API: application programming interface

두 애플리케이션이 서로 대화하는 방법. 기계를 위한 인터페이스

REST: Representational State Transfer

6 principles

* Client-server
* Code-on-Demand
* Stateless
* Layered System
* Uniform Interface
* Cacheable
* Stateless

DNA Center

* SDA (software Defined Access, 층간 스위치용 SDN) 컨드롤러
* 무선 NMS : 단말 상태 확인(현재 및 과거)
* 진보된 NMS(PI 및 NMS 대체)

DNA Center Platform - Southbound APIs & SDKs

* Intent-based Network Infrastructure
* Network / IT Process Integrations
* Cross-Domain Integrations
* Network Apps
* Business Apps

YAML : 가독성을 높인 직렬화 언어, 데이터를 표현하는 데이터 언어

자료형

* Scalar : 변수, 따옴표 미사용
* Sequence: list 자료형, ‘-‘를 사용하여 변수 리스트 표현
* Mappings: dict 자료형, ‘:’를 사용하여 key와 value 분리

구조

* 공백에 민감. 공백과 mappings를 사용해 계층구조를 가짐

1. Inventory File – Hosts & Groups

* Inventory file : 앤서블과 연결할 호스트 정보, 그룹 정보를 포함.
* Host 별로 그룹화하고, 관리의 용이성을 위해 따로 관리
* Host는 i에와

– Nested Groups & Ranges

* Children-parent 구조를 사용해 계층구조화된 그룹 표현 가능
* 여러 호스트 그룹을 위한 range표현 방식

1. Variables file

* 변수 표현을 위한 다양한 형식을 제공, 여러 위치에 저장 가능
* 해당 파일을 main 파일인 playbook에서 불러와서 사용

1. Playbook

* 앤서블이 동작할 task들의 집합을 모아놓은 yaml파일
* 여러 개의 task를 가질 수 있음
* Name을 task마다 입력해서 어떤 동작을 실행 중인지 명시 (디버깅에 도움을 줄 수 있음)

Play는 인벤토리 파일의 특정 그룹을 언급하여 해당 그룹과 연결할 수 있음.

‘-‘ 를 사용하여 play를 구분해야 함.

Name: 선택적인 컴포넌트, 어떤 내용으로 구성될 지 표현, 없어도 동작 가능

Hosts: 인벤토리 파일의 어떤 호스트나 그룹이 이 플레이에서 사용할 것인지 명시

Tasks: 타켓 호스트에 실행될 여러 액션들의 리스트를 표현할 수 있음.

Conditionals and loops

when: 조건문이 성립할 때만 task가 실행되도록 도움

ios\_facts

ios\_fackts 모듈은 ios 기기와 연결하여 기본적인 정보를 가져올 때에 사용

ansible net version 변수와 같은 일부 변수의 정보만 가져올 때 ios facts 모듈을 사용

ios\_command

config command가 아닌 모든 ios의 command를 지원하는 모듈

ios\_config

설정 comfig command를 기기에 전송, 추가나 변경 가능

Parents를 사용하여 acl라인들이 생성되기 전의 command를 입력해줄 수 있음.

Before를 사용하여 기존 acl을 지우는 command를 먼저 실행해 줄 수 있음.

멱등성의 원칙을 지키지 않음

Ios\_vrf

Ios\_user

Local user 정보 설정

Netconf\_config

다양한 플랫폼에 앤서블을 통한 configuration이 가능, yang 모델을 사용해 특정 영역을 수정할 수 있음

