ONOS 제어기를 위한
Control Clustering, Security, IoT 확장기능 구현

DevCon 2016

KAIST POSTECH

KWANGWOON UNIVERSITY

K-ONE

Heedo Kang
(kangheedo@kaist.ac.kr)

Network and System Security(NSS) Lab.
School of Computing, KAIST

Jinwoo Kim
(jinwoookim@kaist.ac.kr)

Network and System Security(NSS) Lab.
School of Computing, KAIST

Seunghyeon Lee
(coksm1963@kaist.ac.kr)

Network and System Security(NSS) Lab.
School of Computing, KAIST

Security-Mode ONOS

DevCon 2016

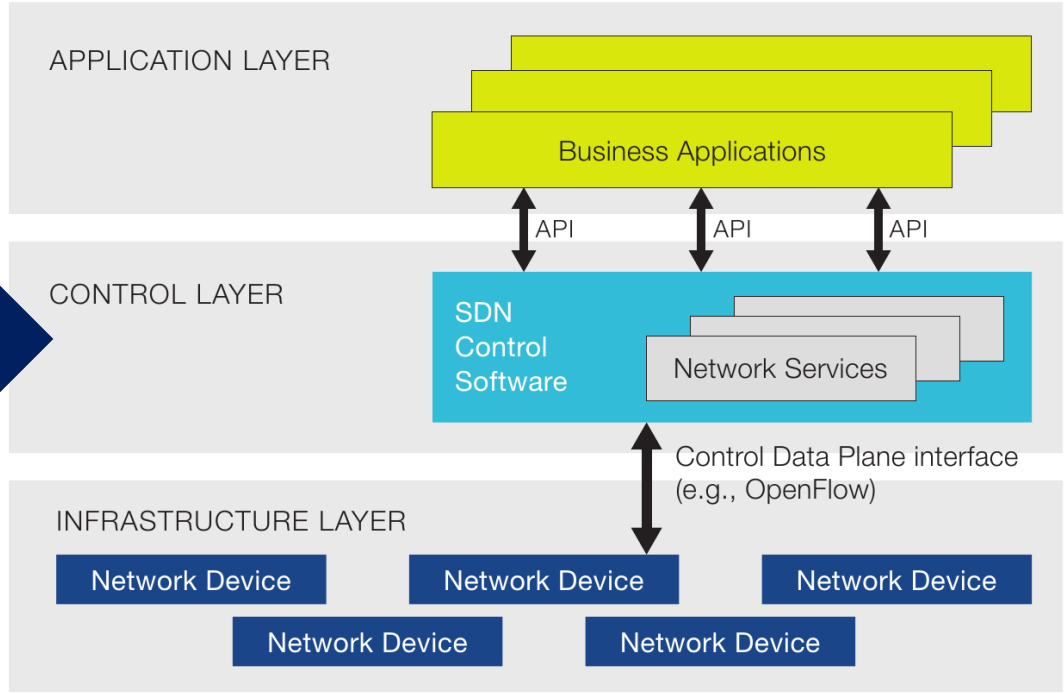
Heedo Kang
KAIST

K-ONE

KAIST

ONOS ?

SDN Controller

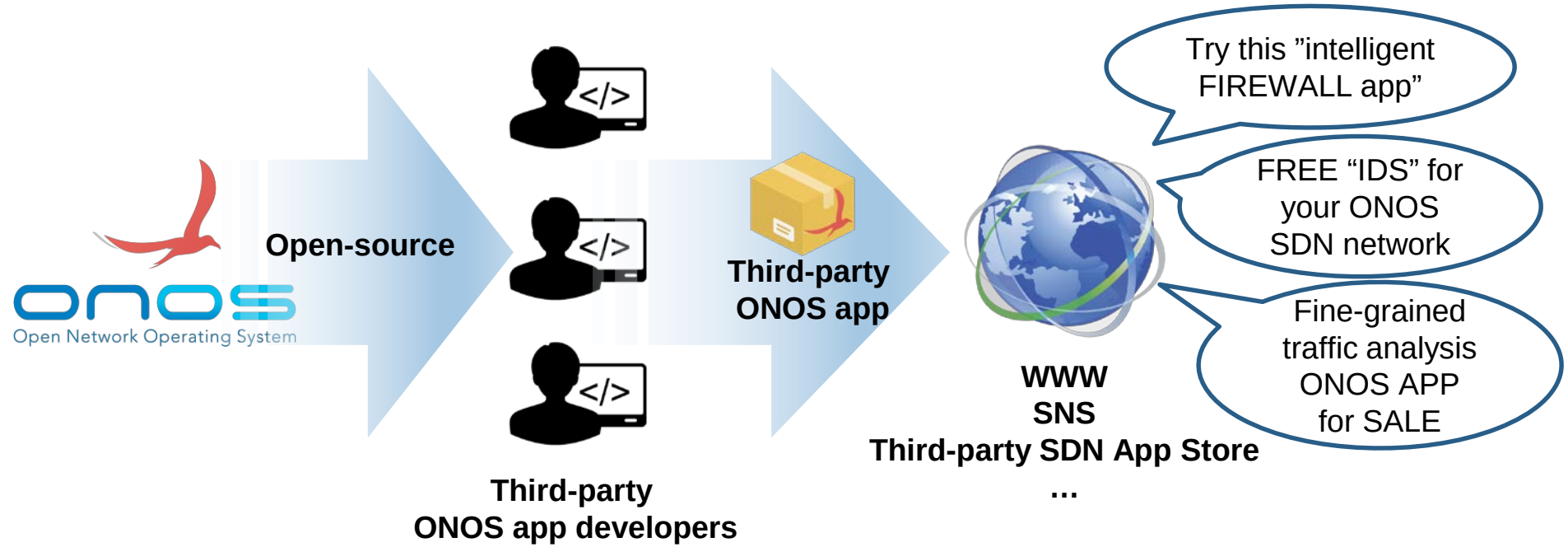


K-ONE

KAIST

Open Application Ecosystem

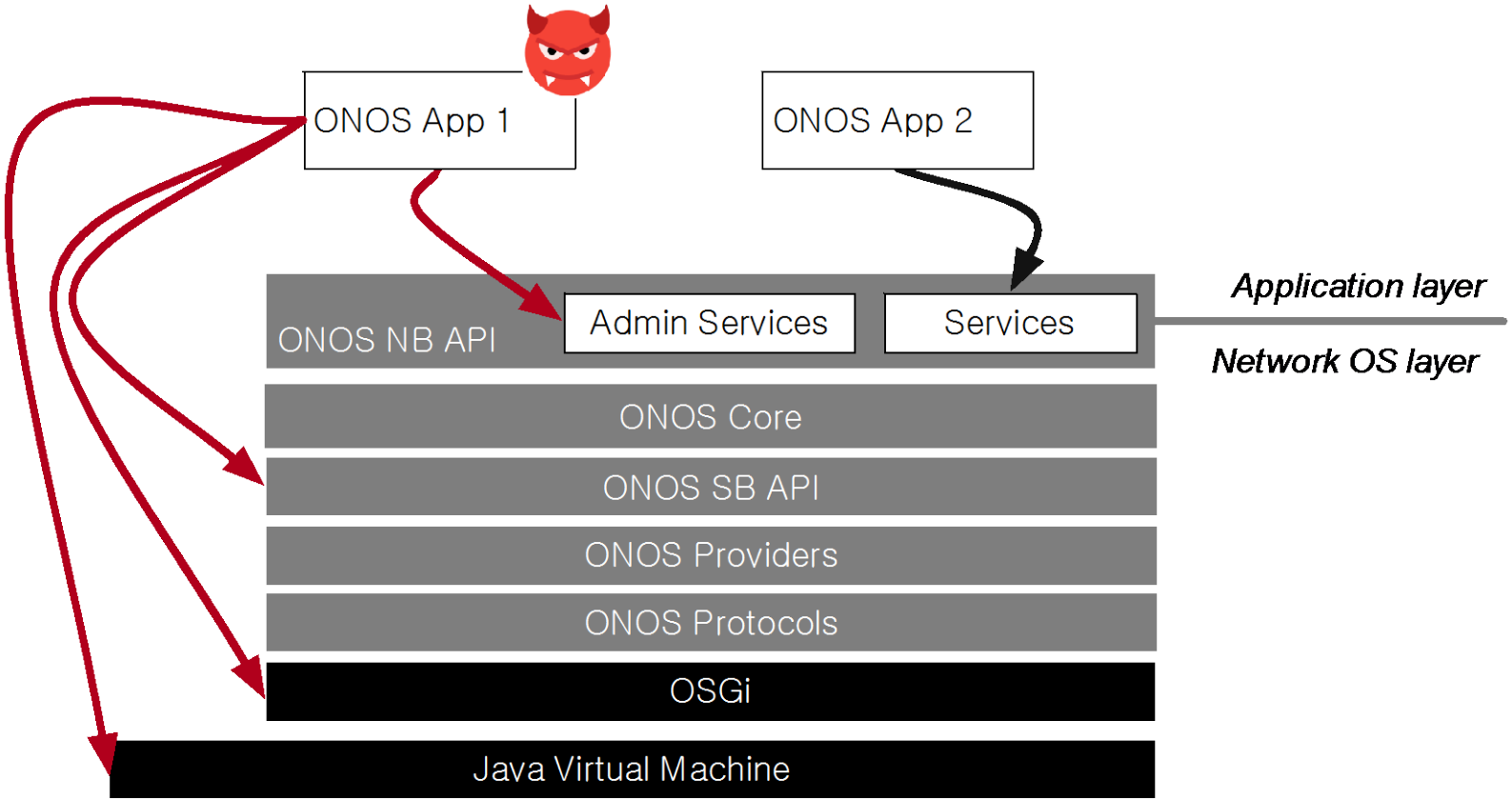
Accelerates and encourages
innovative and useful ONOS application development & distribution



K-ONE KAIST

Open Application Ecosystem

ONOS Architecture

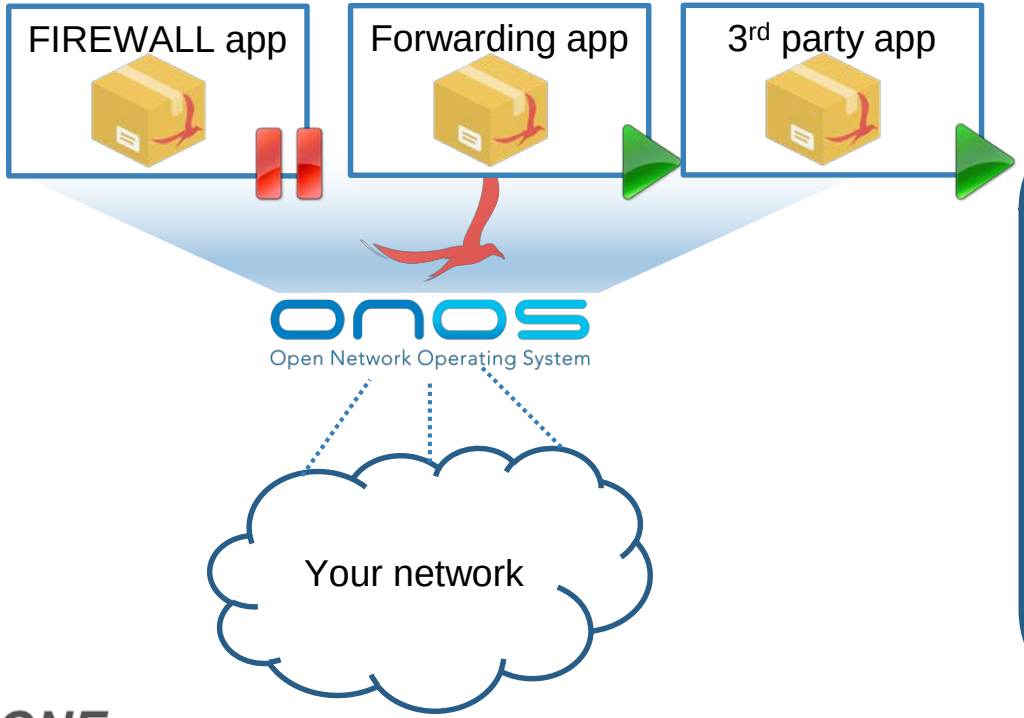


K-ONE

KAIST

Motivating examples

- Administrative Northbound API access



Hi! I am a FREE INTELLIGENT **Intrusion Detection Service** application for your ONOS managed network. **(DEACTIVATE)** I automatically download known **(FIREWALL APPLICATION)** signatures and blacklists from the internet and dynamically protect **(and create security hole)** your network from both internal and external threat. This application will give you the same level of protection as any other commercial security appliances.

