I2NSF Framework Project @ IETF-102 Hackathon

IETF 102, Montreal

July 15, 2018

Champion: Jaehoon Paul Jeong
Sungkyunkwan University

Why Did We Do this Project?

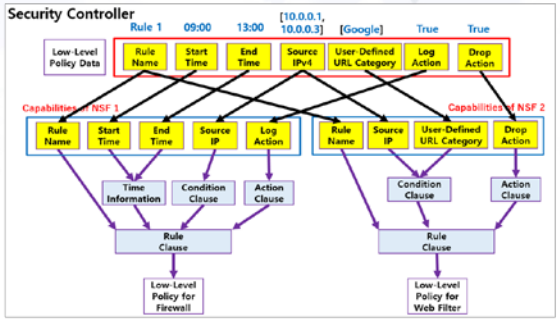
- I2NSF: Use NETCONF/RESTCONF & YANG Data Models
  - Is I2NSF good for the automatic management of network security functions (NSFs)?
  - Can we implement I2NSF using open source software?

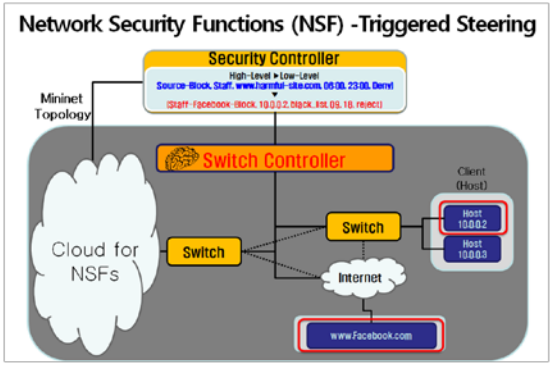
- This work is a student project!!
  - 7 graduate students (Sungkyunkwan Univ.) and 2 graduate students (Soongsil Univ.)
  - 3 professors (Sungkyunkwan Univ. and Chosun Univ.)
  - 3 researchers (ETRI, KT, and Alaxala Networks Corporation)
  - Source Code on Github
    - https://github.com/kimjinyong/i2nsf-framework/tree/master/Hackathon-102

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

Champion: Jaehoon Paul Jeong (SKKU)

I2NSF User (Web)





Professors

- Jaehoon (Paul) Jeong (SKKU)
- Hyoungshick Kim (SKKU)
- Sangwon Hyun (Chosun Univ.)

Collaborators

- Jung-Soo Park (ETRI)
- Tae-Jin Ahn (Korea Telecom)
- Toru Asahina (Alaxala Networks Corporation, Japan)

Students

- Jinyong Tim Kim (SKKU)
- Eunsoo Kim (SKKU)
- Dongjin Hong (SKKU)
- Tae-Kyun Roh (SKKU)
- Sarang Wi (SKKU)
- Seungjin Lee (SKKU)
- Jinhyuk Yang (SKKU)
- Kyoungjae Sun (SSU)
- Hyunsik Yang (SSU)

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
- Mininet: 2.2.1 Version
- OpenDaylight: Distribution-karaf-0.4.3-Beryllium-SR3
- 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 (New Feature)
- Network Security Functions
  - ✓ Firewall using SDN and Suricata
  - ✓ Web filter using Suricata
  - ✓ Mail filter using Suricata (New Feature)
- Advanced Functions
  - ✓ NSF-triggered Traffic Steering using SFC (New Feature)
  - ✓ Security Policy Translation (New Feature)
  - ✓ I2NSF Policy Provisioning (New Feature)

