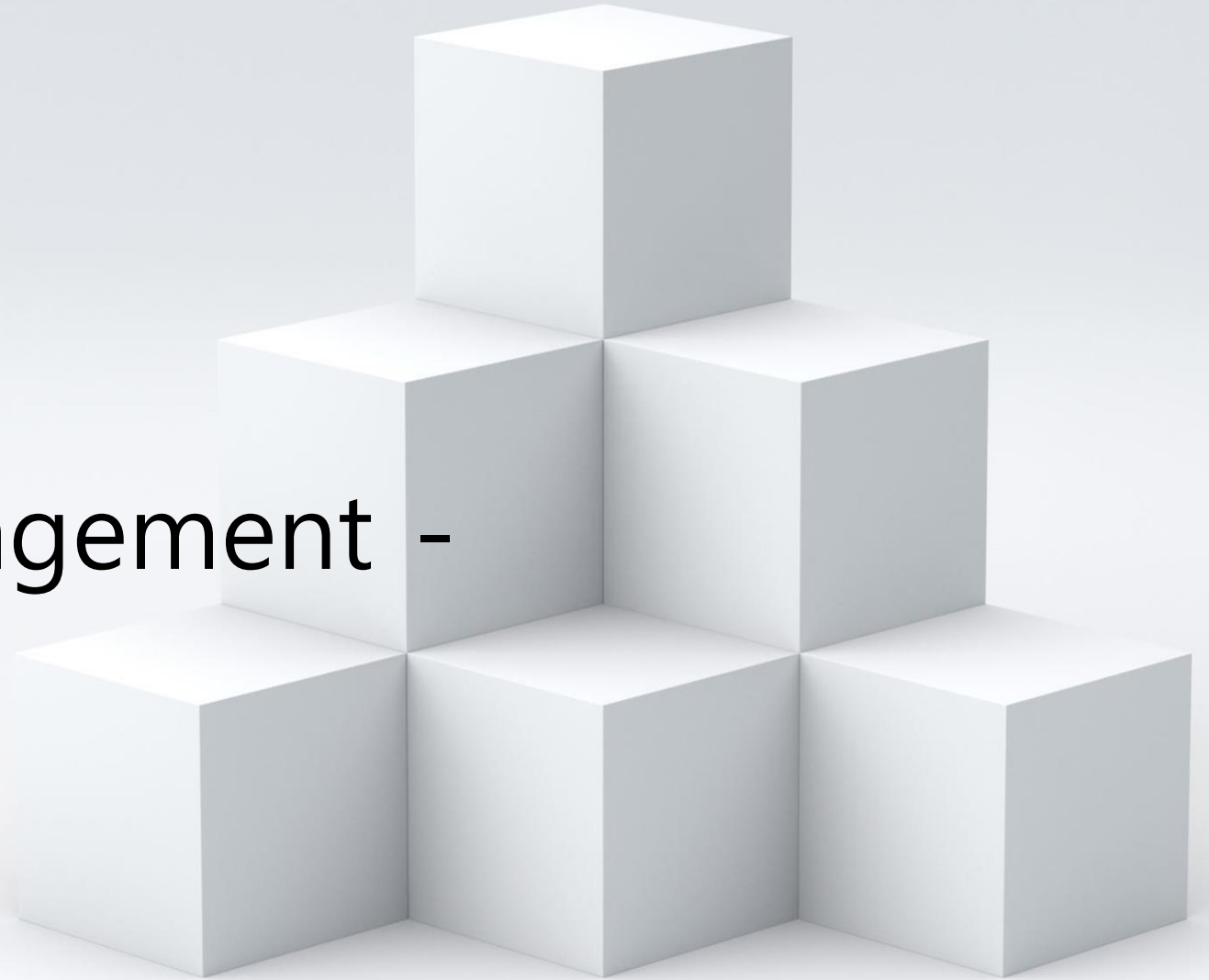


Software Engineering

- Configuration Management -

Professor Han-gyoo Kim

2022



Configuration Management

- Why automated CM?
 - Manage large scale heterogenous computer cluster
 - H/W monitoring
 - S/W install
 - Environment 불일치 문제 해소
 - 업무 Zero Downtime 달성을 위한 가장 기본적인 도구
- Basic tool for IT Automation
- IaC (Infrastructure as Code)
 - IT operation을 자동화
 - 중앙에서 수백 대 이상의 기기 (네트워크 기기 포함)에 각 기기에 필요한 환경을 provisioning
- Shell script vs CM tool
 - 관리자의 숙련도와 무관하게 효율적, 자동적으로 형상 관리

