

대규모 클라우드 인프라를 위한 프로비저닝 시스템

윤무열, 한영주, 최혁승, 전승훈
삼성전자 소프트웨어 R&D 센터
e-mail : { myeol.yun, yj616.han, hyok.choi, shjeon01 }@samsung.com

Automated Provisioning System for Large-scale Cloud Infrastructure

Mu-Yeol Yun, Youngjoo Han, Hyuk-Seung Choi, Seung-Hun Jeon
Software R&D Center, Samsung Electronics

요약

클라우드 기술이 발전되고 서비스가 확산됨에 따라 저장하고 처리해야 할 데이터는 많아지고 있다. 증가하는 데이터에 맞춰 서버 증설이 필요하지만 양이 증가할수록 관리 및 설정에 대한 인력리소스 필요량과 휴면에러에 대한 리스크는 점차 증가하는 문제가 있다. 본문에서는 서버 증설에 의한 작업량과 리스크를 줄이고 관리를 구조적이고 효율적으로 사용할 수 있게 개발된 AP (Automated Provisioning) 시스템을 소개한다.

1. 서론

과거 클라우드 인프라를 구축하려면 수 백대 서버의 설치 및 설정을 수작업으로 하였다. 이러한 과정은 많은 인력 및 시간을 필요로 하며 휴면에러에 대한 위험성을 감수해야했다. 이러한 문제점을 없애기 위해 프로비저닝 도구들이 개발, 연구되었지만 특정 영역만 지원해줄뿐 전체적인 지원은 불가하였다.

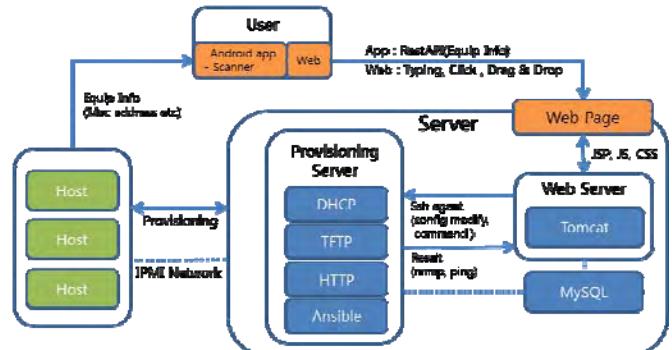
이같은 한계점을 극복하기 위해 AP 시스템이 제안되어 개발 되었다. 본 시스템은 서버의 네트워크 설정 부터 OS 설치, System 환경구축 까지 자동으로 구축해주며 인프라 구축시 사용되는 스위치, 서버, 스토리지 등 장비들을 실장도처럼 표현함과 동시에 관련 정보들을 통합관리 해준다.

2. 본문

클라우드 인프라는 서버 인프라를 지속적으로 유지는 물론 다운타임없이 업데이트를 지원해야 한다. 이를 지원하기 위해서 본 논문에서는 자동화된 프로비저닝 시스템을 제안한다.

2.1 AP 시스템 구조

인프라 환경에는 다수의 장비들이 존재하기 때문에 효율적인 관리를 위해서는 위치 순서와 기능에 따른 닉네임이 정해지고 이를 통합하여 관리하는 것이 필요하다. 이런 통합관리를 위해 정보수집, 시각화 두 단계로 나누었고 셋팅을 위한 프로비저닝의 단계를 두었다. 그림 1은 본 논문에서 소개하는 AP 시스템의 전체적인 구조와 각 모듈간의 정보 및 제어흐름을 나타낸다.

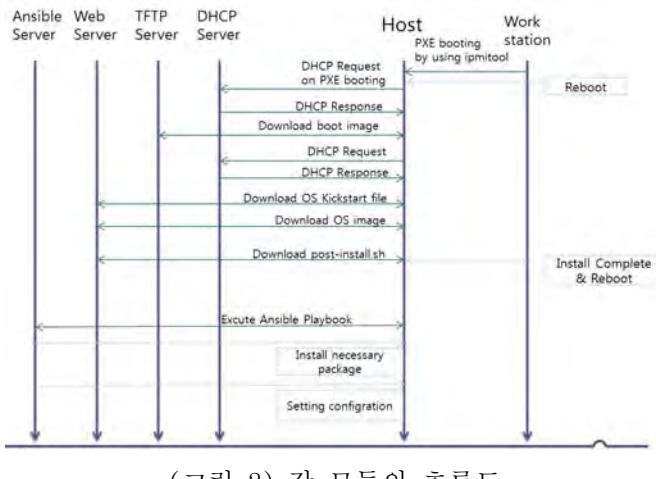


(그림 1) AP 시스템 구조

먼저 바코드(안드로이드 App, Scanner)를 통해 장비의 정보를 수집하고, 수집된 정보는 RestAPI를 통해 시스템의 DB에 저장된다. 웹을 통해 저장된 정보를 가시화시켜주며, 서버 OS 설치 시 기존 저장한 정보와 추후 설정한 정보를 기반으로 DHCP, TFTP, HTTP를 사용해 OS 설치한다. OS 설치가 완료되면 Ansible [REF]를 통해 원하는 환경으로 설정하게 된다.

2.2 Working Procedure

서버의 OS 설치 및 System 설정을 휴면에러 없이 설치하기 위해 Web Portal에 등록된 정보를 Provisioning Server를 통해 OS 설치를 하게 한다. Provisioning Server는 DHCP, TFTP, HTTP, Ansible 서버로 구성되어 있다. 그림 2는 각 모듈의 흐름도이다.