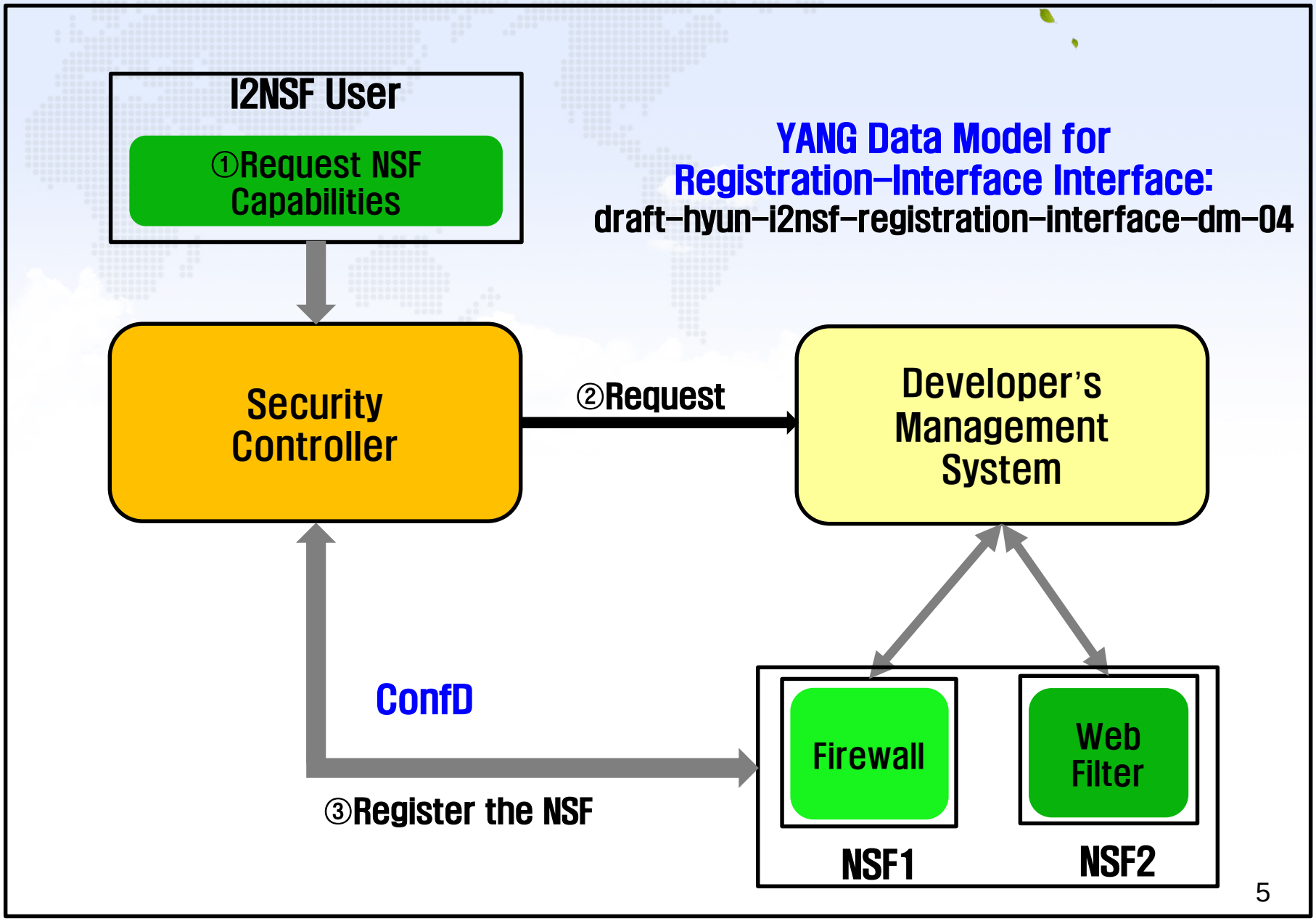
Goal of I2NSF Project

1. **NETCONF Support for Registration Interface according to the Capability YANG Data Model.**