K-ONE

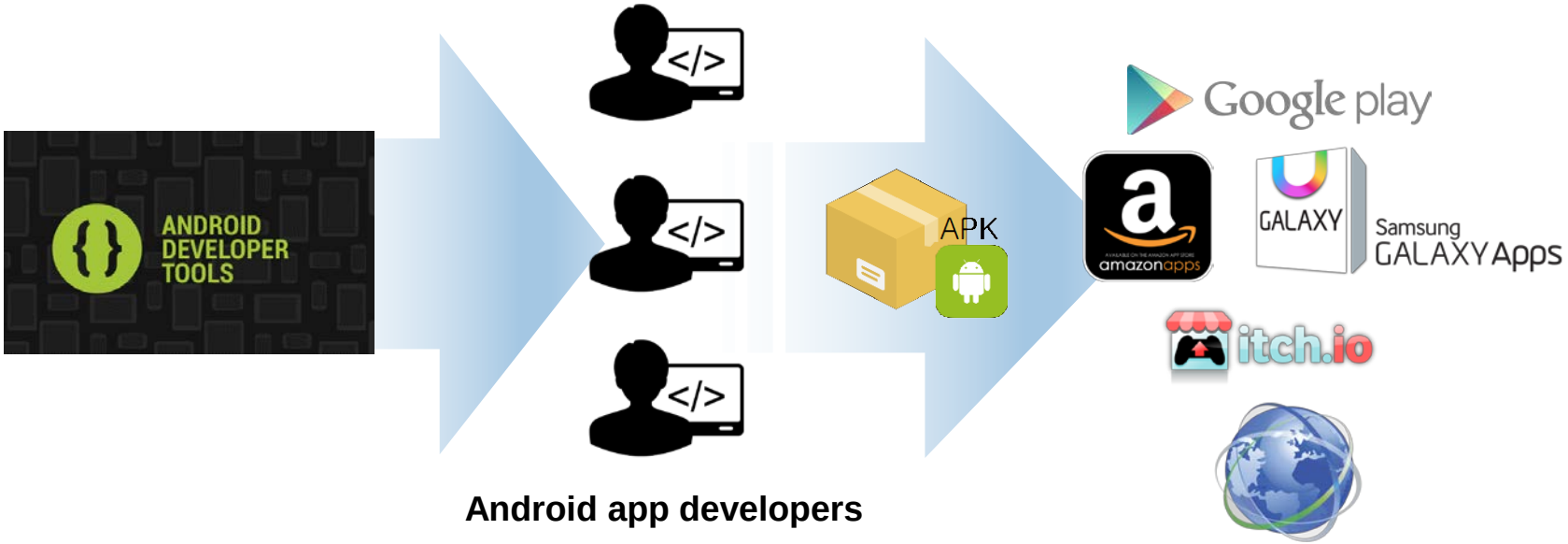
KAIST

Vetting applications

- **Manually inspect** the source code of apps line by line
  - Time-consuming
  - Prone to human error
  - Source-code may not be available

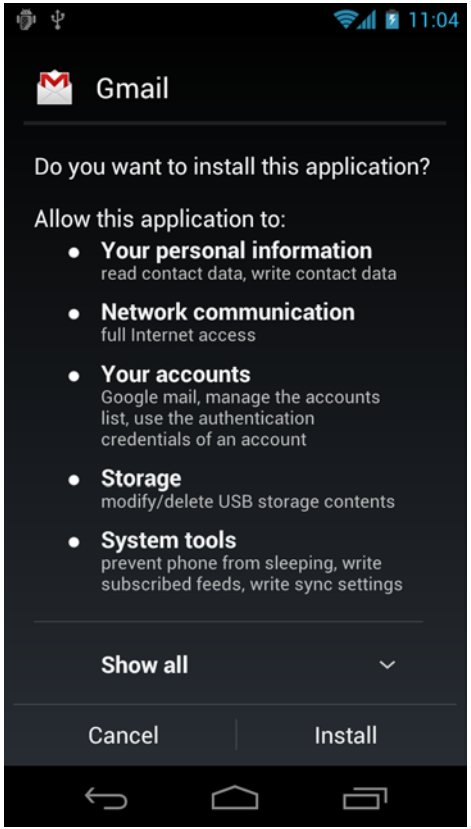
- **Automated analysis methods**
  - Static analysis – source code required
  - Dynamic analysis – expensive, low code coverage

Mobile application ecosystem



Android app developers

Vetting application



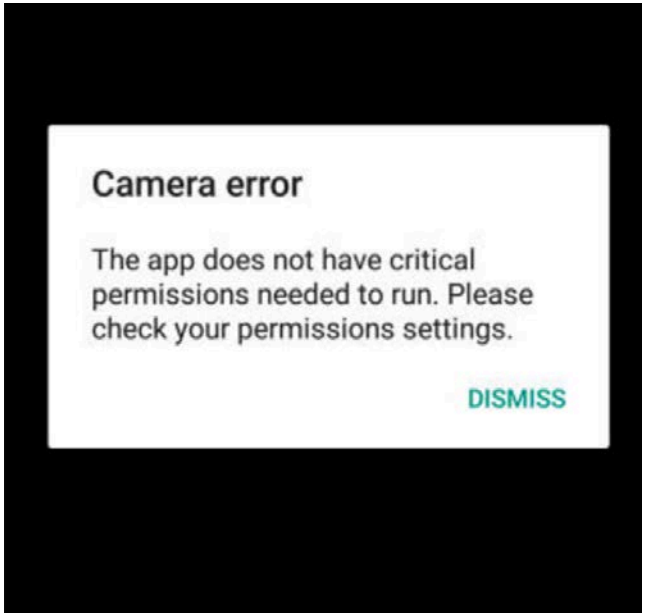
Mobile applications

Users are responsible for installing an app

Before installation,

- User must agree to grant a list of permissions that an app requires
- Show what this app is capable of
- Let the user decide!

Sandboxing application



Once deployed,

- Security policy is enforced (or a set of permissions is granted) to the application
- Application cannot access the resource that requires a certain permission, unless explicitly granted.

Security-Mode ONOS

Inspired by the security mechanism for Mobile systems

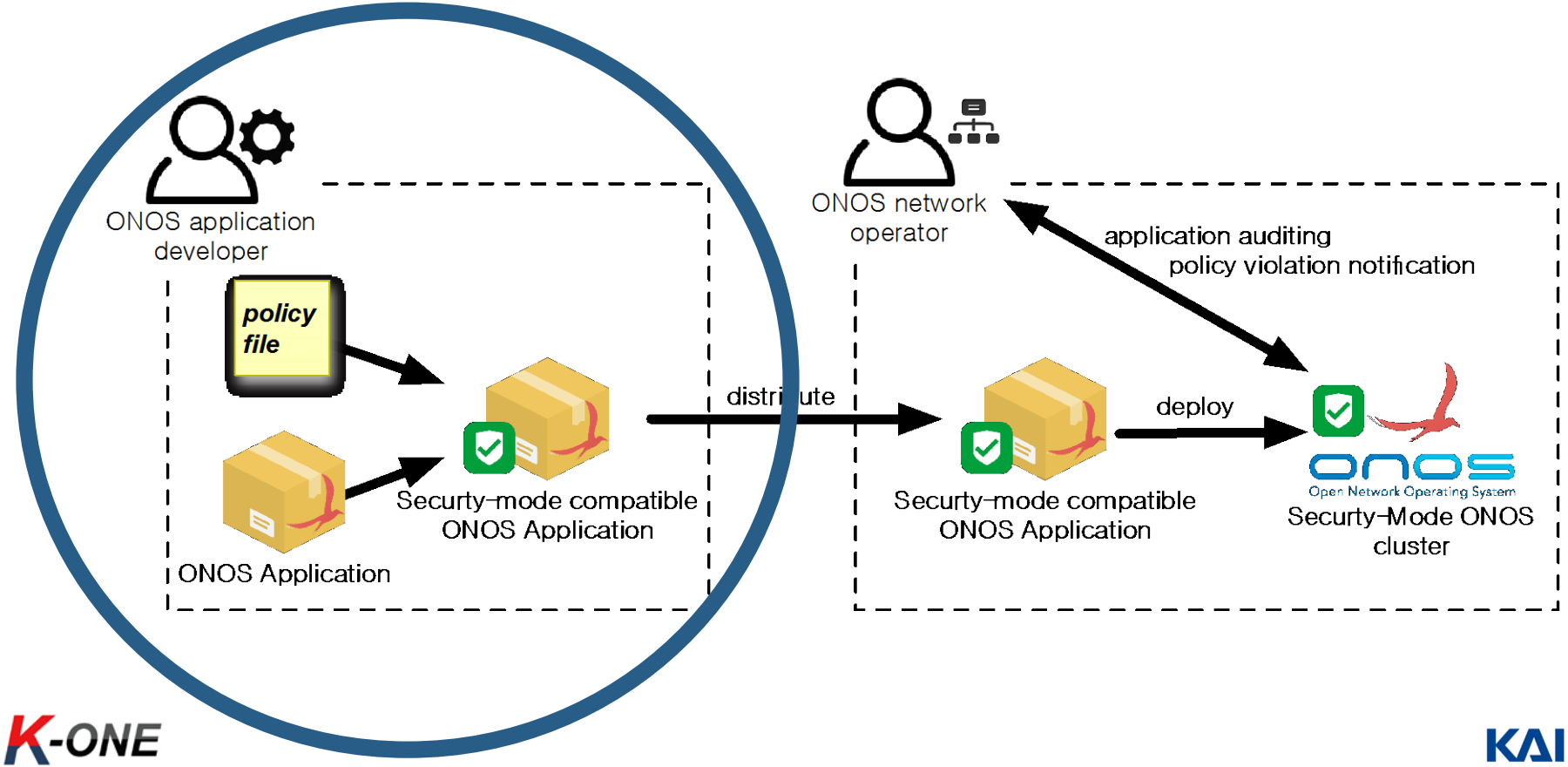
1) Mandatory application auditing prior to deployment

- Provide operators with explicit insight and control over the ONOS core services and APIs used by each ONOS application

2) Constraining application behavior at runtime

- A network application permission-enforce model for managing distributed ONOS applications

Workflow



K-ONE

KAIST

Security Policy File (Dev specified)

```
1  <security>
2      <role>USER</role>
3      <permissions>
4          <app-perm>DEVICE_READ</app-perm>
5          <app-perm>TOPOLOGY_READ</app-perm>
6          <app-perm>FLOWRULE_WRITE</app-perm>
7          <osgi-perm>
8              <classname>ServicePermission</classname>
9              <name>org.onosproject.demo.DemoAPI</name>
10             <actions>get,register</actions>
11         </osgi-perm>
12         <java-perm>
13             <classname>RuntimePermission</classname>
14             <name>modifyThread</name>
15         </java-perm>
16     </permissions>
17 </security>
```

