I2NSF Framework Project @ IETF-103 Hackathon

IETF 103, Bangkok

November 4, 2018

Champion: Jaehoon Paul Jeong
pauljeong@skku.edu
Sungkyunkwan University

I2NSF Framework: What Did We Do This Time?

- I2NSF: Data-Driven Security Policy Enforcement
  - We implemented I2NSF Framework on top of OpenStack with NFV Reference Architecture.

- This work is an Open Source Project!
  - 8 graduate students (Sungkyunkwan University) and 4 graduate students (Soongsil University)
  - 4 professors (Sungkyunkwan, Soongsil, and Chosun Universities)
  - 2 researchers (ETRI and KT Corporation)
  - Open Source Code on Github
    - https://github.com/kimjinyong/i2nsf-framework/tree/master/Hackathon-103

IETF I2NSF (Interface to Network Security Functions) Working Group: I2NSF Framework Project

Champion: Jaehoon Paul Jeong (SKKU)

Professors

- Jaehoon (Paul) Jeong (Sungkyunkwan)
- Hyoungshick Kim (Sungkyunkwan)
- Younghan Kim (Soongsil)
- Sangwon Hyun (Chosun)

Collaborators

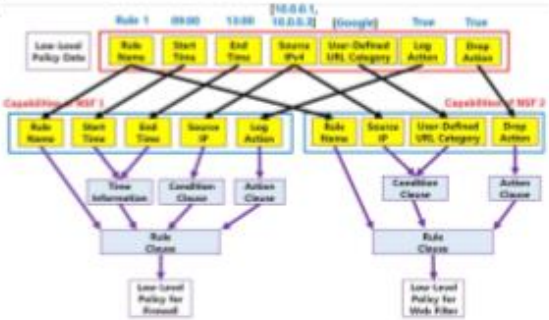
- Jung-Soo Park (ETRI)
- Tae-Jin Ahn (Korea Telecom)

Students

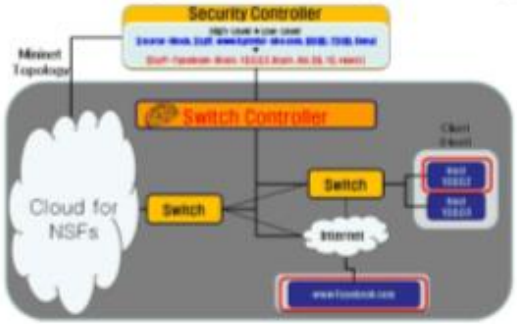
- Jinyong (Tim) Kim
- Eunsoo Kim
- Dongjin Hong
- Tae-Kyun Roh
- Sarang Wi
- Seungjin Lee
- Jinhyuk Yang
- Jaehong Jeong
- Hyunsik Yang
- Kyoungjae Sun
- Jaewook Oh
- Xianjun Hong

I2NSF Architecture in NFV Reference

Security Policy Translation



Network Security Functions (NSF) -Triggered Steering



Where to get code

- Github – Source code
  - ✓ https://github.com/kimjinyong/i2nsf-framework

What to pull down to set-up environment

- OS: Ubuntu 14.04TL
- ConfD for NETCONF: 6.2 Version
- Apache2: 2.4.7 Version
- MySQL: 14.14 Version
- PHP: 5.5.9 Version
- OpenStack: Networking-SFC, Tacker
- Jetconf: Python Open API for RESTCONF

Manual for Operation Process

- Detailed descriptions about operation process in README.txt (can be found in the VM image)

Contents of Implementation

- I2NSF Framework for provisioning Network Security Functions (NSFs)
  - ✓ Consumer-Facing Interface via RESTCONF/YANG
  - ✓ NSF-Facing Interface via NETCONF/YANG
  - ✓ Registration Interface via NETCONF/YANG
  - ✓ I2NSF Framework in NFV Environment using OpenStack (New Feature)
- Network Security Functions
  - ✓ Firewall using SDN and Suricata
  - ✓ Mail-filter and Web-filter using Suricata
- Advanced Functions
  - ✓ Security Policy Translation
  - ✓ I2NSF Policy Provisioning
  - ✓ NSF-triggered Traffic Steering using OpenStack Networking-SFC (New Feature)