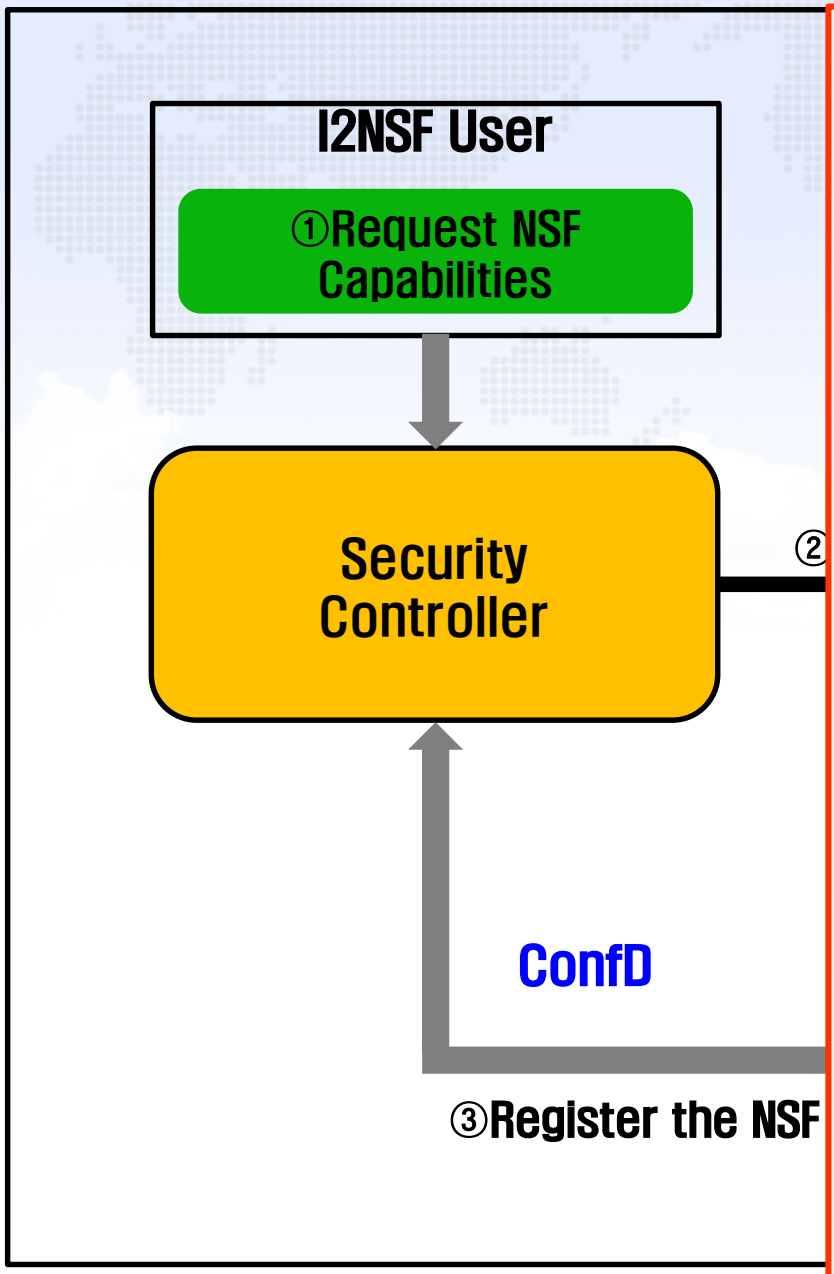
2. **I2NSF Security Policy Translation from a high-level security policy to a low-level security policy using automata.**

3. **Automatic Policy Provisioning to appropriate NSFs using the Capability of NSFs.**

