Ansible

신정훈 (jhshin9@gmail.com)

발표자

- 신정훈
- Korea Jboss User Group 멤바
- Cloudera APAC 에서 Solution Architect
- 러닝을 좋아하는 개발자
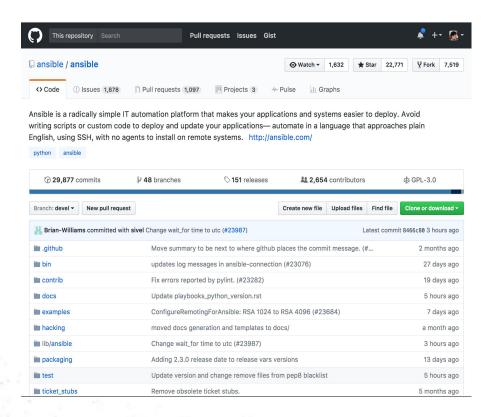
‘어슐러 르 귄’의 ‘로캐넌의 세계’라는 소설에서
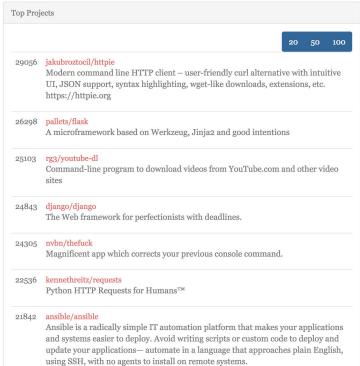행성간의 시공간을 초월하는 통신장비.
"answerable"에서 유래

Ansible이 뭐지?

프로비져닝(Provisioning) 툴

인기짱

Python Github project(ranking 7)
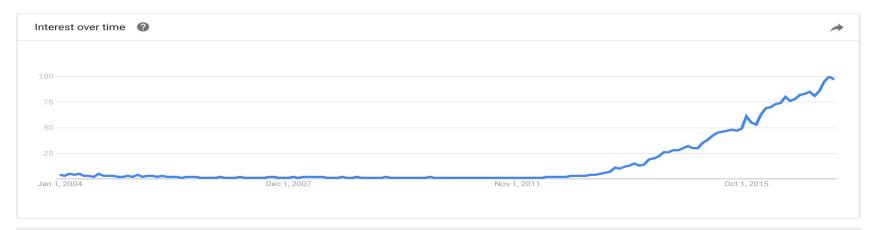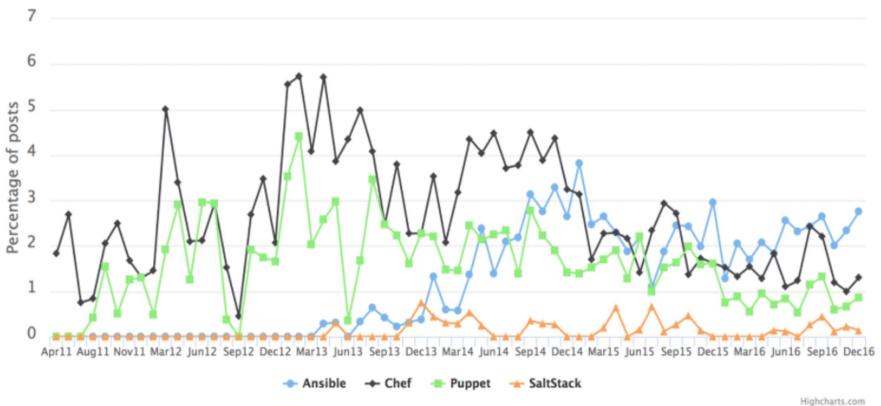
인기짱

Google trend

인기짱

Hackers news

그럼 왜 인기가 좋은데?

- 참 좋은 오픈소스
  - 활동성이 높음
  - 믿을만한 회사(레드햇)가 서포트
  - Training & Certification
  - Consulting
- 배우기 너무 쉬워
- 에이전트가 없네
- 병렬 프로비져닝
- 플러그앤 플레이
- 멱등성(Idempotency)
  - 연산을 여러 번 적용하더라도 결과가 달라지지 않는 성질

ansible

- 개발언어
  - python
- 정의
  - YAML
- 통신
  - SSH
- 실행결과
  - ok, failed, changed, unreachable

일단 **3**개만 먼저 알고 넘어갈까요

YAML

- YAML Ain't Markup Language
- JSON보다 더 읽기 쉬운 언어
  - **Human friendly** data serialization standard for all programming languages.

```yaml
- name: Configure Cloudera Manager Repository
  hosts: cdh_servers
  roles:
    - cm_repo
  tags: cm_repo

- name: Install Java
  hosts: cdh_servers
  roles:
    - java
  tags: java
```