ONOS Application role

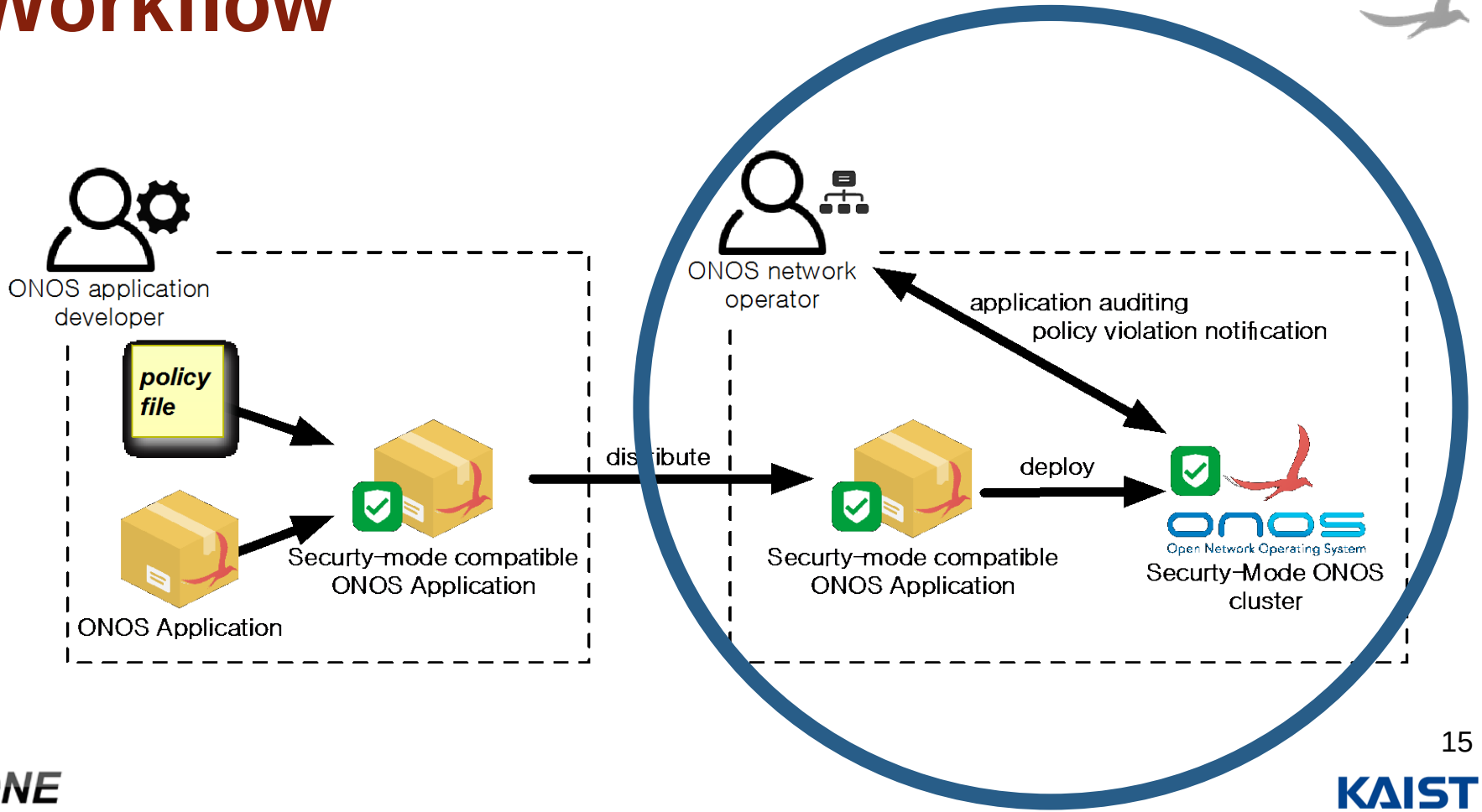
ONOS Application permissions

OSGi permissions

Java native permissions

Provides a clear understanding of application behavior

Workflow



K-ONE

KAIST

Mandatory Application Vetting

```
Mailing lists: lists.onosproject.org

Come help out! Find out how at: contribute.onosprojec

Hit '<tab>' for a list of available commands
and '[cmd] --help' for help on a specific command.
Hit '<ctrl-d>' or type 'system:shutdown' or 'logout'

onos> app activate org.onosproject.attack

*********************************
      SM-ONOS APP WARNING
*********************************
org.onosproject.attack has not been secured.
Please review before activating.
```

This application has **NOT** been **reviewed** and **approved** by an ONOS operator

Mandatory Application Vetting

```
onos> review org.onosproject.attack

******************************
       SM-ONOS APP REVIEW
******************************
Application name: org.onosproject.attack
Application role: USER

Developer specified permissions:
        [APP PERMISSION] HOST_EVENT
        [APP PERMISSION] DEVICE_READ
        [APP PERMISSION] FLOWRULE_WRITE
        [APP PERMISSION] INTENT_READ
        [APP PERMISSION] INTENT_WRITE
        [CLI SERVICE] org.apache.karaf.shell.console.CompletableFunction(register)
        [CLI SERVICE] org.apache.karaf.shell.commands.CommandWithAction(register)
        [CLI SERVICE] org.apache.felix.service.command.Function(register)
        [CLI SERVICE] org.osgi.service.blueprint.container.BlueprintContainer(register)
        [Other SERVICE] org.onosproject.attack.Attack(get,register)
        [SB SERVICE] org.onosproject.net.link.LinkProviderRegistry(get,register)
        [CRITICAL PERMISSION] RuntimePermission exitVM.0 ()

Permissions granted:
```

Must review **PERMISSIONS** before activating an app

Network admin may decide either to

1) Accept and grant the permissions
2) Reject and uninstall the app

```
onos> review org.onosproject.attack accept

******************************
       SM-ONOS APP REVIEW
******************************
Application name: org.onosproject.attack
Application role: USER

Developer specified permissions:
        [APP PERMISSION] HOST_EVENT
        [APP PERMISSION] DEVICE_READ
        [APP PERMISSION] FLOWRULE_WRITE
        [APP PERMISSION] INTENT_READ
        [APP PERMISSION] INTENT_WRITE
        [CLI SERVICE] org.apache.karaf.shell.console.CompletableFunction(register)
        [CLI SERVICE] org.apache.karaf.shell.commands.CommandWithAction(register)
        [CLI SERVICE] org.apache.felix.service.command.Function(register)
        [CLI SERVICE] org.osgi.service.blueprint.container.BlueprintContainer(register)
        [Other SERVICE] org.onosproject.attack.Attack(get,register)
        [SB SERVICE] org.onosproject.net.link.LinkProviderRegistry(get,register)
        [CRITICAL PERMISSION] RuntimePermission exitVM.0 ()

Permissions granted:
        [APP PERMISSION] INTENT_WRITE
        [APP PERMISSION] FLOWRULE_WRITE
        [APP PERMISSION] HOST_EVENT
        [APP PERMISSION] DEVICE_READ
        [APP PERMISSION] INTENT_READ
        [CLI SERVICE] org.apache.karaf.shell.console.CompletableFunction(register)
        [CLI SERVICE] org.apache.felix.service.command.Function(register)
        [CLI SERVICE] org.apache.karaf.shell.commands.CommandWithAction(register)
        [CLI SERVICE] org.osgi.service.blueprint.container.BlueprintContainer(register)
        [Other SERVICE] org.onosproject.attack.Attack(get,register)
        [SB SERVICE] org.onosproject.net.link.LinkProviderRegistry(get,register)
        [CRITICAL PERMISSION] RuntimePermission exitVM.0 ()
```

Network admin has agreed to grant the permissions to this application.

The security policy is enforced,
The admin may activate the app!

Runtime security policy violations

SM-ONOS blocks any attempt to violate security policy.

```
2016-03-11 03:22:34,260 | ERROR | l for user karaf | onos-app-attack                  | 181 - org.onosproject.onos-a
pp-attack - 1.5.0.SNAPSHOT | [org.onosproject.attack.AttackProvider(130)] The activate method has thrown an exceptio
n
java.security.AccessControlException: access denied ("org.osgi.framework.ServicePermission" "(service.id=1084)" "get
")
        at java.security.AccessControlContext.checkPermission(AccessControlContext.java:472)[:1.8.0_74]
        at java.security.AccessController.checkPermission(AccessController.java:884)[:1.8.0_74]
        at java.lang.SecurityManager.checkPermission(SecurityManager.java:549)[:1.8.0_74]
        at org.apache.felix.framework.Felix.getAllowedServiceReferences(Felix.java:3546)
```

It throws an **AccessControlException** upon at the time of violation.

Roadmap

Make more Secure ONOS controller !

What is ongoing work?

- Virtual network permission
- Automatic policy extraction tool
- Security policy enforcement on boot

What we will do?

- ONOS Application security-instrumentation framework
  - Static + dynamic analysis of ONOS App!

Any questions?

Demo available at KAIST booth!

Woojoong Kim
(woojoong@postech.ac.kr)

Mobile Networking Lab.
Pohang University of Science and Technology

MAS-Man : Mastership and Auto Scaling Management System in ONOS Controllers

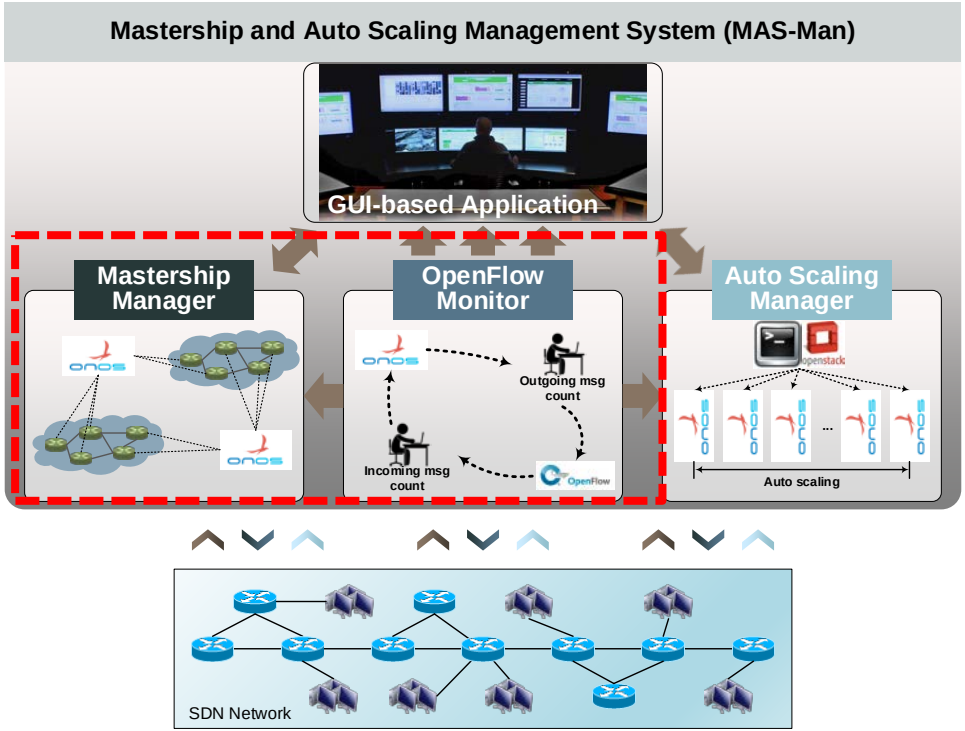
DevCon 2016

K-ONE

POSTECH

MAS-Man 이란?