Goal of IETF-103 I2NSF Framework Project

- **Integration of IETF I2NSF Framework and ETSI NFV Reference Architecture**
  1. **Design of I2NSF Framework in NFV Reference Architecture**
  2. **Dynamic Creation of Virtual NSFs according to Security Policy using OpenStack Tacker**
  3. **Service Function Chaining for Traffic Steering using OpenStack Networking-SFC**

Limitations and Challenges of I2NSF Project

❖ **Limitations** of the Previous I2NSF Open Source

- The I2NSF Framework was implemented in Mininet for Proof of Concept (POC).
- So, it could not be used in the real world.

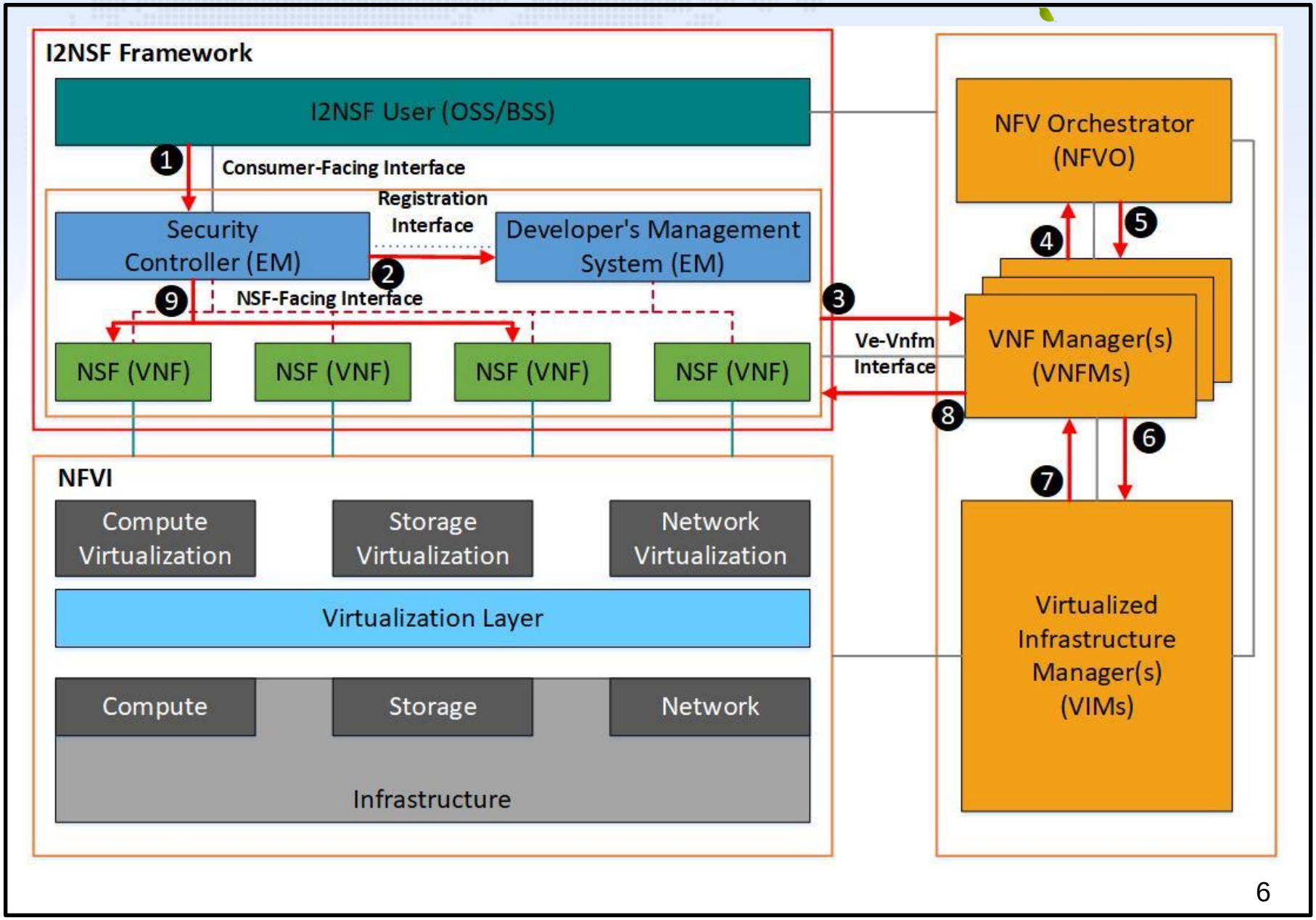
❖ **Challenges** of I2NSF Framework Project

