



meetup

Manipuler des données avec Ansible

Ansible meetup - Montreal

Pierre Blanc
Ingénieur Logiciel - Red Hat
pierre@redhat.com



meetup

Agenda

Le programme !

- ❖ Données
- ❖ Formats
- ❖ Scope
- ❖ Manipulation
- ❖ IA





Données

Définition

Structure de données qui vous permet de stocker des informations de manière hiérarchique.

```
—→ ansible localhost -m setup
localhost | SUCCESS => {
  "ansible_facts": {
    "ansible_all_ipv4_addresses": [
      "10.27.17.4",
      "192.168.100.1",
      "192.168.122.1",
      "192.168.10.126",
      "192.168.1.132"
    ],
    "ansible_all_ipv6_addresses": [
      "fe80::fc54:ff:fe6b:df26",
      "fe80::d07:d4d5:11d0:d154",
      "fe80::fc54:ff:fe9a:9cec",
      "fe80::d587:4459:9860:e37e",
      "fe80::a42a:cc4b:23c8:591a"
    ]
  }
}
```



meetup

Formats et type de données

Utilisation et conversion

Type de formats:

- ❖ Yaml
- ❖ Json

Type de données:

- Données simples
- Listes
- Tableaux associatifs

Filters:

to_json, to_nice_json, to_yaml, to_nice_yaml, dict2items, int, float, ...



meetup

Données

Générés ou importer

Les plugins **Lookup** pour les importer

```
vars:  
  foo: "{{ lookup('file', 'foo.txt') }}"
```

Création de facts avec **set_fact**

```
set_fact:  
  foo: "{{ my_foo_command.stdout }}"
```



Scope

Où sont mes données

Lorsque vous utilisez `set_fact`, le fact est associé au serveur.

Vous pouvez les récupérer avec `hostvars['monserver']`

Attention aussi à l'utilisation de **delegate** !



meetup

Manipulation

Les filtres ou filters in English

On utilise le pipe | pour les chaîner

Builtin: to_json, basename, flatten, ternary

Utils: Ipaddr, macaddr

Les petits nouveaux: lpcut, ipv6form, gpg



Manipulation

Les filters de jinja2

Plus de 50 filtres additionnels utilisés au quotidien:

float, first, int, join, last, default, sort, length

Le fameux **select**

L'indispensable **map**

L'ineestimable **json_query** (pas de jinja2)



meetup

Manipulation

La documentation

Consulter la documentation, cherchez des exemples dans le code même d'Ansible.

<https://jinja.palletsprojects.com/en/3.1.x/templates/#builtin-filters>

BONUS - BONUS - BONUS - BONUS - BONUS

json_query



meetup

Select/Selectattr

Selection de données

- > select pour une liste simple
- > selectattr pour une liste de dictionnaire

```
numbers=[1,10,43,50]
```

```
numbers2=[{nom:ba,prenom:jo,ville:ny,pays:US}]
```

```
numbers|select("lessthan", 42)
```

```
numbers2|selectattr(ville,'eq','ny')| map(attribute=nom)
```



Exemple

Un premier exemple simple

interfaces:

- ip : 192.168.1.12
mtu: 1500
- ip : 172.16.1.12
mtu: 1450

```
interfaces | selectattr('mtu','gt',1480) | map(attribute='ip')
```



Exemple

Un peu plus complexe

- debug:
msg: "{{ ansible_facts.devices | dict2items |
map(attribute='value.partitions') | combine | dict2items |
selectattr('value.size','gt','900') | map(attribute='key') }}"

```
"devices": {
  "dm-0": {
    "vendor": null,
    "virtual": 1
  },
  "nvme0n1": {
    "holders": [],
    "model": "SAMSUNG MZVL21T0HCLR-00BL7",
    "partitions": {
      "nvme0n1p1": {
        "holders": [],
        "sectors": "1228800",
        "sectorsize": 512,
        "size": "600.00 MB",
        "start": "2048",
        "uuid": "95AE-5385"
      },
      "nvme0n1p2": {
        "holders": [],
        "sectors": "2097152",
        "sectorsize": 512,
        "size": "1.00 GB",
        ...
      }
    }
  }
}
```