NETCONF Support for Registration Interface

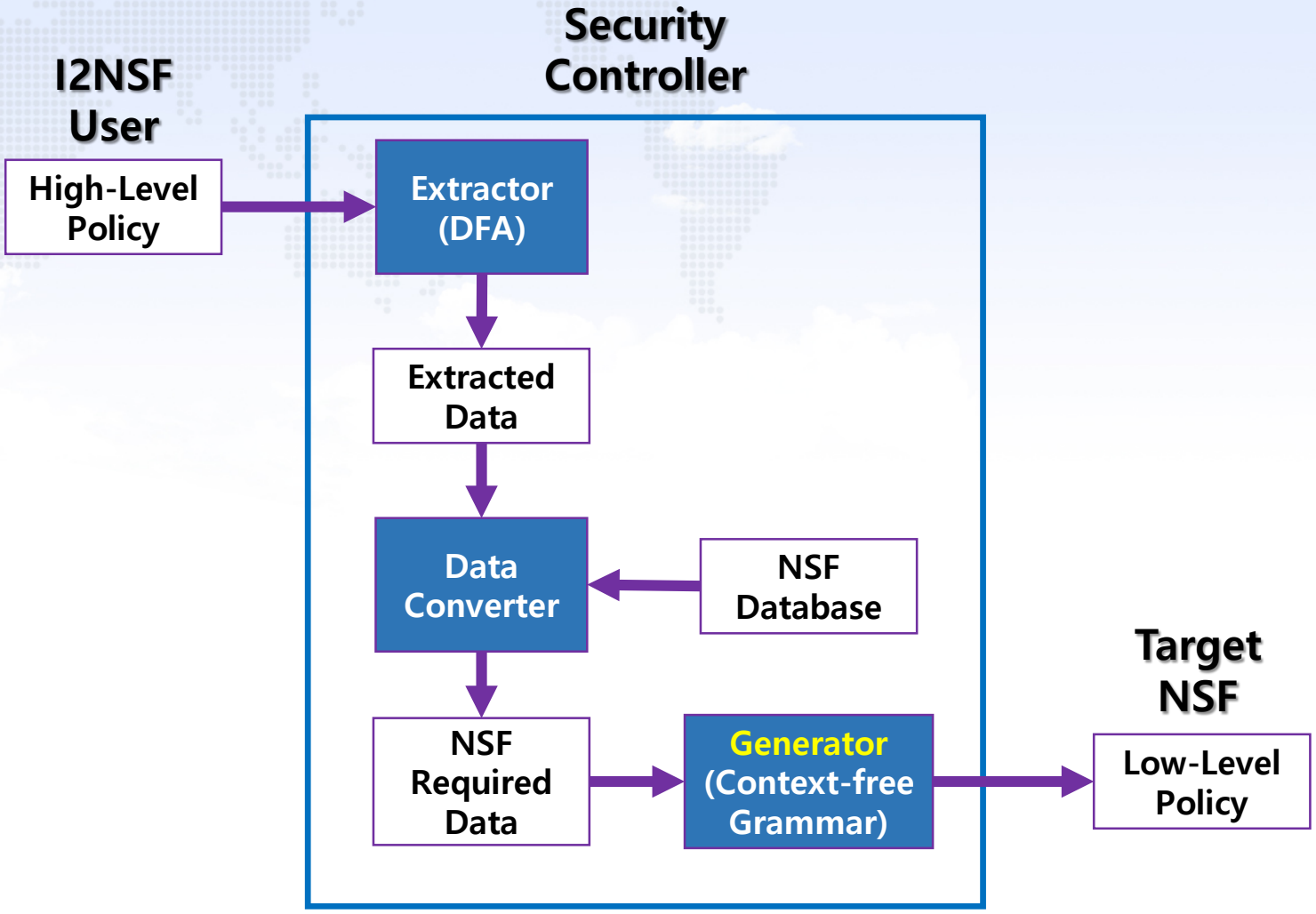


NETCONF Support for Registration Interface



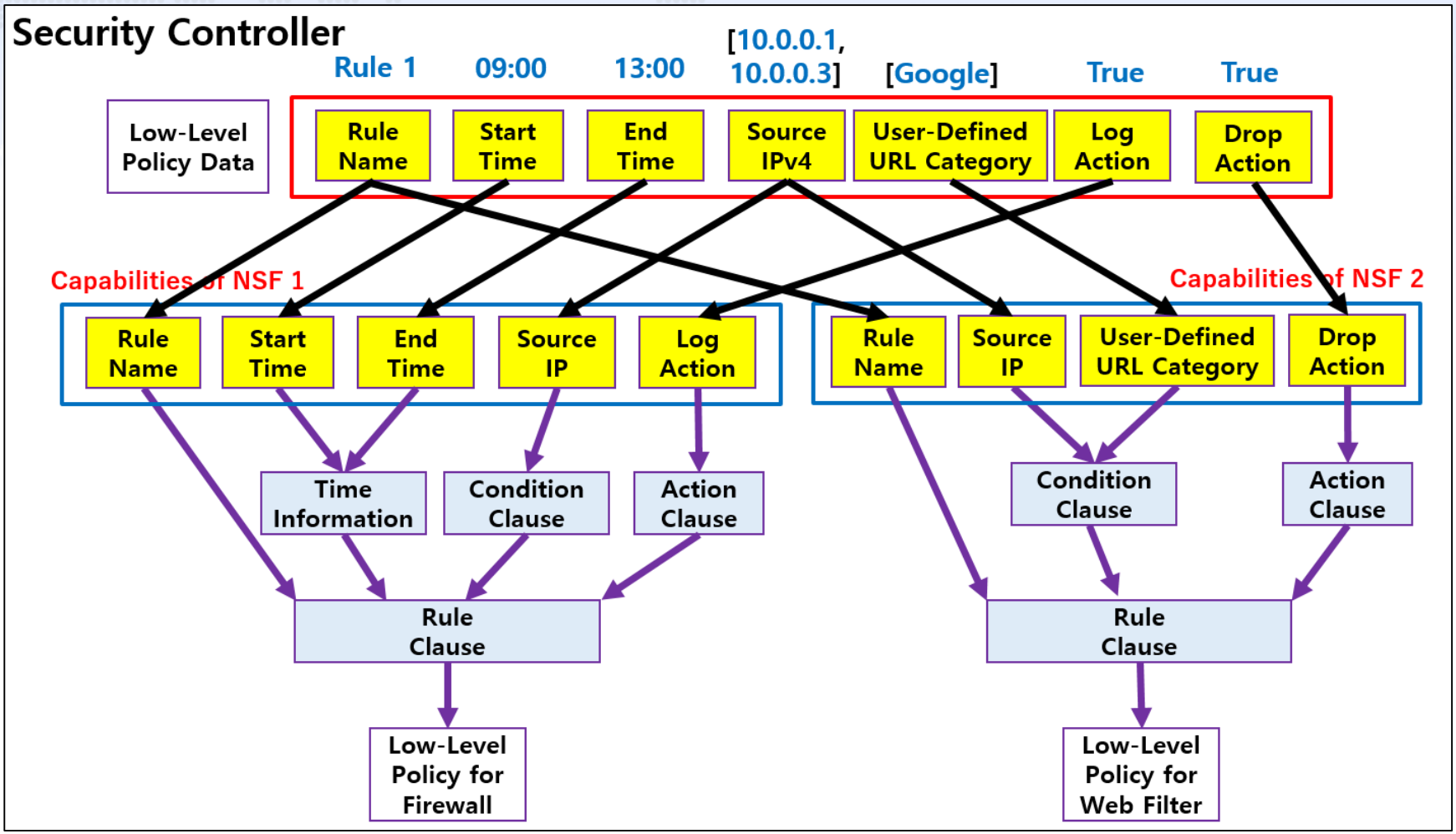

```
Instance Management Request
  +--rw i2nsf-instance-mgnt-req
     +--rw req-level uint16
     +--rw req-id uint64
     +--rw (req-type)?
        +--rw (instanciation-request)
           +--rw in-nsf-capability-information
           |  uses i2nsf-nsf-capability-information
        +--rw (deinstanciation-request)
           +--rw de-nsf-access-info
           |  uses i2nsf-nsf-access-info
        +--rw (updating-request)
           +--rw update-nsf-capability-information
           |  uses i2nsf-nsf-capability-information

NSF Capability Information
  +--rw i2nsf-nsf-capability-information
     +--rw i2nsf-capability
     |  uses ietf-i2nsf-capability
     +--rw performance-capability
     |  uses i2nsf-nsf-performance-caps

NSF Access Information
  +--rw i2nsf-nsf-access-info
     +--rw nsf-address  inet:ipv4-address
     +--rw nsf-port-address inet:port-number