Mastership and Auto Scaling Management System



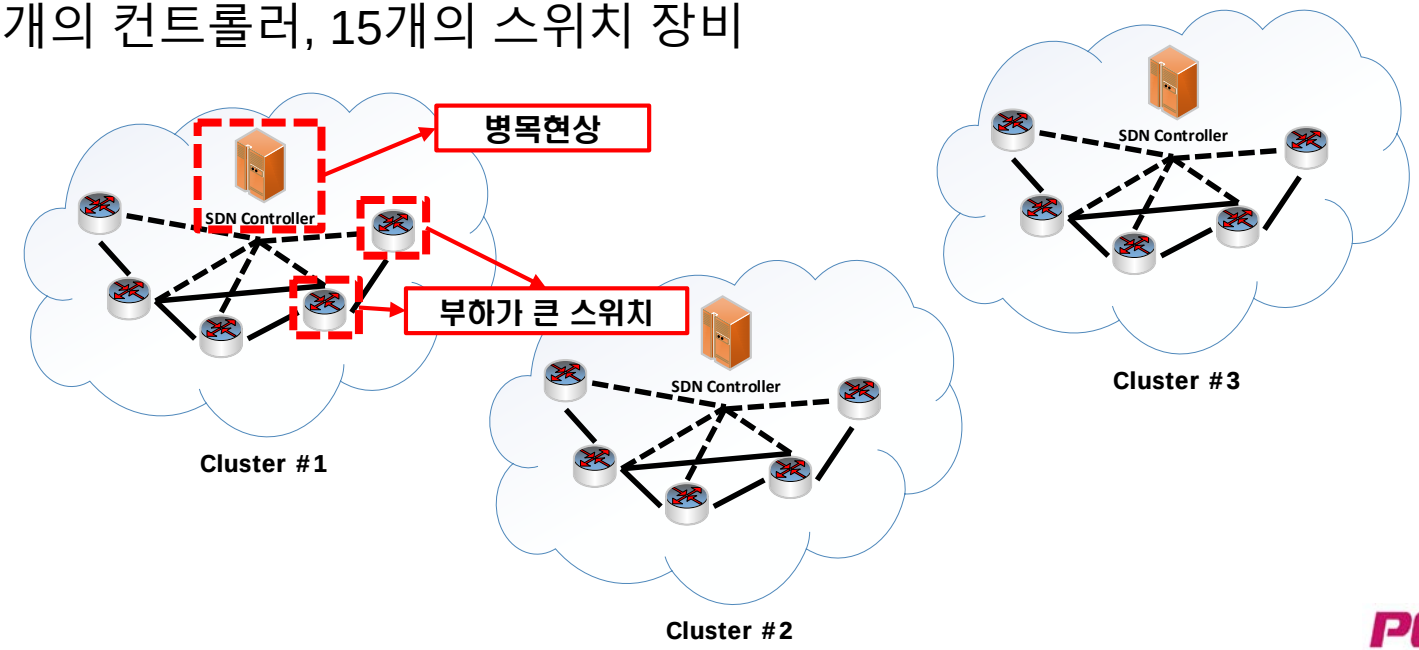
MAS-Man의 필요성

Mastership manager

기존 문제점

ONOS 상에서 네트워크 규모만을 고려하여 마스터쉽 수행

예시: 3개의 컨트롤러, 15개의 스위치 장비



MAS-Man의 필요성

스위치에서 발생되는 제어평면의
트래픽 양을 고려하여 마스터쉽 수행

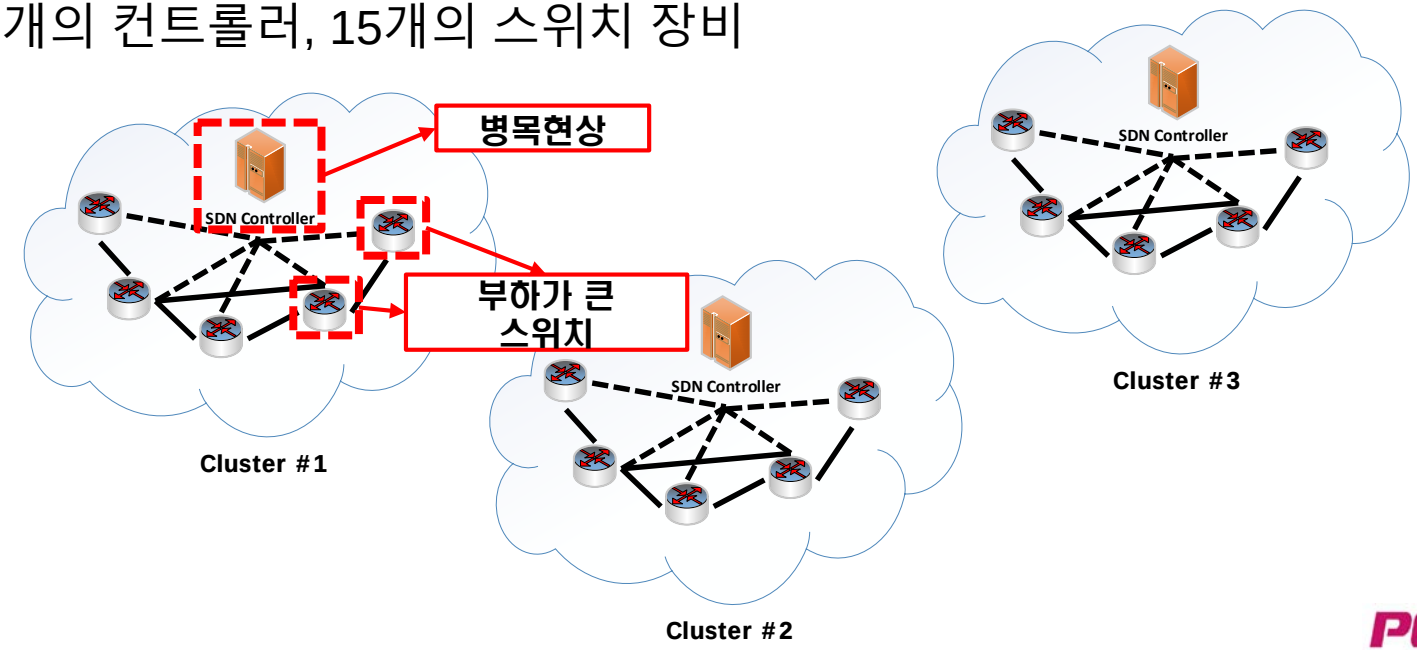
➔ 제어 평면 모니터와 새로운 마스터쉽 필요

Mastership manager

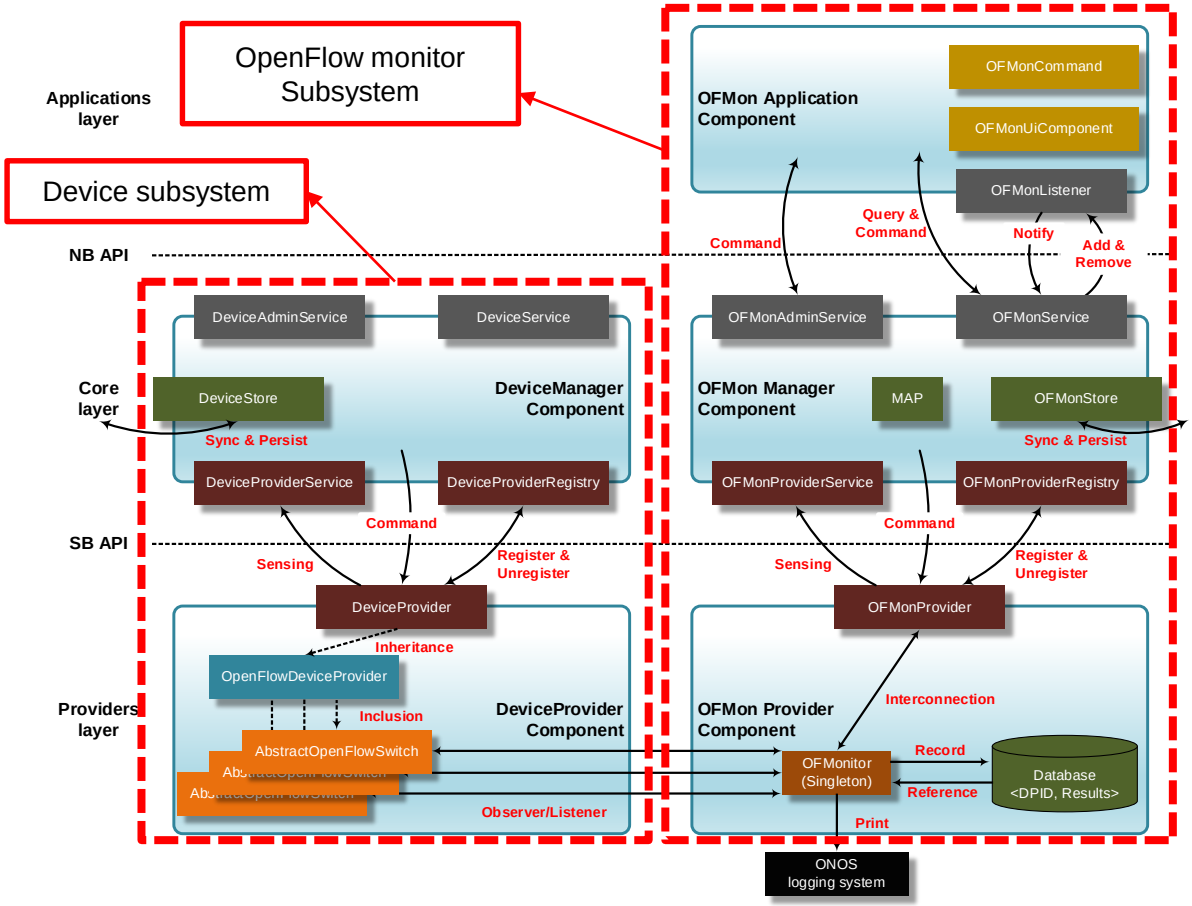
기존 문제점

ONOS 상에서 네트워크 규모만을 고려하여 마스터쉽 수행

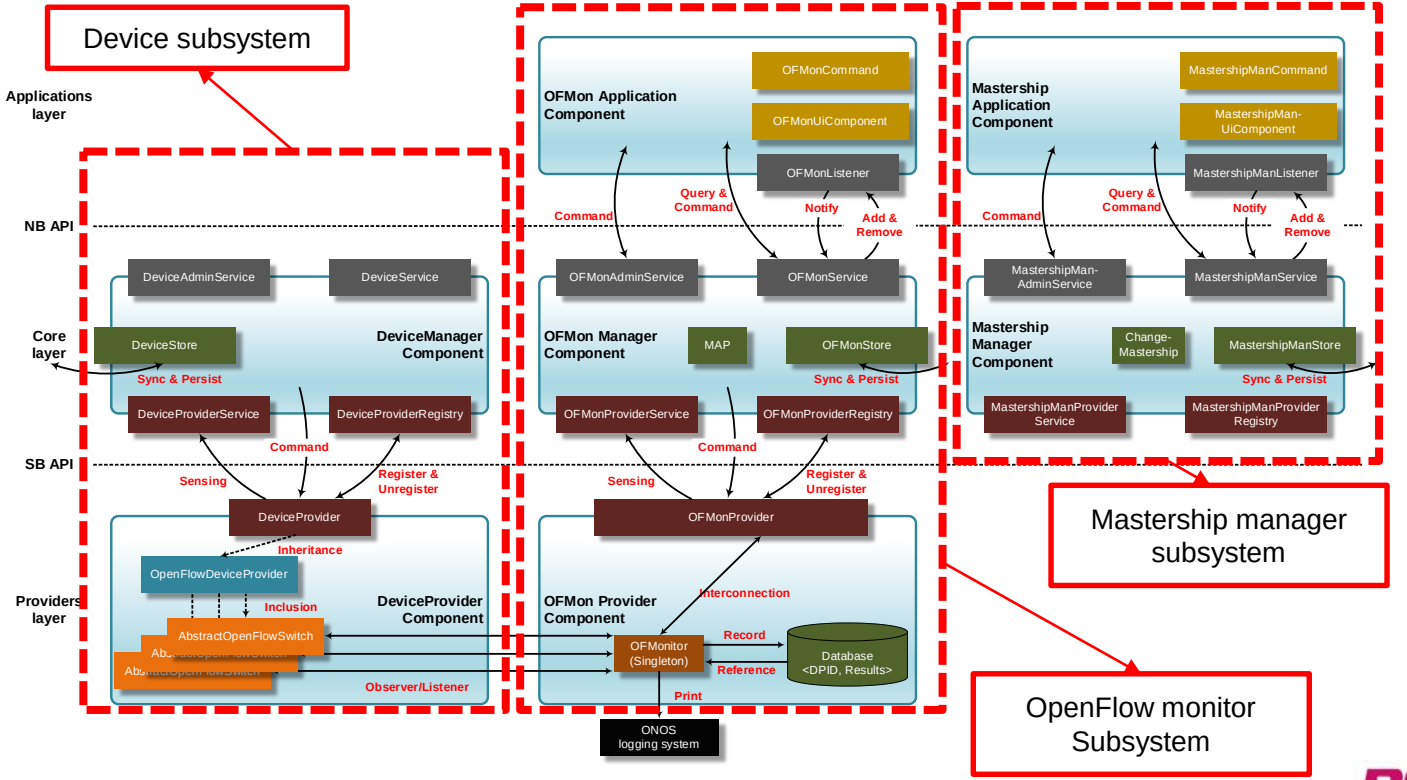
예시: 3개의 컨트롤러, 15개의 스위치 장비



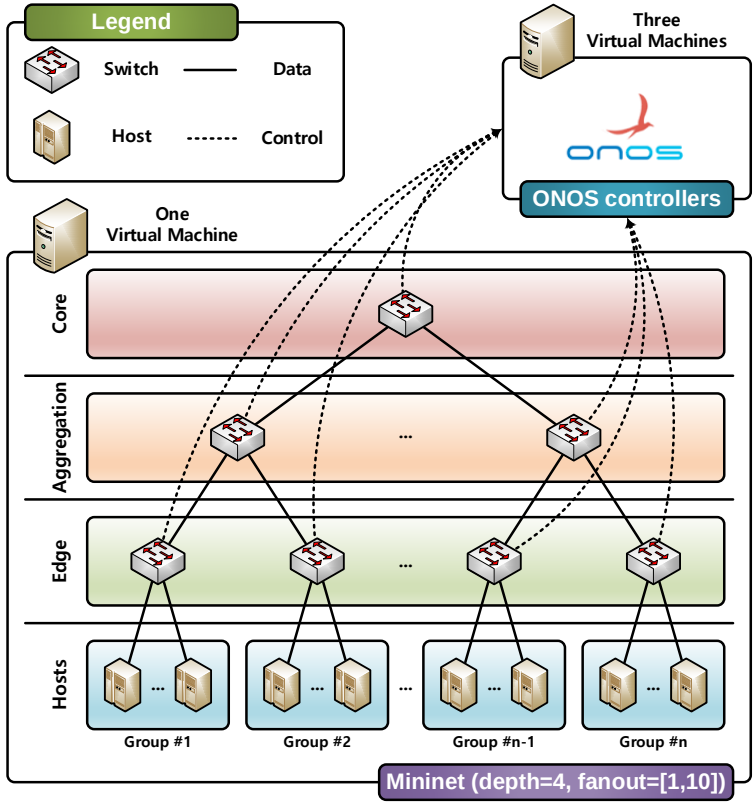
OpenFlow Monitor 구조



Mastership Manager 구조



시연 환경 구성



K-ONE

POSTECH

Seungho Ryu
(seunghoryu@postech.ac.kr)

Mobile Networking (MoNet) Lab.

