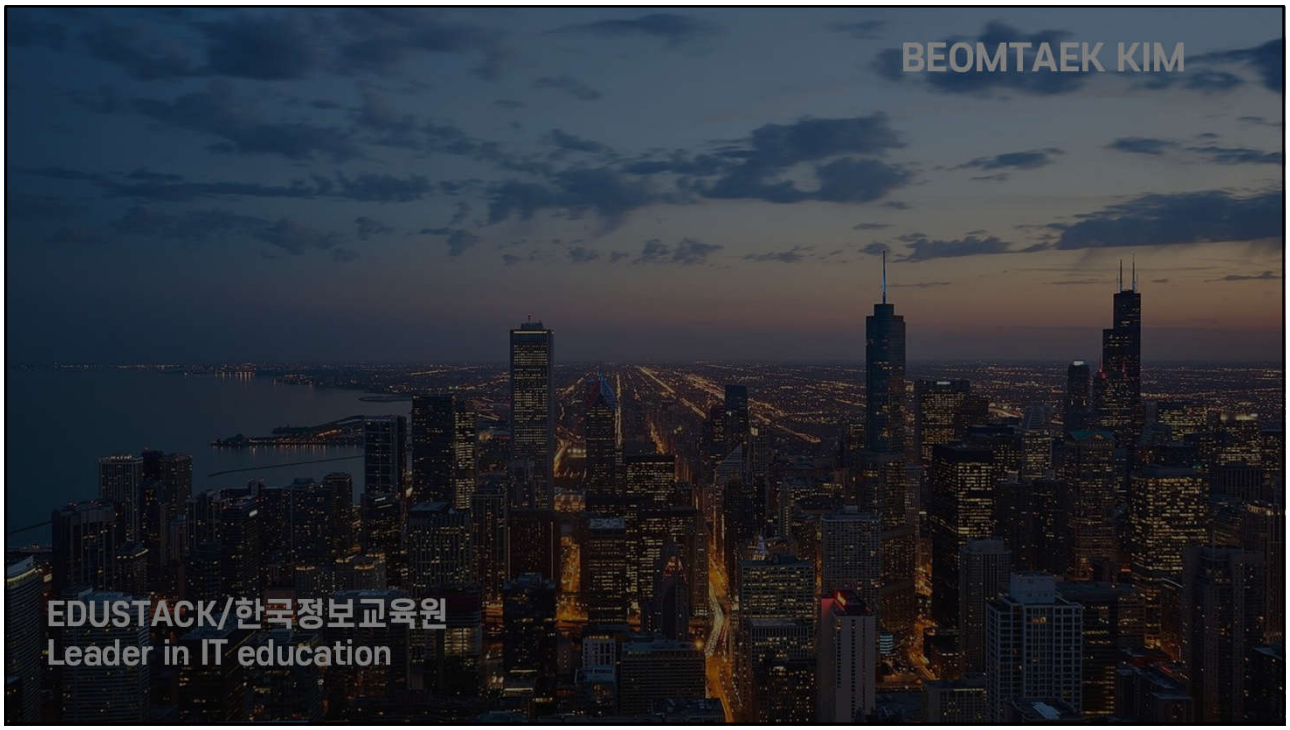
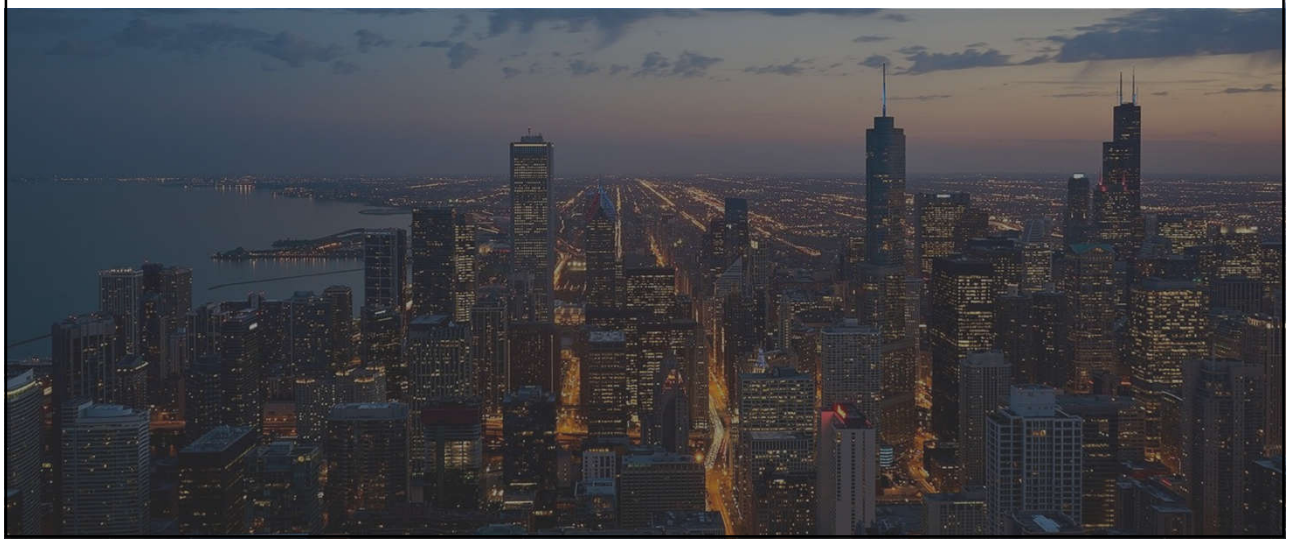


