.60.21]
ok: [192.168.60.2]
ok: [192.168.60.22]
```

ok: [192.168.60.13]  
ok: [localhost]

TASK [group] \*\*\*\*\*  
ok: [192.168.60.2]  
ok: [192.168.60.12]  
ok: [192.168.60.21]  
ok: [192.168.60.22]  
ok: [192.168.60.11]  
ok: [localhost]  
ok: [192.168.60.13]

TASK [create devops user with admin previleges] \*\*\*\*\*  
ok: [192.168.60.21]  
ok: [192.168.60.2]  
ok: [192.168.60.12]  
ok: [192.168.60.22]  
ok: [192.168.60.11]  
ok: [192.168.60.13]  
ok: [localhost]

TASK [install htop package] \*\*\*\*\*  
ok: [192.168.60.21]  
ok: [192.168.60.2]  
ok: [192.168.60.11]  
ok: [192.168.60.12]  
ok: [192.168.60.22]  
ok: [localhost]  
ok: [192.168.60.13]

PLAY [www] \*\*\*\*\*

TASK [setup] \*\*\*\*\*  
ok: [192.168.60.13]  
ok: [192.168.60.11]  
ok: [192.168.60.12]

TASK [add official nginx repository] \*\*\*\*\*  
ok: [192.168.60.11]  
ok: [192.168.60.13]  
ok: [192.168.60.12]

TASK [install nginx web server and ensure its at the latest version] \*\*\*\*\*  
ok: [192.168.60.12]  
ok: [192.168.60.11]  
ok: [192.168.60.13]

TASK [service] \*\*\*\*\*

ok: [192.168.60.11]  
ok: [192.168.60.12]  
ok: [192.168.60.13]

```
PLAY RECAP *****
192.168.60.11 : ok=8 changed=0 unreachable=0 failed=0
192.168.60.12 : ok=8 changed=0 unreachable=0 failed=0
192.168.60.13 : ok=8 changed=0 unreachable=0 failed=0
192.168.60.2 : ok=4 changed=0 unreachable=0 failed=0
192.168.60.21 : ok=4 changed=0 unreachable=0 failed=0
192.168.60.22 : ok=4 changed=0 unreachable=0 failed=0
localhost : ok=4 changed=0 unreachable=0 failed=0
```