Pohang University of Science and Technology (POSTECH)

Yoonseon Han
(seon054@postech.ac.kr)

Distributed Processing and Network Management (DPNM) Lab.

Pohang University of Science and Technology (POSTECH)

ONOS-LISP : Location/Identifier Separation Protocol (LISP) Subsystem Development for ONOS South-bound Interface

DevCon 2016

K-ONE

POSTECH

LISP 소개

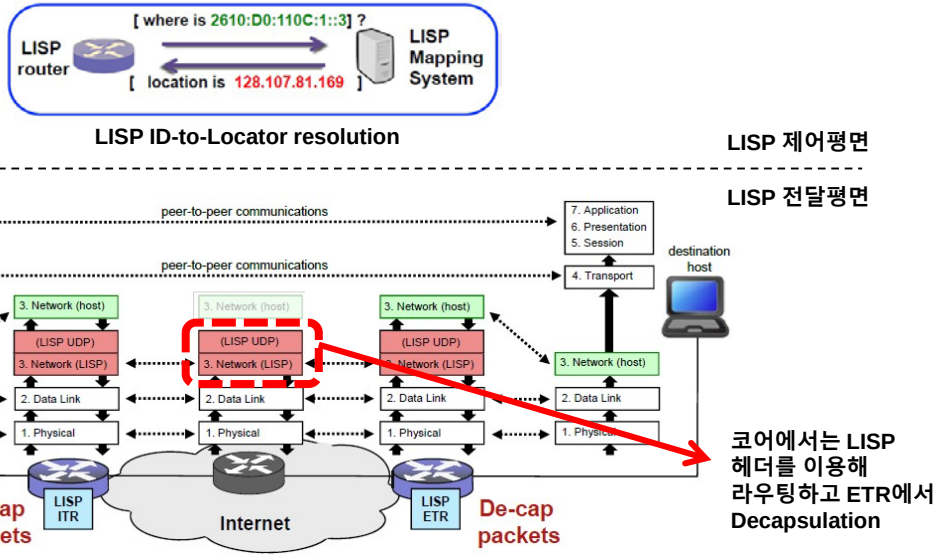
EID (End System ID): 단말에 부여된 IP 주소

RLOC (Routing Locator): 라우터에 부여된 IP주소

LISP Mapping System : EID-RLOC 매핑 정보 관리 및 분배 (DNS와 유사)

xTR : ITR과 ETR의 기능을 함께 수행하는 LISP 라우터를 지칭

- ITR (Ingress Tunnel Router): LISP Encapsulation 수행
- ETR (Egress Tunnel Router): LISP Decapsulation 수행



K-ONE

POSTECH

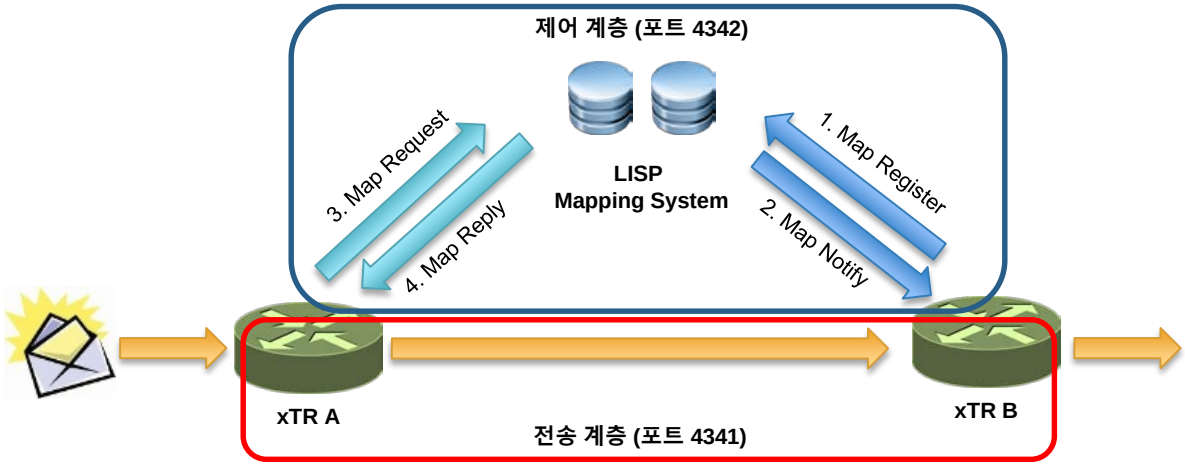
LISP의 제어 평면

LISP Mapping System

LISP Mapping System (MS)은 DNS와 유사하게 동작

- Map server : 각 xTR의 Mapping 정보를 수집/저장
- Map resolver : Mapping 정보 요청에 응답

정보 교환을 위한 제어 평면과 실제 데이터 전송을 위한 2개의 채널이 존재



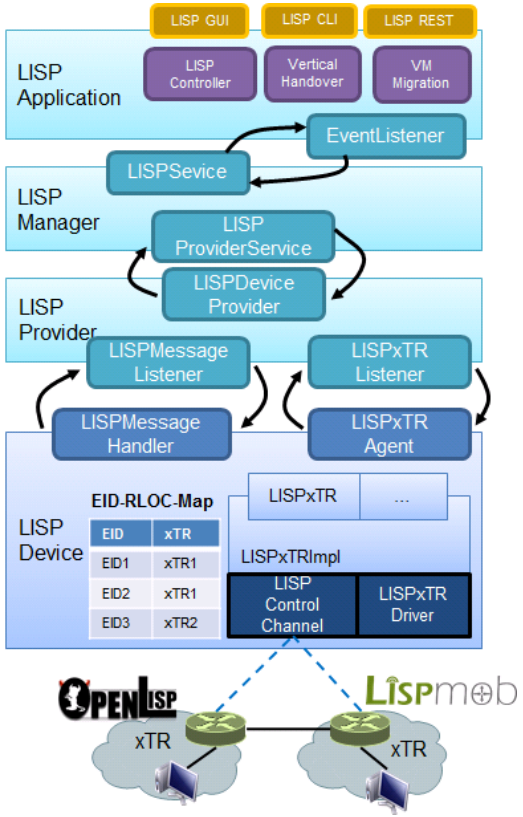
LISP subsystem 요구사항 및 설계

Mapping System 요구 사항

- xTR 상태 감시
- RTR 위치 저장 및 전달
- EID/RLOC Map 정보 저장
- MAC-SHA 1 인증 구현
- Nonce 저장
- 분산 정보 동기화

LISP controller 요구 사항

- 외부 API 지원
- LISP 장치 중앙 관리 메커니즘



K-ONE

POSTECH

LISP Provider 구현

LISP 메시지 송수신 기능

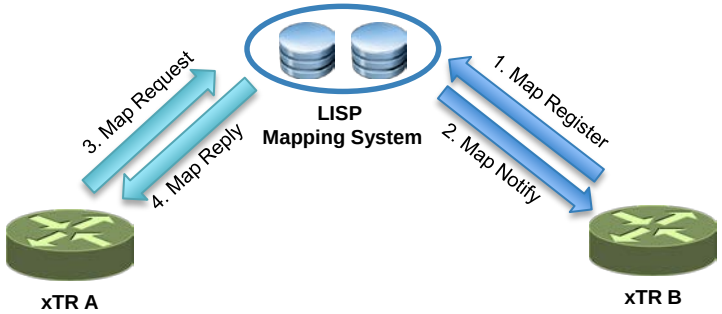
LISP Message Type : 1,2,3,4

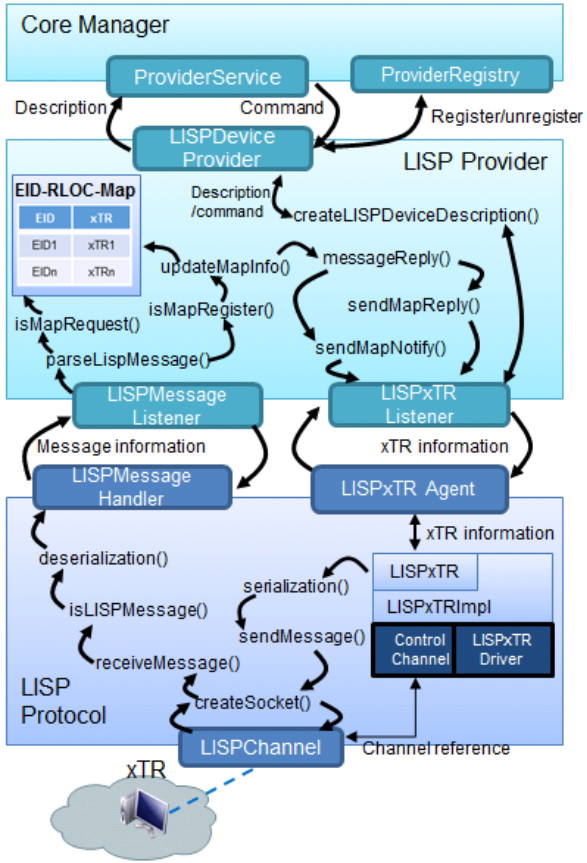
사용 채널 : 4342

새로운 xTR이 제공하는 EID/RLOC의 매핑을 저장 및 제공

Sender 측에서 생성하는 nonce의 저장

RLOC/EID 저장 DB 구현 필요





K-ONE

POSTECH

LISP API 설계 및 구현

전체 Mapping 참조

```
GET HTTP://LOCAHOST:8080/LISP/MAPPING/
```

EID 참조

```
GET HTTP://LOCAHOST:8080/LISP/MAPPING/?EID=1.1.1.1
```

RLOC 참조

```
GET HTTP://LOCAHOST:8080/LISP/MAPPING/?RLOC=141.223.1.1
```

매핑 정보 생성

```
POST HTTP://LOCAHOST:8080/LISP/MAPPING/
```

매핑 정보 갱신

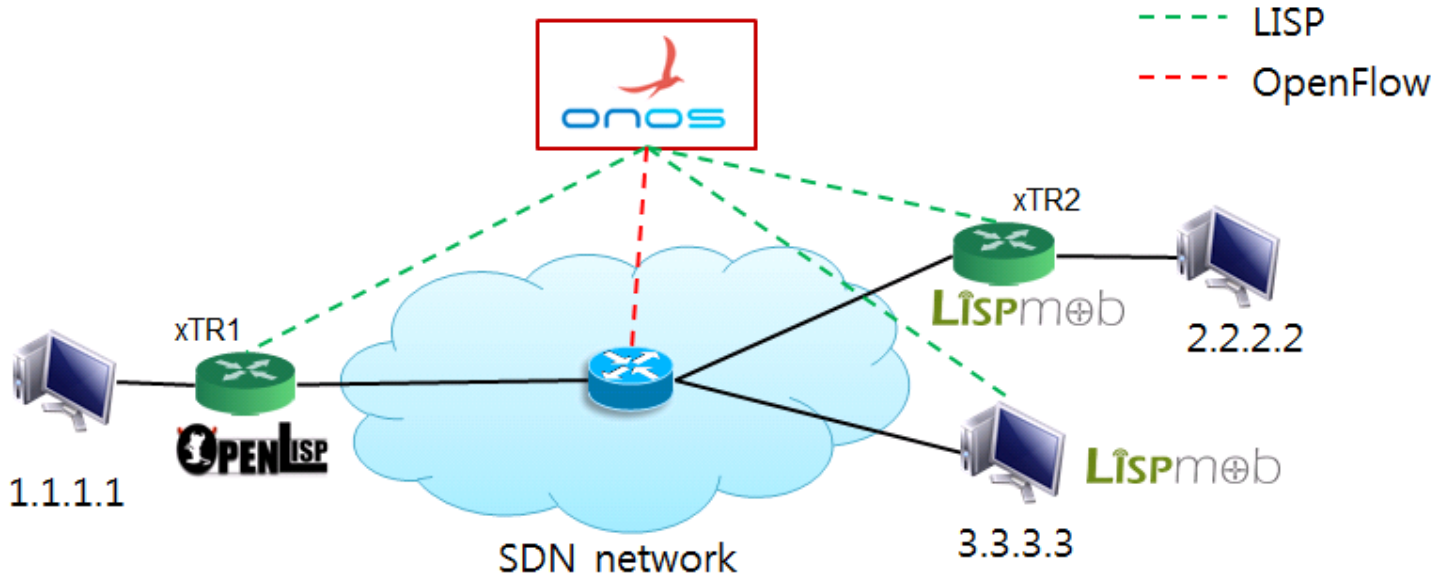
```
PUT HTTP://LOCAHOST:8080/LISP/MAPPING/?ID=12
```