1. Design: How to place the elements of I2NSF Framework in NFV Environment (NFV-I2NSF)?
   - I2NSF User, Security Controller, Developer's Management System, and Network Security Functions

2. Dynamic Creation of Virtual NSFs in NFV-I2NSF
   - Procedure of Security Policy Translation and Enforcement

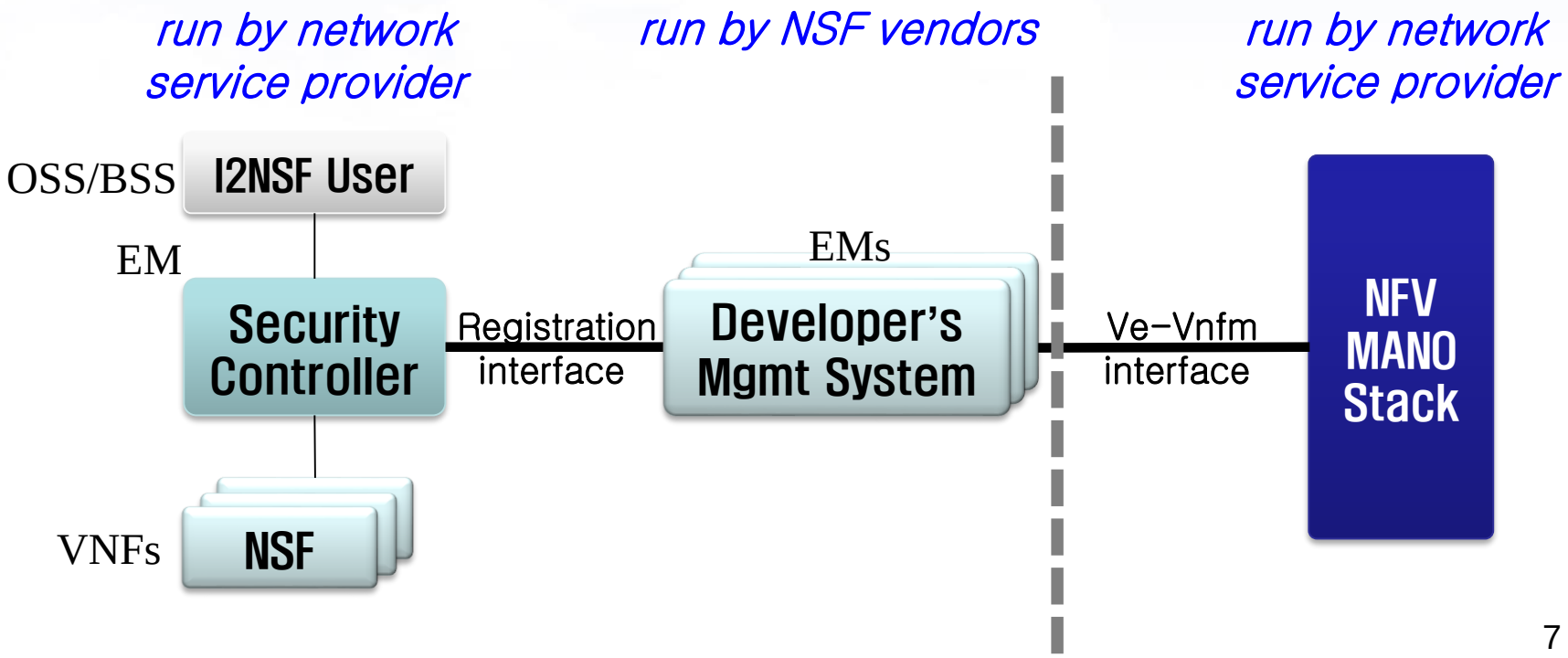
3. Service Function Chaining in NFV-I2NSF
   - Traffic Steering Control for Security Policy in NFV-I2NSF