한국정보교육원

ansible Fundamentals



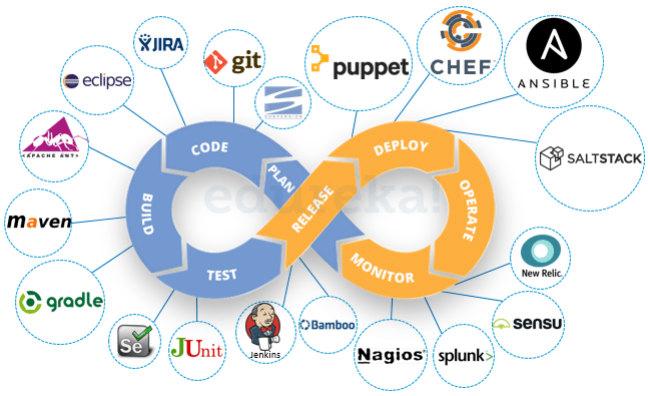
한국정보교육원

앤서블

한국정보교육원

앤서블의 현재

- DevOps



한국정보교육원

앤서블의 특징

- 에이전트가 없는 구조
- 협업도구
- 다양한 운영체제 통합관리
- 높은 보안과 신뢰성
- 멱등성

한국정보교육원

앤서블 체험하기

한국정보교육원

실습환경

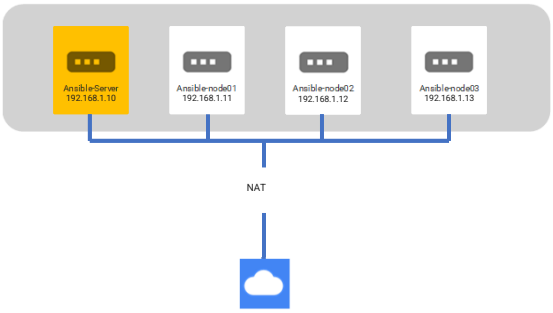
- vmware workstation
- CentOS-7-x86_64-Minial-1804.iso
- 모든 루트 패스워드

  test123
- 모든 노드

  1 CORE

  1 GB RAM

  20 GB DISK



한국정보교육원

앤서블 서버에 코어 설치

- 앤서블 서버에서 아래의 명령 실행을 통해 서버를 설치

yum -y update

yum -y install epel-release

yum -y install ansible

ls /

- 앤서블 서버에 노드 추가

```
## db-[99:101]-node.example.com
```

```
192.168.1.11
192.168.1.12
192.168.1.13
```