매핑 정보 삭제

```
DELETE HTTP://LOCAHOST:8080/LISP/MAPPING/?ID=12
```

시연 계획

ONOS SDN 컨트롤러을 이용하여 LISP Site간 통신



K-ONE

POSTECH

POSTECH Roadmap

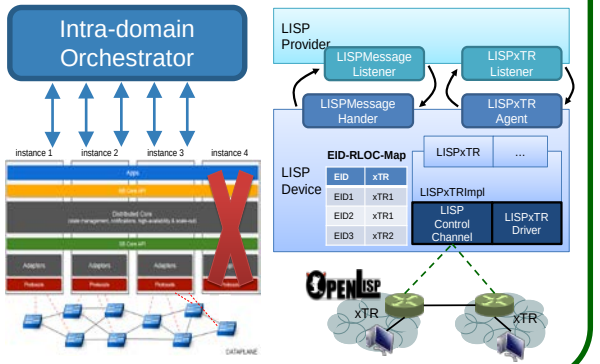
Now

1 Year

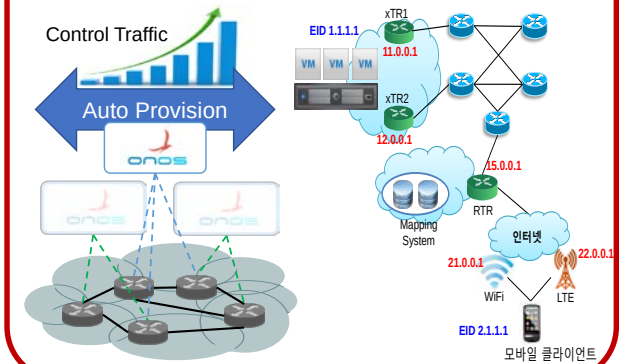
2 Years

5 Years

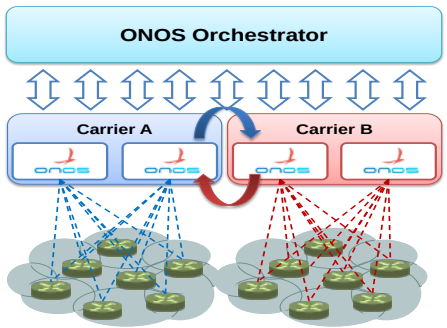
ONOS 클러스터 모니터링 및 클러스터 마스터쉽 관리 및 LISP 프로토콜 수용



네트워크 상황을 인지한 ONOS 자동 Provision 기능 개발 및 LISP Application 개발



다중 ONOS 마스터쉽 관리 및 캐리어 간 상호 운용을 위한 Inter-domain ONOS 인스턴스 관리 기술 개발



Any questions?

Demo available at POSTECH booth!

Jaeyoung So
(threeout56@gmail.com)

Embedded System (ESL) Lab.

KwangWoon University (KWU)

Hongsuk Kim
(quaqtuu@naver.com)

Embedded System (ESL) Lab.

KwangWoon University (KWU)

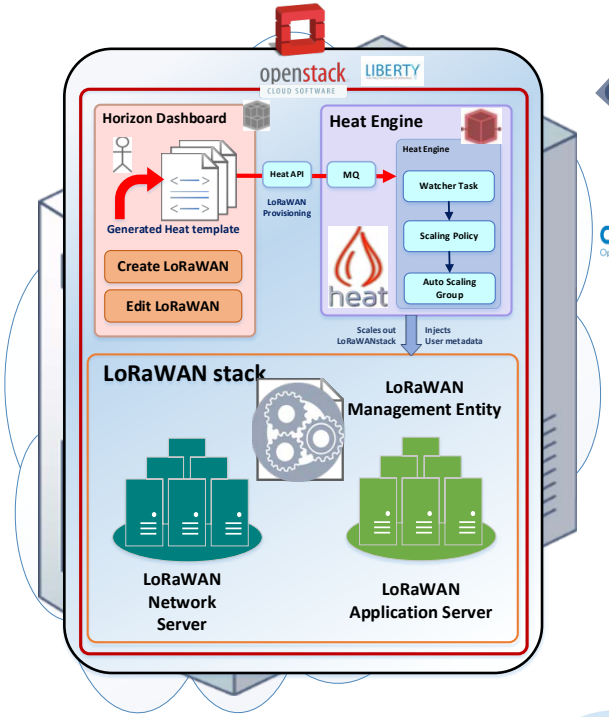
Provisioning of LoRaWAN Testbed

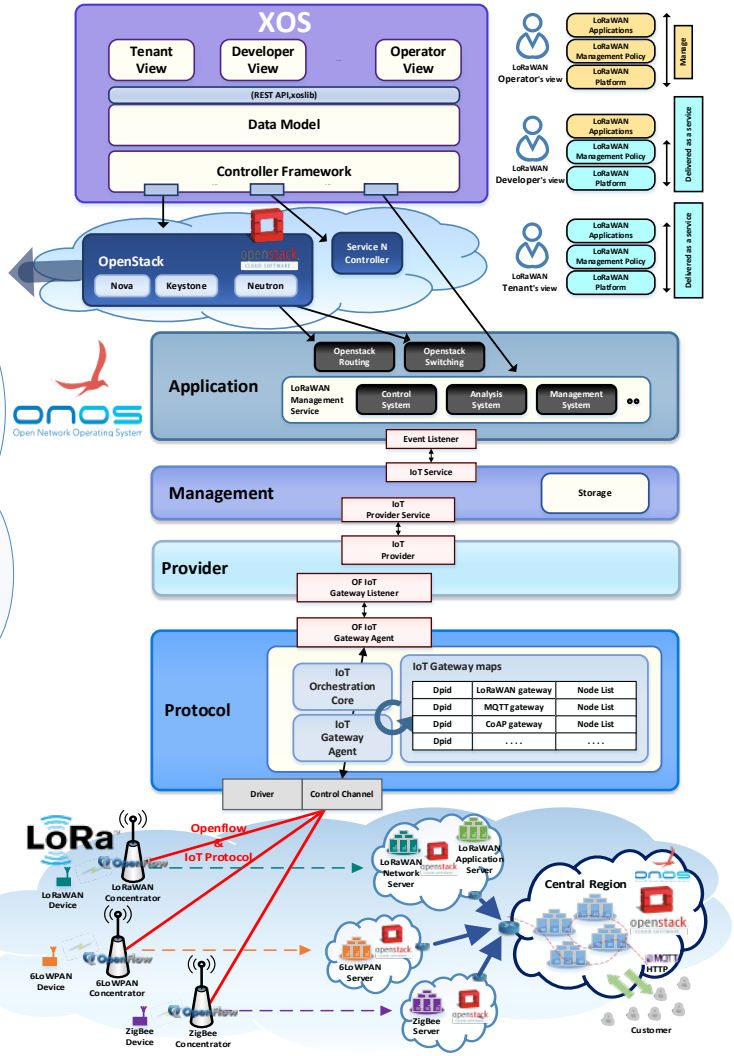
DevCon 2016

K-ONE

KWANGWOON UNIVERSITY

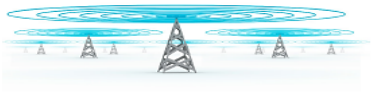
Roadmap


openstack
CLOUD SOFTWARE
LIBERTY
Horizon Dashboard
Generated Heat template
Create LoRaWAN
Edit LoRaWAN
Heat API
LoRaWAN Provisioning
Heat Engine
MQ
Heat Engine
Watcher Task
Scaling Policy
Auto Scaling Group
heat
Scales out LoRaWANstack
Injects User metadata
LoRaWAN stack
LoRaWAN Management Entity
LoRaWAN Network Server
LoRaWAN Application Server


XOS
Tenant View
Developer View
Operator View
(REST API,xoslib)
Data Model
Controller Framework
LoRaWAN Operator's view
LoRaWAN Developer's view
LoRaWAN Tenant's view
LoRaWAN Applications
LoRaWAN Management Policy
LoRaWAN Platform
Manage
Delivered as a service
OpenStack
Nova
Keystone
Neutron
Service N Controller
onos
Open Network Operating System
Application
Openstack Routing
Openstack Switching
LoRaWAN Management Service
Control System
Analysis System
Management System
Event Listener
IoT Service
Management
Storage
IoT Provider Service
IoT Provider
Provider
OF IoT Gateway Listener
OF IoT Gateway Agent
Protocol
IoT Orchestration Core
IoT Gateway Agent
IoT Gateway maps
Dpid
LoRaWAN gateway
MQTT gateway
CoAP gateway
Node List
Driver
Control Channel
LoRa
LoRaWAN Device
LoRaWAN Concentrator
Openflow
IoT Protocol
LoRaWAN Network Server
LoRaWAN Application Server
Central Region
6LoWPAN Device
6LoWPAN Concentrator
6LoWPAN Server
ZigBee Device
ZigBee Concentrator
ZigBee Server
MQTT
HTTP
Customer

LoRaWAN

- **Introduction of Long Range Wide Area Network**
  - LoRa는 저전력 장거리 통신에 적절한 RF 변조 기술
  - LoRaWAN 표준은 2015년 2월에 결성된 LoRa Alliance에서 제정된 저전력 장거리 무선 통신 프로토콜
  - LoRaWAN에서 사용되는 단말은 구조가 간단하여 배터리 수명이 늘어남
  - 망 구축 비용이 줄어들고 ISM 대역을 사용함으로써 망 사용료에 대한 부담 해소
  - LoRaWAN은 양방향 통신으로 스마트 시티, 센서 네트워크 산업 자동화 어플리케이션 등 여러 분야에 활용될 수 있음



Long Range

- Greater than cellular
- Deep indoor coverage
- Star topology

Max Lifetime

- Low power optimized
- 10-20yr lifetime
- >10x vs cellular M2M

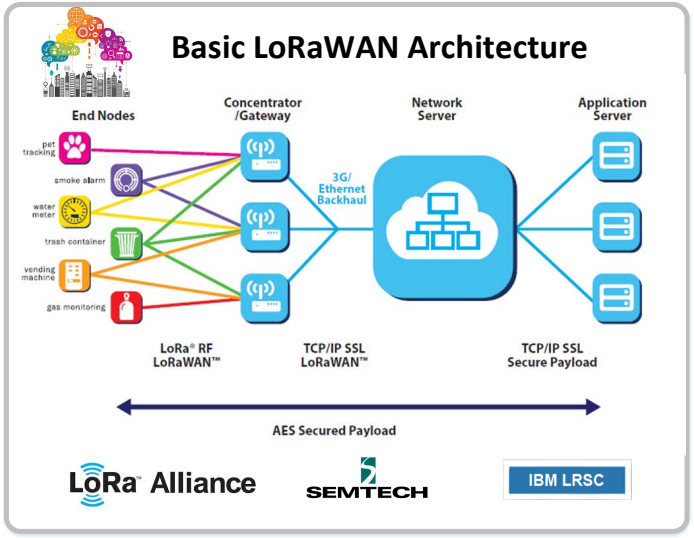
Multi-Usage

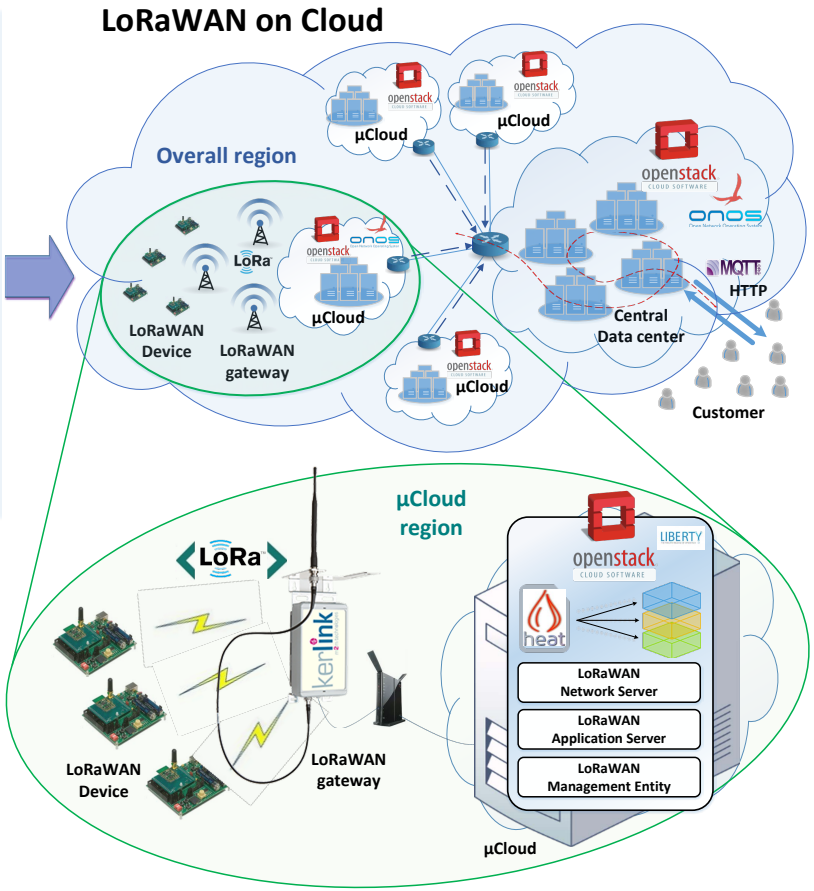
- High capacity
- Multi-tenant
- Public network