I2NSF Framework in NFV Reference Architecture



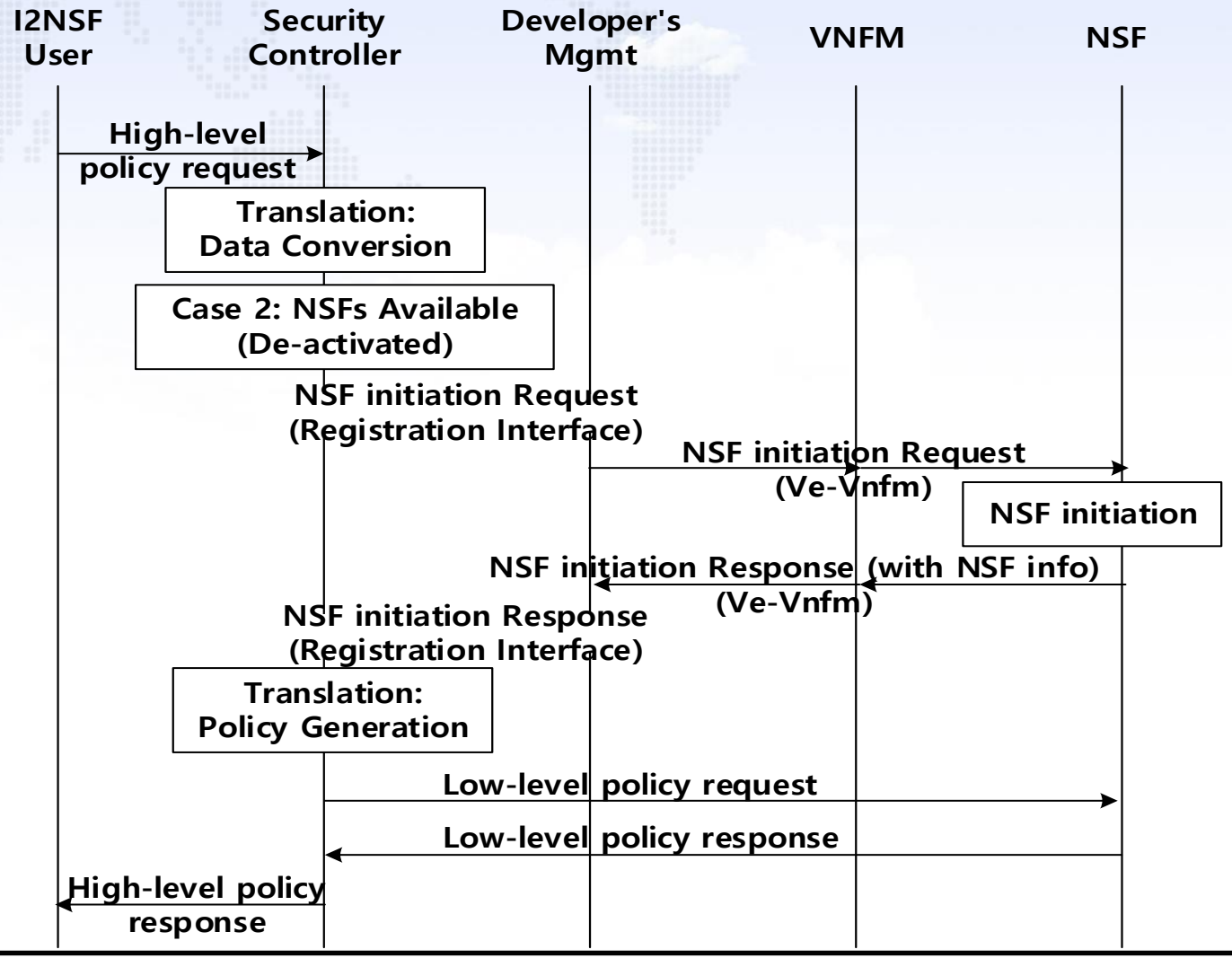
I2NSF Framework in NFV Reference Architecture: NFV-I2NSF System