한국정보교육원

앤서블 서버에 코어 설치

- 앤서블 서버에서 노드들의 “known_hosts_key” 값을 받고 연결상태를 확인해 본다

  # ansible all -m ping // yes 를 세번 입력

  # ansible all -m ping -k

```
[root@Ansible-Server ~]# ansible all -m ping -k
SSH password:
192.168.1.12 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.1.13 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.1.11 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[root@Ansible-Server ~]#
```

9

한국정보교육원

기본연결

- 아래와 같이 hosts 파일에 [nginx] 섹션 추가 후 all 이 아닌 nginx 로 연결확인과 파일 이용한 접속

```
[root@Ansible-Server ~]# tail -5 /etc/ansible/hosts

[nginx]
192.168.1.11
192.168.1.12
192.168.1.13
[root@Ansible-Server ~]# ansible nginx -m ping -k
SSH password:
192.168.1.13 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.1.12 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.1.11 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[root@Ansible-Server ~]#
```

```
[root@Ansible-Server ~]# cat customized_inven.lst
192.168.1.11
192.168.1.12
192.168.1.13
[root@Ansible-Server ~]# ansible all -i customized_inven.lst -m ping -k
SSH password:
192.168.1.11 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.1.13 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.1.12 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[root@Ansible-Server ~]#
```

10

한국정보교육원

기본연결

- 노드 확인, 노드확인만을 하므로 -k 옵션이 없어도 동작한다

```
[root@Ansible-Server ~]# ansible all -m ping -k --list-hosts
SSH password:
  hosts (3):
    192.168.1.11
    192.168.1.12
    192.168.1.13
[root@Ansible-Server ~]# ansible -i customized_inven.lst all -m ping --list-hosts
  hosts (3):
    192.168.1.11
    192.168.1.12
    192.168.1.13
[root@Ansible-Server ~]# ansible -i customized_inven.lst all -m shell -a "echo $HOSTNAME" -k
SSH password:
192.168.1.11 | CHANGED | rc=0 >>
Ansible-Server
192.168.1.12 | CHANGED | rc=0 >>
Ansible-Server
192.168.1.13 | CHANGED | rc=0 >>
Ansible-Server
[root@Ansible-Server ~]#
```

11

한국정보교육원

다양한 모듈 사용(user)

- 사용자 생성과 확인

```
[root@Ansible-Server ~]# ansible all -m user -a "name=beomtaek" -k
SSH password:
192.168.1.13 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "comment": ""
```

- 이하 생략 -

```
[root@Ansible-Server ~]# ansible all -m shell -a "cat /etc/passwd | grep beomtaek" -k
SSH password:
192.168.1.13 | CHANGED | rc=0 >>
beomtaek:x:1001:1001::/home/beomtaek:/bin/bash
192.168.1.11 | CHANGED | rc=0 >>
beomtaek:x:1001:1001::/home/beomtaek:/bin/bash
192.168.1.12 | CHANGED | rc=0 >>
beomtaek:x:1001:1001::/home/beomtaek:/bin/bash
[root@Ansible-Server ~]#
```

12

한국정보교육원

다양한 모듈 사용(user)

- 사용자 삭제와 확인

```
[root@Ansible-Server ~]# ansible all -m user -a "name=beomtaek state=absent" -k
SSH password:
192.168.1.12 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "force": false,
    "name": "beomtaek",
    "remove": false,
    "state": "absent"
```

- 이하 생략 -

```
[root@Ansible-Server ~]# ansible all -m shell -a "cat /etc/passwd | grep beomtaek" -k
SSH password:
192.168.1.11 | FAILED | rc=1 >>
non-zero return code
192.168.1.12 | FAILED | rc=1 >>
non-zero return code
192.168.1.13 | FAILED | rc=1 >>
non-zero return code
[root@Ansible-Server ~]#
```


한국정보교육원

다양한 모듈 사용(yum)

- 패키지 설치와 삭제

```
[root@Ansible-Server ~]# ansible all -m yum -a "name=git state=present" -k
SSH password:
192.168.1.13 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "changes": {
        "installed": [
            "git"
```