Low Cost

- Minimal infrastructure
- Low cost end node
- Open SW

Provisioning of LoRaWAN



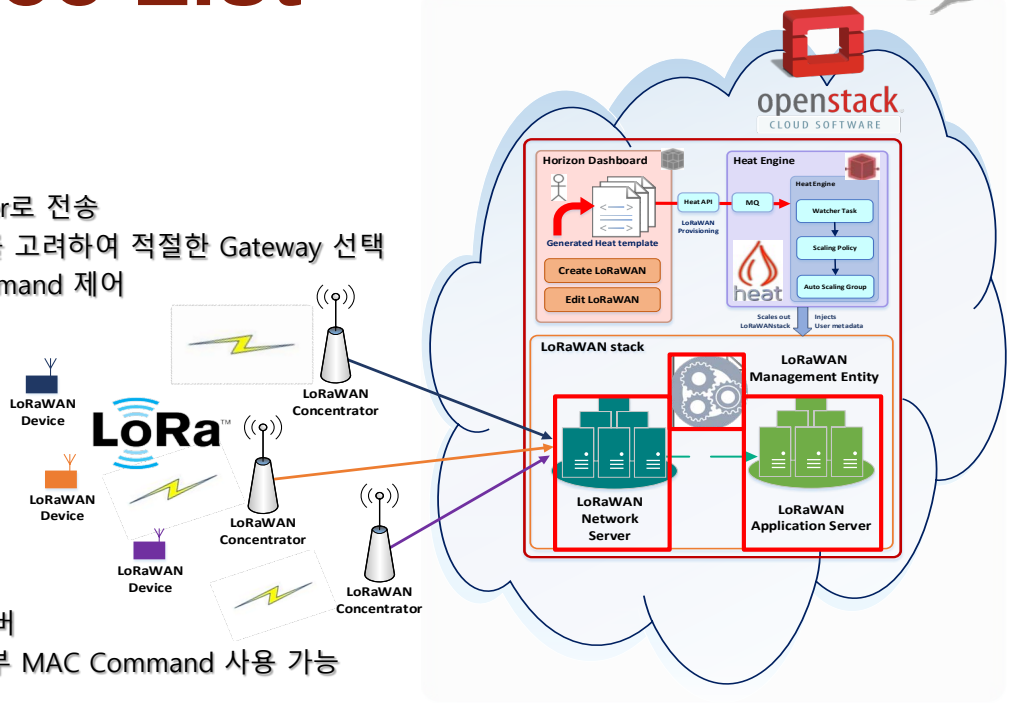


LoRaWAN Provisioning

- Horizon을 통해 LoRaWAN Provisioning이 가능한 환경 설정
- Heat를 활용한 LoRaWAN 인스턴스 생성 및 수정
- Management entity를 통해 인스턴스간 통합 연결 관리

LoRaWAN Instance List

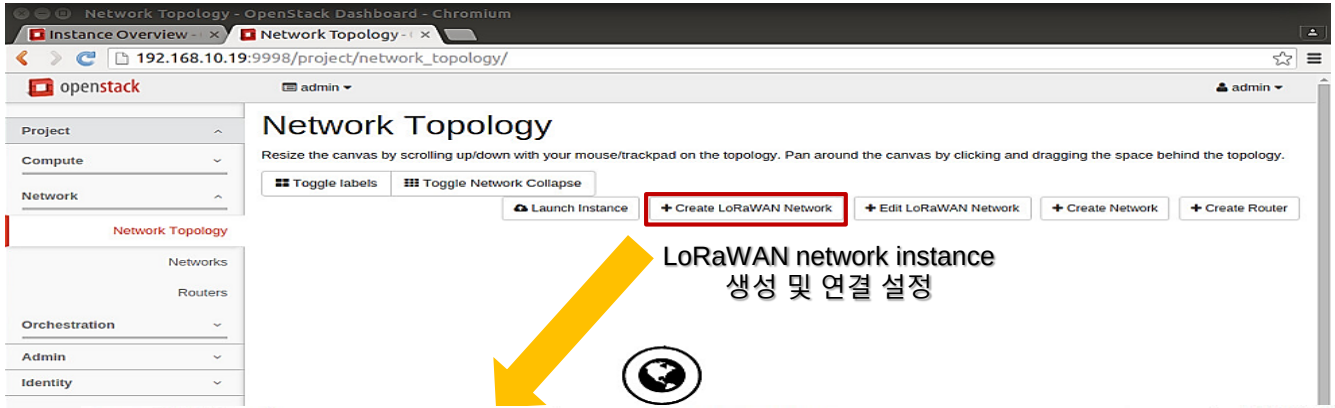
- **Network Server**
  - Device 인증 (Over-The-Air Activation)
  - End device의 데이터의 암호화 및 복호화
  - End device가 보낸 데이터를 적절한 Application server로 전송
  - End device로 메시지를 전송하기 위해 무선통신상태를 고려하여 적절한 Gateway 선택
  - End device의 통신 상태에 따른 LoRaWAN MAC Command 제어
  - Gateway 부하 및 상태 관리



- **Application Server**
  - 해당하는 어플리케이션의 데이터 시각화
  - 웹페이지 이용하여 사용자에게 서비스를 제공하는 서버
  - LoRaWAN MAC command에 관련된 API제공하여 일부 MAC Command 사용 가능

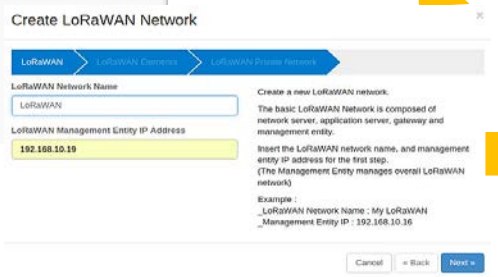
- **LoRaWAN Controller**
  - Horizon의 LoRaWAN 인터페이스를 통해 생성된 인스턴스들간의 연결을 관리
  - Network server는 LoRaWAN Controller로 부터 받은 Gateway , Application Server들만 접속을 허용

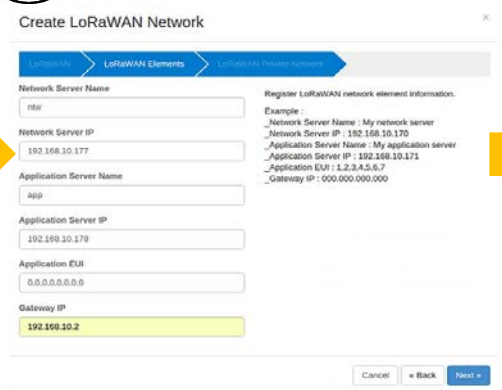
Create LoRaWAN Network

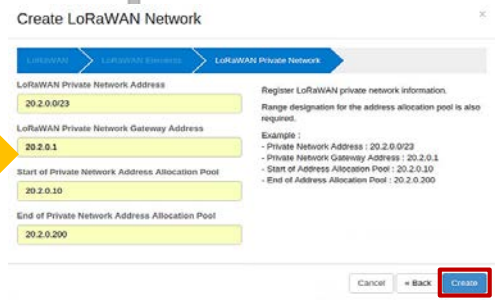


LoRaWAN network instance
생성 및 연결 설정

필요한 항목들을 작성 후,
Create버튼을 통해
LoRaWAN Network를 생성





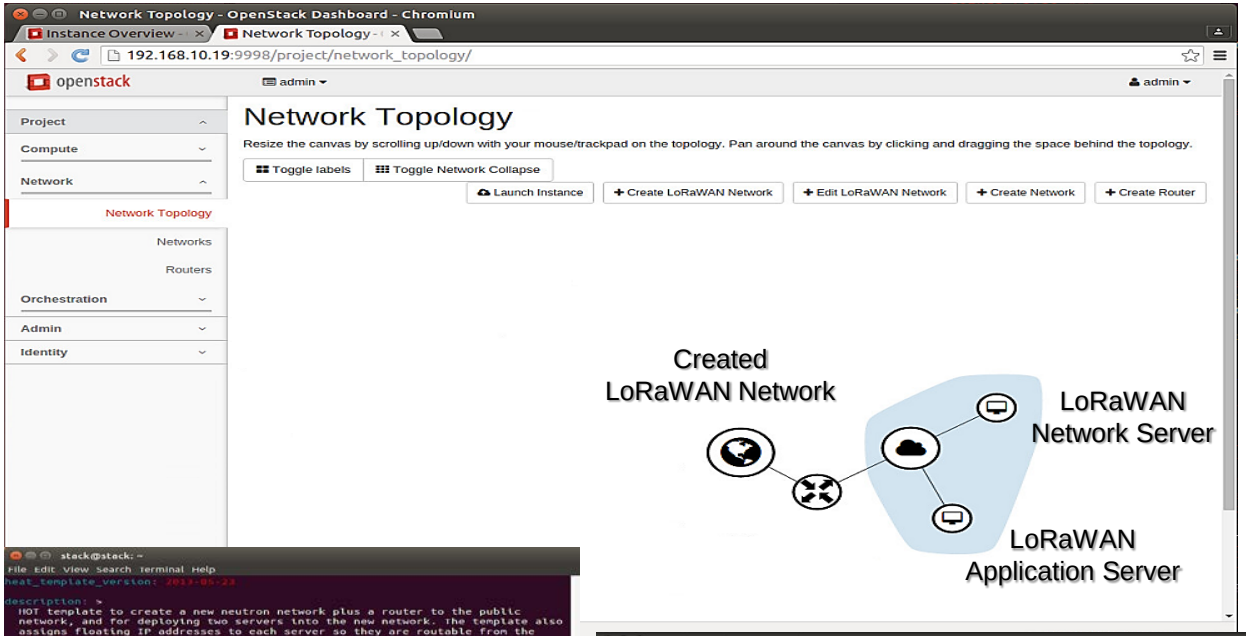


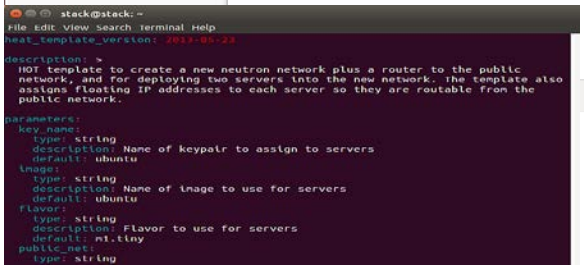
- LoRaWAN Controller IP 설정 - **- Network server, Application server, Gateway 설정 -** **- LoRaWAN private Network 설정-**