CM Tools – Puppet, Chef, Saltstack, Ansible

- 특정 도구가 우수하다고 주장하기 어려움

- 1) Scalability

- 2) Ease of Setup

- Ansible

- 3) Ease of Management

- Ansible

- 4) Interoperability

- Ansible installable on any OS

- 5) CM Language

- Ansible - YAML 파일

- 6) Popularity

- 7) Cost

- USD 10,000 / year / 1,000 devices

CM Architectures

- Pull Architecture

- Agents (clients)가 주기적으로 master (server)에게 configuration 문의
- 독자적인 기능 추가 편리

- Push Architecture

- Server가 주기적으로 node들에 configuration을 주입
- No client side setup at all
- How? <= Shell command with root privileges
- SSH connection from server to machines
- 간편

Ansible

- 상대적으로 새로운 도구 / Red Hat이 인수
- shell script 기반 / YAML 로 관리 사항을 기재
(procedural vs declarative)

(1) IT Automation

IT 업무자의 작업을 자동화하는 명령 기재

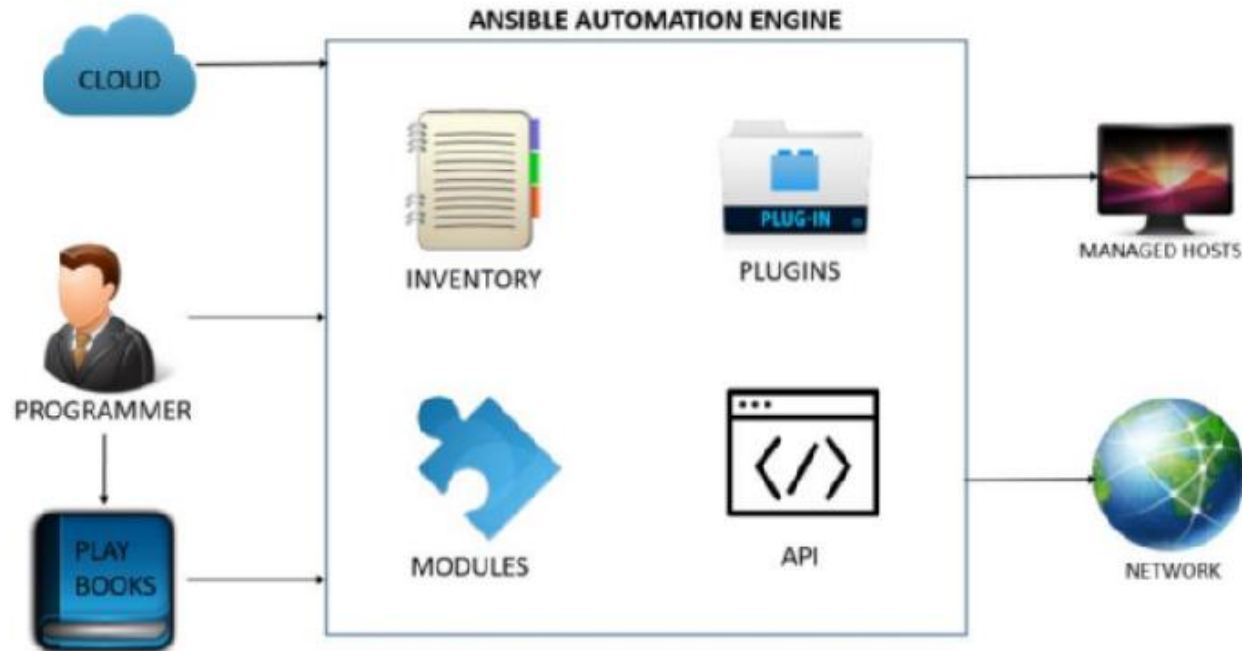
(2) Configuration Management

전체 infrastructure 의 일관성을 유지

(3) Automatic Deployment

다양한 환경에서 동작하는 앱들을 자동으로 배치(deploy)