- 이하 생략 -

```
[root@Ansible-Server ~]# ansible all -m yum -a "name=git state=absent" -k
SSH password:
192.168.1.11 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "changes": {
        "removed": [
            "git"
```

[실습]

yum 모듈을 이용하여 각 노드에 httpd 를 설치하세요

한국정보교육원

다양한 모듈 사용(copy)

- 생성된 파일을 디렉토리에 이동시키기

```
[root@Ansible-Server ~]# curl www.keduit.com -o index.html
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 68113    0 68113    0     0   109k      0 --:--:-- --:--:-- --:--:--  109k
[root@Ansible-Server ~]#
[root@Ansible-Server ~]# ansible all -m copy -a "src=index.html dest=/var/www/html/index.html" -k
SSH password:
192.168.1.12 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
```

- 이하 생략 -

```
[root@Ansible-Server ~]# ansible all -m shell -a "ls /var/www/html" -k
SSH password:
192.168.1.11 | CHANGED | rc=0 >>
index.html
192.168.1.13 | CHANGED | rc=0 >>
index.html
192.168.1.12 | CHANGED | rc=0 >>
index.html
[root@Ansible-Server ~]#
```


한국정보교육원

다양한 모듈 사용(service)

- 서비스의 실행과 확인

```
[root@Ansible-Server ~]# ansible all -m shell -a "systemctl stop firewalld" -k
SSH password:
192.168.1.11 | CHANGED | rc=0 >>

192.168.1.13 | CHANGED | rc=0 >>

192.168.1.12 | CHANGED | rc=0 >>

[root@Ansible-Server ~]# ansible all -m service -a "name=httpd state=started" -k
SSH password:
192.168.1.11 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
```

- 이하 생략 -

- 192.168.1.11 ~ 13 에서 동일 페이지 확인 -

한국정보교육원

플레이북 playbook

- 서비스 배포까지 복잡한 단계를 플레이 북으로 통합
- yaml 파일에 필요한 내용을 절차적으로 작성하여 실행
- 연산을 여러번 하더라도 결과가 달라지지 않는 성질(멱등성) 적용

한국정보교육원

플레이북 playbook

- 플레이 북의 작성과 실행을 통한 서비스 배포

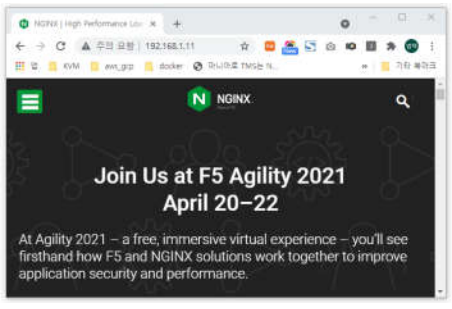
```
[root@Ansible-Server ~]# cat nginx_install.yml
---
- name: Install nginx on linux
  hosts: nginx
  gather_facts: no

  tasks:
    - name: install epel-release
      yum: name=epel-release state=latest
    - name: install nginx web server
      yum: name=nginx state=present
    - name: upload default index.html for web server
      get_url: url=https://www.nginx.com dest=/usr/share/nginx/html/ mode=0644
    - name: stop httpd web server
      service: name=httpd state=stopped
    - name: start nginx web server
      service: name=nginx state=restarted
[root@Ansible-Server ~]#
[root@Ansible-Server ~]# ansible-playbook nginx_install.yml -k
SSH password:

PLAY [Install nginx on linux] **************************************************************************************

TASK [install epel-release] ****************************************************************************************
ok: [192.168.1.12]
ok: [192.168.1.13]
ok: [192.168.1.11]
```

- 이하 생략 -



한국정보교육원

플레이북 playbook

- 플레이 북의 코드 내용

```
---
- name: Install nginx on linux
  hosts: nginx
```

"---" 는 야믈 파일의 시작을 의미
(선택사항) "name" 을 작성하여 일종의 설명을 첨부한다
"hosts" 는 /etc/ansible/hosts 에 등록된 호스트 중 [nginx] 섹션 하에 있는 호스트를 의미한다

- 이하 생략 -