Jinja2

- 파이썬 템플릿 언어

```
- name: Create MysqlDB log file
  file: path={{ mysql_log }} state=touch owner=mysql group=mysql mode=0775
```

```
{% if krb5_kdc_type == 'ad' %}
"value" : "Active Directory"
{% else %}
"value" : "MIT KDC"
{% endif %}
```

```
- name: Install MysqlDB packages
  yum: name={{ item }} state=present
  with_items:
    - mysql-server
    - MySQL-python
```

Inventory file

- 리모트 서버에 대한 메타 데이터
  - OS의 hosts 파일과 유사
- 기본 파일은 /etc/ansible/hosts
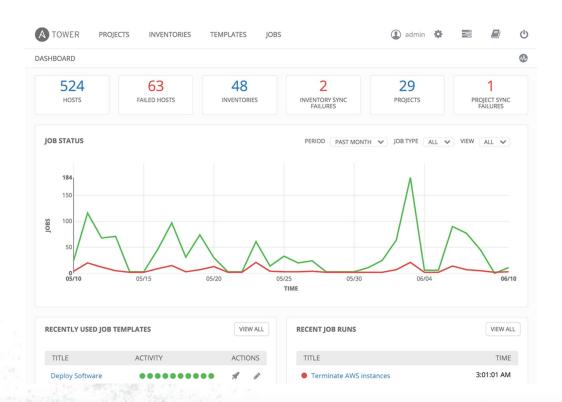 - 옵션(-i) 사용해서 따로 정의가능
- Grouping 가능

$ansible mn -m ping **-i** hosts

```
[mn]
mn[1:3].hadoop.com
[en]
en.hadoop.com
[dn]
dn[1:100].hadoop.com
```

mn1.hadoop.com
mn2.hadoop.com
mn3.hadoop.com

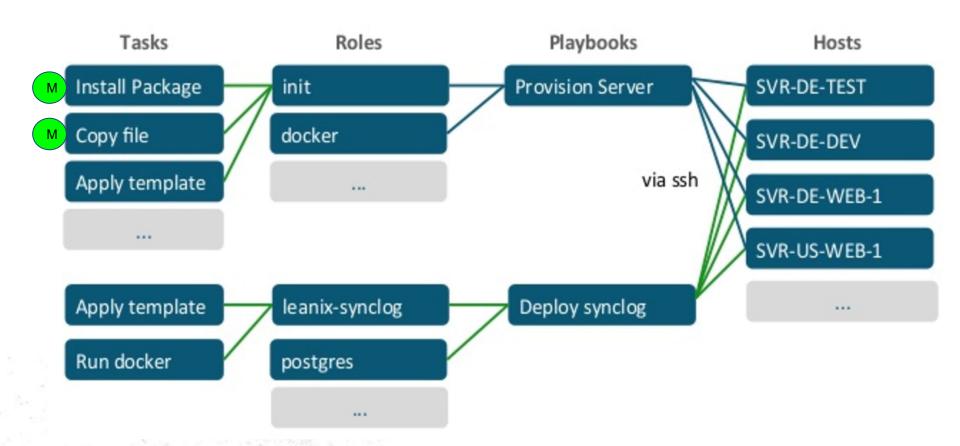# 실행 종류는 뭐가있나?

- Ad-hoc
- Playbook
- Tower

Ad-hoc

● 처음에는 이걸로 시작해서 자신감을 가져요
  $ansible all -m ping -i hosts

```
[root@ip-10-0-0-120 ansible]# ansible all -m command -a "hostname -f" -i hosts
10.0.0.124 | SUCCESS | rc=0 >>
ip-10-0-0-124.ap-northeast-2.compute.internal

10.0.0.123 | SUCCESS | rc=0 >>
ip-10-0-0-123.ap-northeast-2.compute.internal

10.0.0.125 | SUCCESS | rc=0 >>
ip-10-0-0-125.ap-northeast-2.compute.internal

10.0.0.126 | SUCCESS | rc=0 >>
ip-10-0-0-126.ap-northeast-2.compute.internal

10.0.0.120 | SUCCESS | rc=0 >>
ip-10-0-0-120.ap-northeast-2.compute.internal

10.0.0.122 | SUCCESS | rc=0 >>
ip-10-0-0-122.ap-northeast-2.compute.internal

10.0.0.121 | SUCCESS | rc=0 >>
ip-10-0-0-121.ap-northeast-2.compute.internal
```