Ansible Architecture Details



- Ansible automation engine is installed on local (server) machine
- 수백 ~ 수천, 수만의 노드 기계와 네트워크 장비들이 configuration의 대상물
- Module : 각 노드가 정해진 configuration이 되도록 하는 명령 프로그램
- Playbook : module들을 포함한 각 노드들의 configuration을 지휘하는 매뉴얼로서 각 노드들이 유지해야 하는 policy나 IT process step들을 기재함
- Ansible 엔진이 playbook 에 따라 각 노드들에게 SSH 연결을 통해 해당 module을 Push
- Inventory : Ansible이 관리하는 노드들의 이름을 모아둔 text file

Ansible master와 node들 사이의 ssh 연결

- (1) Root가 아닌 일반 사용자로써 ansible 운영 필요
- (2) Ansible 운영자(master/nodes)들 사이에서 ssh 연결
- (3) Ansible 운영자에 sudo 권한 필요
- (4) 마스터와 노드 기계 사이에서 ssh 연결에 필요한 key 공유
ssh-keygen
cat ~/.ssh/id_rsa.pub
ssh-copy-id <SSH key를 공유할 노드의 private ip address>
- (5) 노드에 SSH 연결을 위해 노드의 password 입력 필요
- (6) 매 번 SSH 연결 때마다 password 묻지 않고 자동으로 작업할 수 있도록 노드의 ssh demon configuration 수정

Ansible 설치

- Master와 node 모두에 설치
- \$sudo su -
- \$ apt update
- \$ apt install software-properties-common
- \$ add-apt-repository --yes --update ppa:ansible/ansible
- \$ apt install ansible
- \$ ansible --version

Inventory

- <https://ansible.com>
- https://docs.ansible.com/ansible/latest/user_guide/intro_inventory.html
- (Master 기계) /etc/ansible/hosts 편집하여 node 기계들의 private IP 주소 입력

Ansible 시작하기

(master 기계에서)

- `ansible --list-host all`
- `ansible <group name> -m ping`
- `ssh-keygen`
- `ls .ssh`
- `ssh-copy-id <private ip address of nodes>`

(node 기계들에서)

- `/etc/ssh/sshd_config` 파일 수정
 - `PasswordAuthentication yes`
 - `PermitEmptyPasswords yes`
 - `PermitRootLogin yes`
- `systemctl restart sshd`

(master 기계에서)

- `ssh-copy-id <private ip address of nodes>`

(node 기계에서)

- `passwd root` 실행하여 적당한 root password 입력

(master 기계에서)

- `ssh <private IP address of node machine>` 을 실행하여 node로 들어가는 것 확인
- `exit` 명령 실행하여 node 기계에서 master로 복귀
- `ssh-copy-id <private ip address of nodes>`

Ansible yaml 파일 예

(master 기계에서)

- hello.txt 파일 작성
 - Hello, Hongik
- test.yml 파일 작성

```
---  
- name : test  
  hosts : <group name>  
  gather_facts : true  
  tasks :  
    - name : copy hello.txt to nodes as /root/yap.txt  
    - copy : src=/root/hello.txt dest=/root/yap.txt
```
- ansible-playbook test.yml --syntax-check
- ansible-playbook test.yml

(node 기계에서)

- yap.txt 복제되어 생성된 것 확인

Ad-hoc commands

command hostgroups module/arguments[options]

ansible-doc -l | less

ansible-doc | grep -i user

ansible-doc user

ansible-doc apt

ansible <group name> reboot

ansible all -a uptime

ansible all -a "uname -a"

ansible all -m ping

ansible all -a "apt-get -y install apache2"

ansible all -m apt -a "name=apache2 state=latest"