- Management Subjects and Objects in NFV-I2NSF
  - **Network Service Provider**
    - ✓ I2NSF User, Security Controller, NSFs, and NFV MANO
  - **NSF Vendors**
    - ✓ Developer's Management System



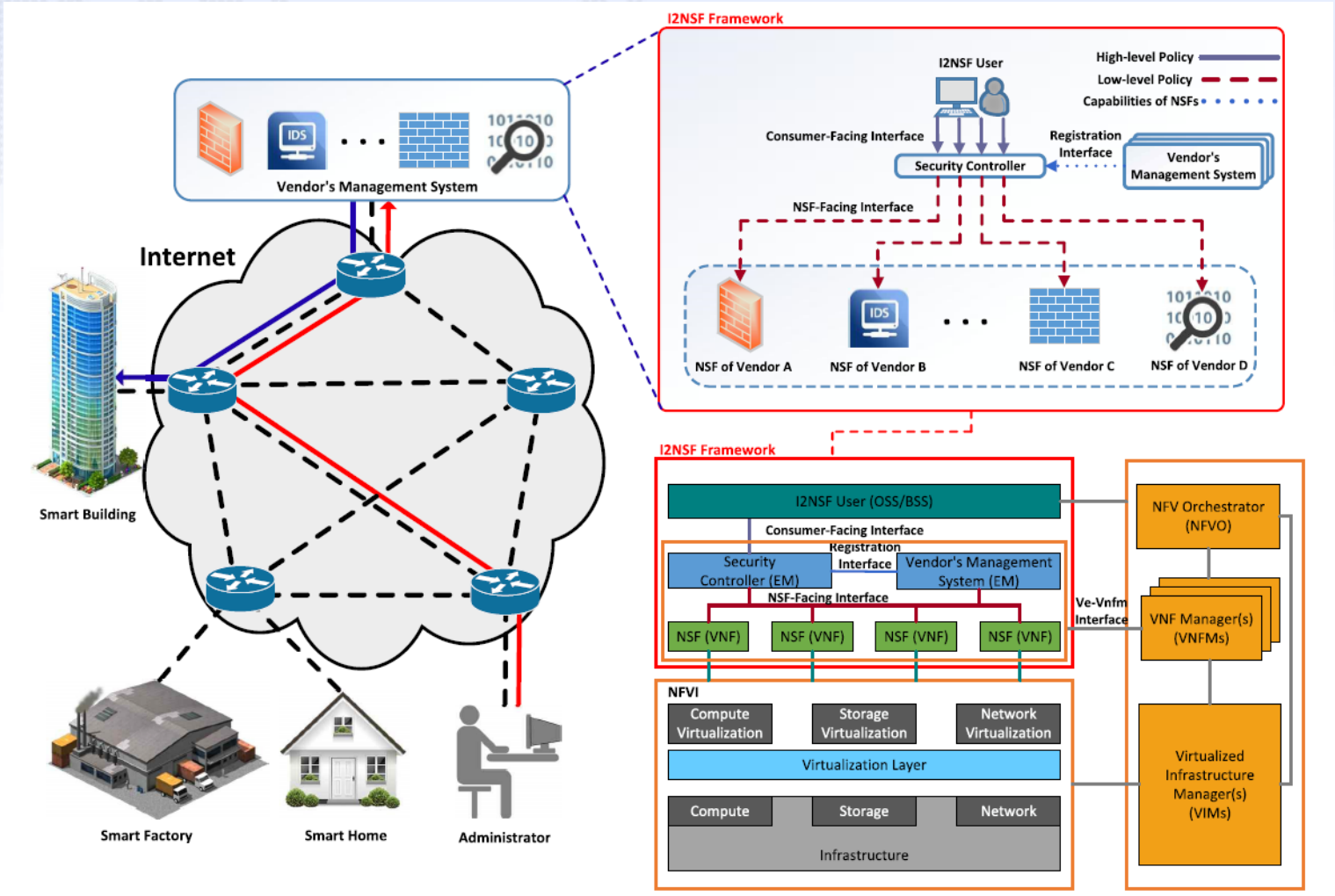
I2NSF Operations in NFV-I2NSF System

- Procedure of Security Policy Enforcement in NFV-I2NSF System (Case: NSF is De-activated in NFV)