K-ONE

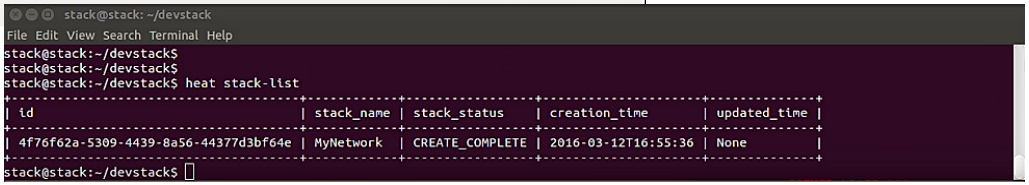
KWANGWOON UNIVERSITY

Create LoRaWAN Network





- LoRaWAN Heat template -



- 생성된 Heat stack -

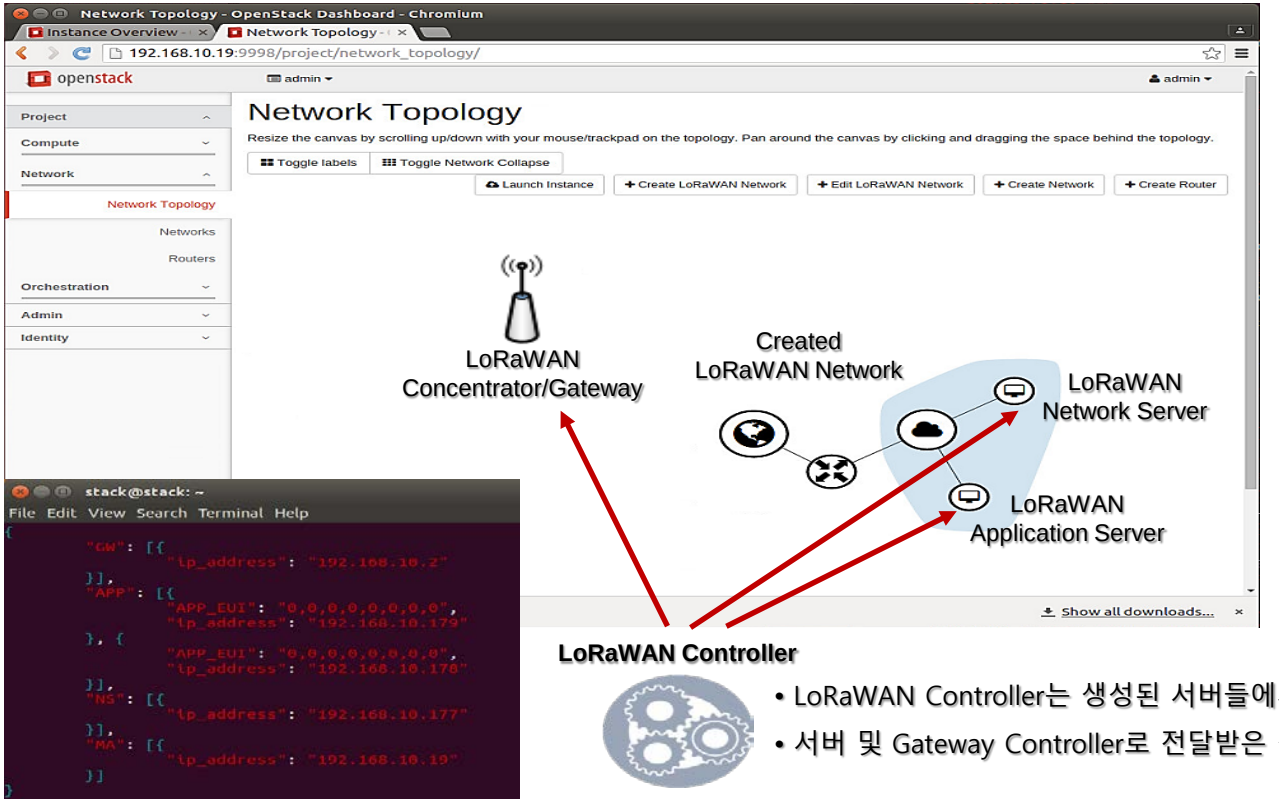
Horizon 을 통해 수집된 정보들을 토대로
Heat template을 생성 후,
LoRaWAN에 대한 Heat stack을 생성

Template내 각 인스턴스들 부팅 완료 시,
서버를 자동으로 구동하도록 script 추가

K-ONE

KWANGWOON UNIVERSITY

Create LoRaWAN Network



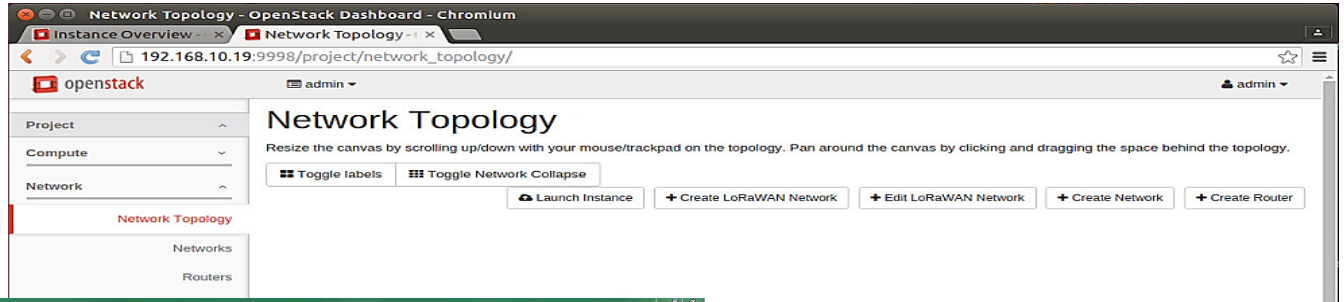
- LoRaWAN Controller는 생성된 서버들에게 연결할 목록 전달
- 서버 및 Gateway Controller로 전달받은 정보를 토대로 연결

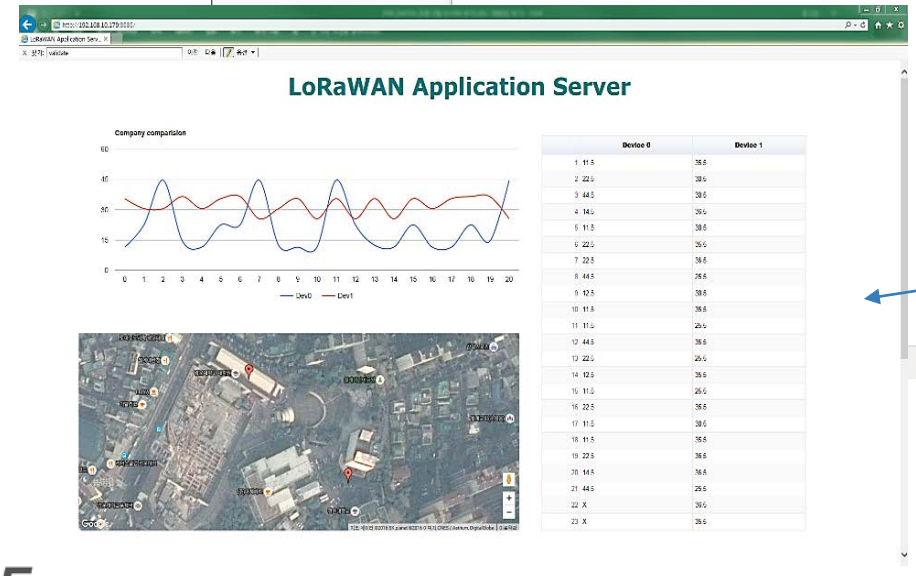
- Horizon을 통해 생성된 LoRaWAN 정보가 담긴 JSON -

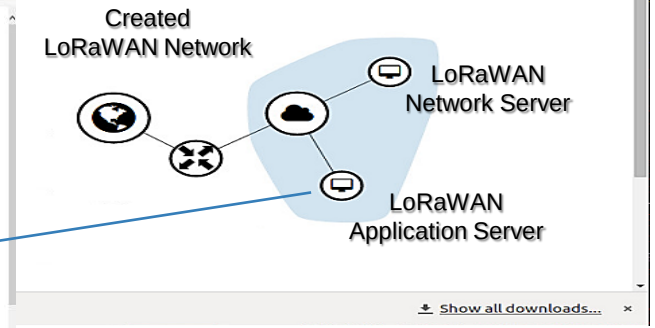
K-ONE

KWANGWOON UNIVERSITY

Create LoRaWAN Network





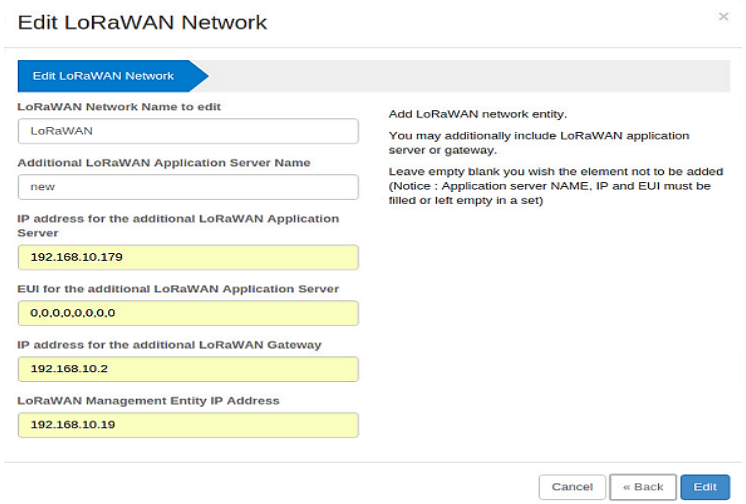


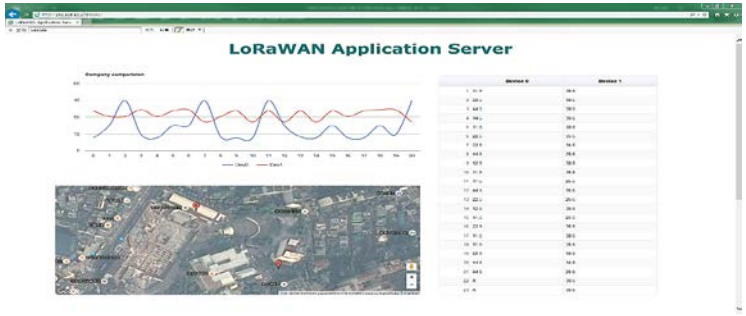
LoRaWAN network 생성 후,
Application server가 수신하는 Device의 데이터 확인

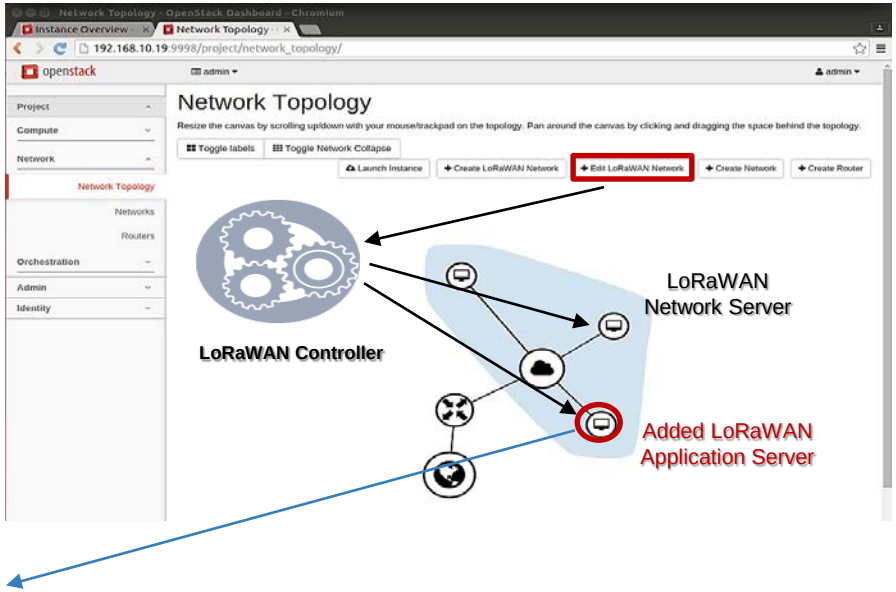
K-ONE

KWANGWOON UNIVERSITY

Edit LoRaWAN Network







- Edit LoRaWAN 인터페이스를 통해 Application server 추가 가능
- 마찬가지로 LoRaWAN Controller의해 서버 간 연결이 자동으로 이뤄짐
- 추가된 Application server를 통해 해당 어플리케이션 device의 데이터 수신 확인

K-ONE

KWANGWOON UNIVERSITY

Any questions?

Demo available at KWU booth!

K-ONE

KWANGWOON UNIVERSITY