Exemple

Un peu plus complexe

- debug:

```
msg: "{{ ansible_facts.devices | dict2items |  
map(attribute='value.partitions') | combine | dict2items |  
selectattr('value.size', 'gt', '900') | map(attribute='key') }}"
```

```
"devices": [
  {
    "key": "dm-0",
    "value": {
      "holders": [],
      "host": "",
      "links": {
        "ids": [
          "nvme-SAMSUNG_MZVL21T0HCLR-00BL7",
          "nvme-SAMSUNG_MZVL21T0HCLR-00BL7_S64PNX0RC08136-part1",
          "nvme-SAMSUNG_MZVL21T0HCLR-00BL7_S64PNX0RC08136_1-part1",
          "nvme-eui.002538bc11b36458-part1"
        ],
        "labels": [],
        "masters": [],
        "uuids": [
          "95AE-5385"
        ]
      }
    }
  },
  {
    "key": "nvme0n1",
    "value": {
      "model": "SAMSUNG MZVL21T0HCLR-00BL7",
      "partitions": {
        "nvme0n1p1": {
          "holders": [],
          "links": {
            "ids": [
              "nvme-SAMSUNG_MZVL21T0HCLR-00BL7_S64PNX0RC08136-part1",
              "nvme-SAMSUNG_MZVL21T0HCLR-00BL7_S64PNX0RC08136_1-part1",
              "nvme-eui.002538bc11b36458-part1"
            ],
            "labels": [],
            "masters": [],
            "uuids": [
              "95AE-5385"
            ]
          }
        }
      }
    }
  }
]
```

up



meetup

Exemple

Un peu plus complexe

- debug:
msg: "{{ **ansible_facts.devices | dict2items |
map(attribute='value.partitions') | combine | dict2items |
selectattr('value.size', 'gt', '900') | map(attribute='key')** }}"


```
[
  {
    "nvme0n1p1": {
      "sectors": "1228800",
      "sectorsize": 512,
      "size": "600.00 MB",
      "start": "2048",
      "uuid": "95AE-5385"
    },
    "nvme0n1p2": {
      "sectors": "2097152",
      "sectorsize": 512,
      "size": "1.00 GB",
      "start": "1230848",
      "uuid": "cb6e18aa-be7e-491c-9316-92d2d56f3e89"
    },
    "nvme0n1p3": {
      "sectors": "1997080576",
      "sectorsize": 512,
      "size": "952.28 GB",
      ...
    }
  },
  ...
]
```



Exemple

Un peu plus complexe

- debug:
msg: "{{ ansible_facts.devices | dict2items |
map(attribute='value.partitions') | combine | dict2items |
selectattr('value.size','gt','900') | map(attribute='key') }}"



meetup

Map

L'outils magique

Filtre un élément d'une structure de donnée ou applique une opération à chaque élément d'une liste ou d'un dictionnaire.

Et donne accès à l'ensemble des autre filtre.

```
# whatami => 'able'
whatami: "{{ 'ansible' | regex_replace('^a.*i(.*?)$', 'a\\1') }}"

# whatami => [ 'able1', 'able2', 'able3' ]
whatami: "{{ ['ansible1', 'ansible2', 'ansible3', ] | map('regex_replace', '^a.*i(.*?)$', 'a\\1') }}"
```



Couper couper

Garder une bonne visibilité

- debug:

msg: >

```
}} ansible_facts.devices | dict2items |  
  map(attribute='value.partitions') | combine | dict2items |  
  selectattr('value.size','gt','900') | map(attribute='key') }}
```



meetup

json_query

Le meilleur pour la fin !

Utilise JMESPath. Un langage simple, populaire et complet.

- Librairie pour tous les principaux langages
- Jpterm, un terminal pour tester en live vos "requêtes"

<https://jmespath.org/tutorial.html>

```
[pblanc][void][~/it/events/20231108-Ansible-Meetup]
```

```
|
```

```
I
```

BONUS - BONUS - BONUS - BONUS - BONUS



meetup

Un T-Shirt a Gagner

Le premier qui trouve gagne !

Transcrire cette tache ansible vers **json_query**

- debug:

```
msg: "{{ ansible_facts.devices | dict2items |
map(attribute='value.partitions') | combine | dict2items |
selectattr('value.size','gt','900') | map(attribute='key') }}"
```

pierre@redhat.com

BONUS - BONUS - BONUS - BONUS - BONUS



meetup

I.A.

Il faut vivre avec son temps

Aide précieux pour gagner du temps.

Peut vous donner des exemples ou des inspirations

Comment gere ANSIBLE LIGHTSPEED ?



meetup

IA

Pas magique non plus

Attention aux resultats !

```
new_facts: "{{ ansible_facts | recursive_update(ansible_facts | dict2items | map('combine', {'key': item.key, 'value': (item.value | string if item.value is string else item.value)})) | items2dict }}"
```

BONUS - BONUS - BONUS - BONUS - BONUS - BONUS



meetup

Anonymiser

Attention données sensible !

Ne pas envoyer des informations sensibles à des services tiers

Utiliser `ansible_anonymiser` !

<https://github.com/ansible/anonymizer>

BONUS - BONUS - BONUS - BONUS - BONUS



meetup

Question ?