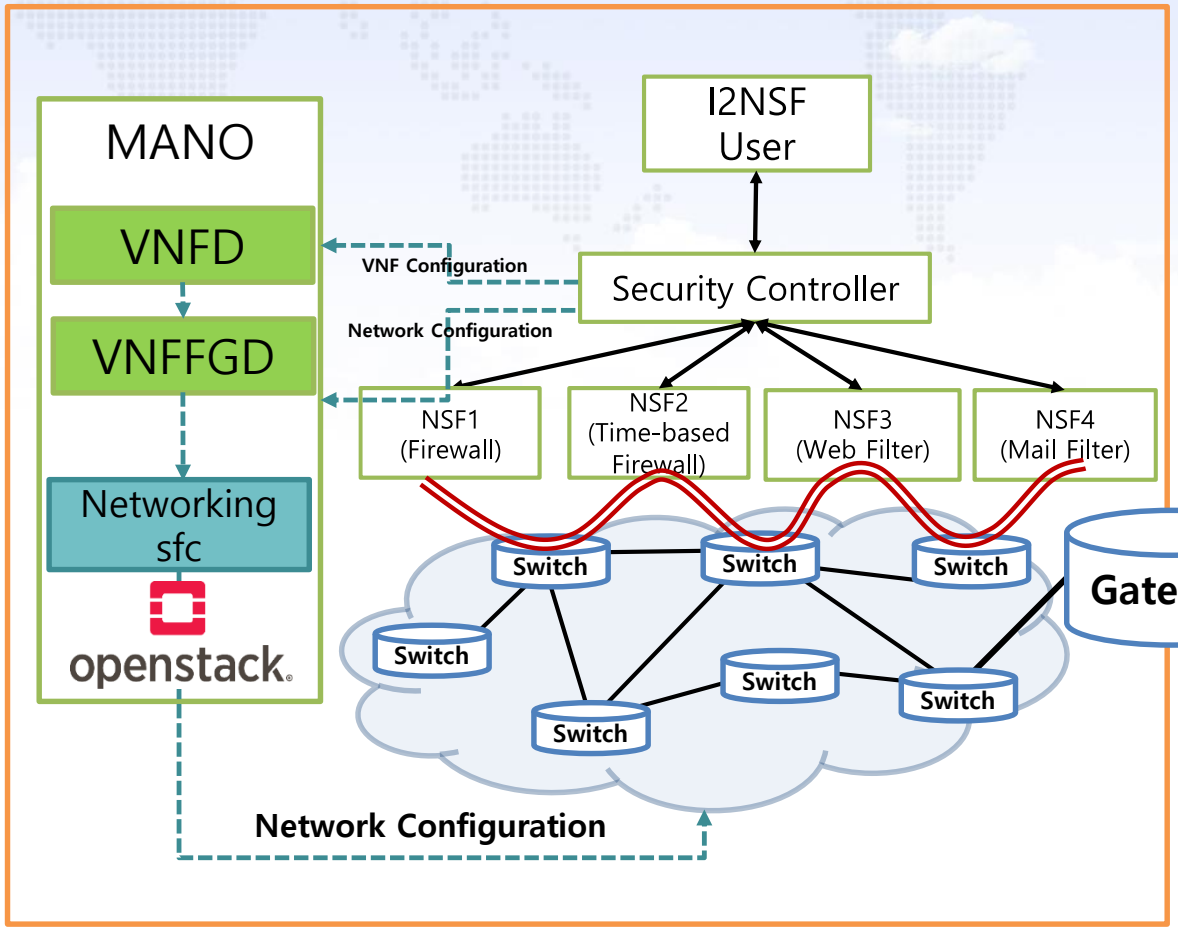
Cloud-Based Security Service System

- NFV-I2NSF Framework for Security Services (e.g., Time-Based Firewall, Web Filter, and Attack Mitigator).



Network Topology of this Hackathon Project

- NFV-I2NSF Framework for Time-Based Firewall and Web Filter works in the network below.



Web Filter

Malicious Websites

Internet

facebook

YouTube

twitter

SNS Websites

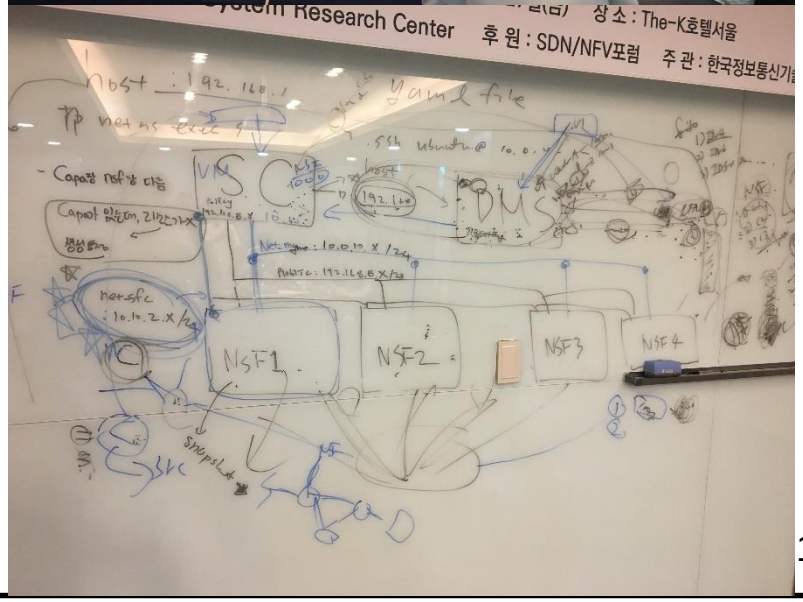
Time-based Firewall

Lessons from IETF-103 Hackathon

- **Proof of Concept (POC) of I2NSF Framework**
  - **I2NSF Interfaces (Consumer-Facing, NSF-Facing, and Registration Interface)**
  - **I2NSF Security Policy Translator**
- **Design and Implementation of I2NSF in NFV**
  - **Design of I2NSF Framework in NFV**
  - **Dynamic Creation of Virtual NSFs**
  - **Service Function Chaining for Traffic Steering**
- **Hands-on Experience of OpenStack**
  - **Dynamic NSF Creation: Tacker**
  - **SFC-based Traffic Steering: Networking-SFC**

Hackathon at IETF-Korea

➢ Preparation for IETF-103 Hackathon



Appendix

- Hackathon Development Environment
- Open-Source Depository of I2NSF Project
- Video clip of I2NSF Project