JBoss Community

Playbook

- 복잡한것도 척척해서 잘난척 좀 할까

Playbook

$ansible-playbook **site.yml** -i hosts

```
- name: Configure Cloudera Manager Repository
  hosts: cdh_servers
  roles:
    - cm_repo            play1
  tags: cm_repo

- name: Install Java
  hosts: cdh_servers
  roles:               play2
    - java
  tags: java

- name: Install MysqlDB and create databases
  hosts: db_server
  roles:               play3
    - mysqldb
  tags: mysql

- name: Install MySQL Connector
  hosts: utility_servers:master_servers:worker_servers
  roles:
    - mysql_connector
  tags: mysql
```

Playbook Standard Layout

```
production              # inventory file for production servers
staging                 # inventory file for staging environment

group_vars/
    group1              # here we assign variables to particular groups
    group2              # ""
host_vars/
    hostname1           # if systems need specific variables, put them here
    hostname2           # ""

library/                # if any custom modules, put them here (optional)
filter_plugins/         # if any custom filter plugins, put them here (optional)

site.yml                # master playbook
webservers.yml          # playbook for webserver tier
dbservers.yml           # playbook for dbserver tier

roles/
    common/             # this hierarchy represents a "role"
        tasks/          #
            main.yml    #  <-- tasks file can include smaller files if warranted
        handlers/       #
            main.yml    #  <-- handlers file
        templates/      #  <-- files for use with the template resource
            ntp.conf.j2 #  <------- templates end in .j2
        files/          #
            bar.txt     #  <-- files for use with the copy resource
            foo.sh      #  <-- script files for use with the script resource
        vars/           #
            main.yml    #  <-- variables associated with this role
        defaults/       #
            main.yml    #  <-- default lower priority variables for this role
        meta/           #
            main.yml    #  <-- role dependencies
        library/        # roles can also include custom modules
        lookup_plugins/ # or other types of plugins, like lookup in this case

    webtier/            # same kind of structure as "common" was above, done for the webtier role
    monitoring/         # ""
    fooapp/             # ""
```

Module

- 실제로 일하는 친구
- 여러분들이 생각하는건 다 있습니다

Tags

- Playbook에 Play가 많을 경우 유용
- 특정 Tag 만 실행

  $ansible-playbook site.yml **--tags** "java,mysql"

- 특정 Tag 무시

  $ansible-playbook site.yml **--skip-tags** "java"

```
- name: Configure Cloudera Manager Repository
  hosts: cdh_servers
  roles:
    - cm_repo
  tags: cm_repo

- name: Install Java
  hosts: cdh_servers
  roles:
    - java
  tags: java

- name: Install MysqlDB and create databases
  hosts: db_server
  roles:
    - mysqldb
  tags: mysql

- name: Install MySQL Connector
  hosts: utility_servers:master_servers:worker_servers
  roles:
    - mysql_connector
  tags: mysql
```

JBoss Comm

Variables

group_vars/db_server.yml

```
mysql_datadir: /var/lib/mysql
mysql_socket: /var/lib/mysql/mysql.sock
mysql_port: 3306
mysql_log_bin: /var/lib/mysql/mysql_binary_log
mysql_log: /var/log/mysqld.log
mysql_pid_dir: /var/run/mysql
mysql_pid_file: "{{ mysql_pid_dir }}/mysqld.pid"
mysql_root_password: maste

databases:
  scm:
    name: 'scm'
    user: 'scm'
    pass: 'scm_password'
    type: 'mysql'
```

roles/mysqldb/**templates**/my.cnf.j2

```
[mysqld]
datadir   = {{ mysql_datadir }}
socket    = {{ mysql_socket }}
port      = {{ mysql_port }}

transaction-isolation = READ-COMMITTED
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links = 0

key_buffer          = 16M
key_buffer_size     = 32M
max_allowed_packet  = 32M
thread_stack        = 256K
thread_cache_size   = 64
query_cache_limit   = 8M
```

Handler

- 반복되는 작업을 수행
- 대부분의 작업은 애플리케이션 restart
- Task의 notify에 의해 호출됨

roles/mysqldb/**tasks**/main.yml

```
- name: Install MysqlDB packages
  yum: name={{ item }} state=present
  with_items:
    - mysql-server
    - MySQL-python

- name: Create MysqlDB configuration file
  template: src=my.cnf.j2 dest=/etc/my.cnf
  notify:
    - restart mysql
```

roles/mysqldb/**handlers**/main.yml

```
- name: restart mysql
  service: name=mysql state=restarted
```

Demo

http://www.jboss.org
https://www.facebook.com/groups/jbossusergroup