(그림 2) 각 모듈의 흐름도

Web에서 서버 필요 정보를 입력하고 Launch 버튼을 누르면 각 서버에 설정된 IPMI를 통해 PXE 부팅을 시켜준다. 그러면 저장된 정보를 기반으로 생성된 DHCP를 통해 동적 IP 할당을 받게 되며 TFTP를 통해 Server 설치시 필요한 커널이미지를 받는다. 그 후 HTTP를 통해 Kickstart 파일 및 OS 이미지를 받아 OS 설치가 진행된다. 완료 후 Post-install.sh가 실행되며 Ansible과 연결고리를 생성해주고 설정된 Configuration을 수행하여 완료시킨다.

2-3. 구현

인프라의 하드웨어를 구조적으로 관리하기 위해 장비 구매 시 제공되는 바코드를 이용한다. 바코드에는 장비의 S/N, 모델명, 스펙, MAC 주소 등 해당 장비가 가지는 정보를 포함하고 있으며 이를 Android App, Barcode Reader 기를 통해 읽고 RestAPI를 통해 Web Portal로 전송, 저장하게 된다. 저장 전 Web에서 위치할 랙을 선택하게 되는데 이를 통해 실장도를 구성 할 수 있다. 랙에 구성장비로 들어간 장비는 Typing, Click, Drag&Drop을 통해 IP, IPMI, 메모 설정, 위치 설정을 할 수 있다.

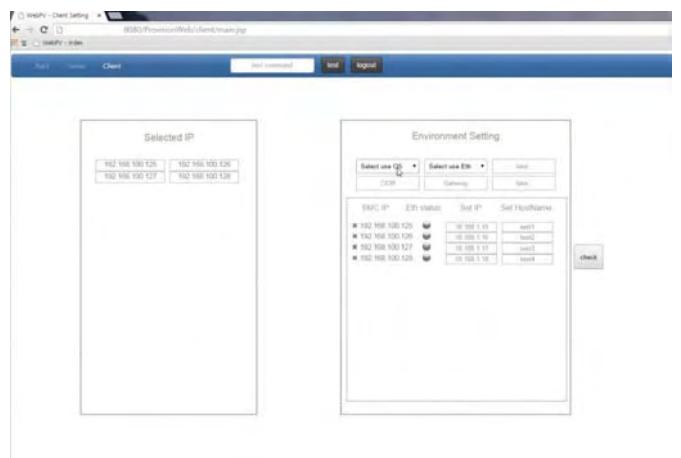
유저 인터페이스는 크게 Web에서의 Rack, Client, Server 페이지와 안드로이드 barcode app으로 나뉘어 진다. 그림 3은 AP 시스템을 통해 장비의 정보를 바코드로 추출하여 설정하고 있는 Rack[REF] 페이지 화면이다.



(그림 3) Rack 페이지 화면

Rack 페이지에서는 랙의 크기, 이름 등을 설정하며 해당 랙에 구성된 장비들의 갯수 및 목록을 볼 수 있다. 또한 barcode app을 통해 추출된 장비의 순서와 네임 등 정보를 지정하여 실제 구성된 실장도를 표현하여 관리의 편의성을 제공해준다.

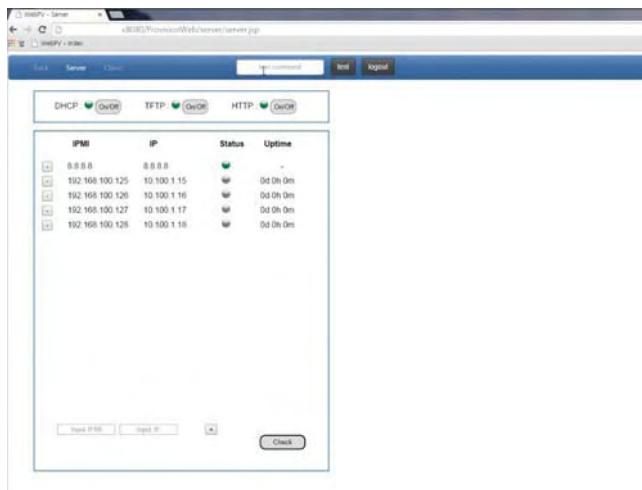
그림 4은 해당 서버를 OS 프로비저닝 하기 위해 셋팅중인 Client 페이지 화면이다.



(그림 4) Client 페이지 화면

Client 페이지에서는 OS 프로비저닝 할 서버를 선택 후 설정 값을 확인 및 입력 할 수 있게 해 준다. OS 선택, Network 및 Server 정보를 셋팅할 수 있다.

그림 5는 서버 OS 설치 후 설치상태를 확인 할 수 있는 Server 페이지 화면이다.



(그림 5) Server 페이지 화면

Server 페이지에서는 OS 설치시 사용되는 프로토콜의 상태, 설치된 또는 상태를 보고 싶은 장비의 통신 상태를 확인 할 수 있다.

3. 결론

다수의 장비로 구성되는 인프라 환경을 효율적으로 관리하기 위해서는 모든 장비를 통합하여 관리하는 것이 상당히 중요하며 서버 셋팅시 작업량과 휴면에 러를 줄이기 위해 자동으로 설정 및 배치하는 것이 필요하다. 이를 위해 본 논문에서는 바코드와 Web, Provisioning 서버의 연계의 방법을 제안하고 이를 구현하여 실제 서비스에 활용하고 있다.

향후 서버 외 다른 장비들도 안정된 운용을 위해 프로비저닝에 대한 연구가 필요하다.

Reference

- [1] Ansible : 시스템 환경 설정 및 애플리케이션 배포 자동화 플랫폼
- [2] Rack : 전산장비를 수납 보관하는 프레임