Hackathon Development Environment

Build Environment

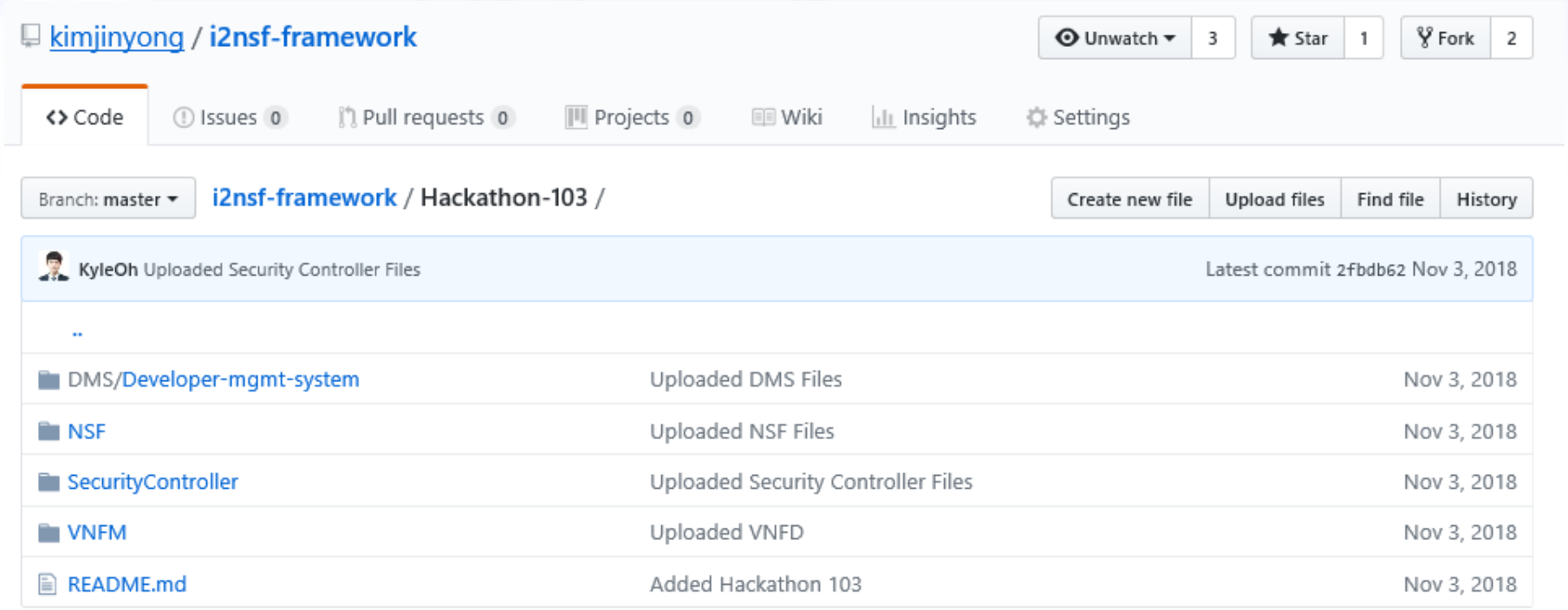
1. **OS**
   - Ubuntu 14.04TL
2. **ConfD**
   - 6.2 Version
3. **Apache2**
   - 2.4.7 Version
4. **MySQL**
   - 14.14 Version
5. **PHP**
   - 5.5.9 Version

5. **OpenStack**
   - Networking-SFC, Tacker
6. **Suricata**
   - 3.2.1 RELEASE
7. **Jetconf**
   - Python Open API for RESTCONF

Open-Source Depository of I2NSF Project

Github for I2NSF Framework Project

- **Documents and Source Code**
  https://github.com/kimjinyong/i2nsf-framework/tree/master/Hackathon-103



Demonstration Video Clip of I2NSF Project

Youtube for I2NSF Framework Project

- https://youtu.be/2fTzrDRVqsg