NSF Performance Capability
  +--rw i2nsf-nsf-performance-caps
    +--rw processing
     |    +--rw processing-average uint16
     |    +--rw processing-peak uint16
    +--rw bandwidth
     |    +--rw outbound
     |    |  +--rw outbound-average uint16
     |    |  +--rw outbound-peak uint16
     |    +--rw inbound
     |    |  +--rw inbound-average uint16
     |    |  +--rw inbound-peak uint16
Role-Based ACL
  +--rw role-based-acl
       uses ietf-netmod-acl-model
```

I2NSF Security Policy Translation



Automatic Policy Provisioning

Generator



Network Configuration for Hackathon

VPN

VPN

Small and Medium-Sized Enterprises

Security Controller

NSF1 (Firewall)

NSF2 (Web Filter)

NSF3 (Time-based Firewall)

NSF4 (Mail Filter)

SDN Network

Switch Switch Switch Switch

Switch Switch

Gateway (NAT)

Internet

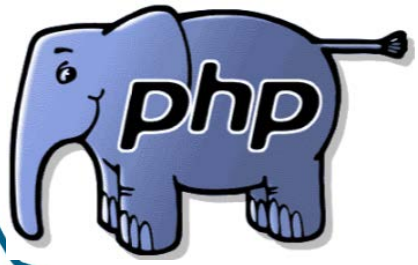
Malicious Websites

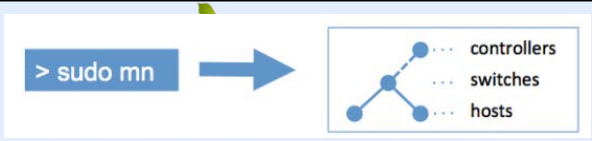
Malicious Emails

Hackathon Development

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



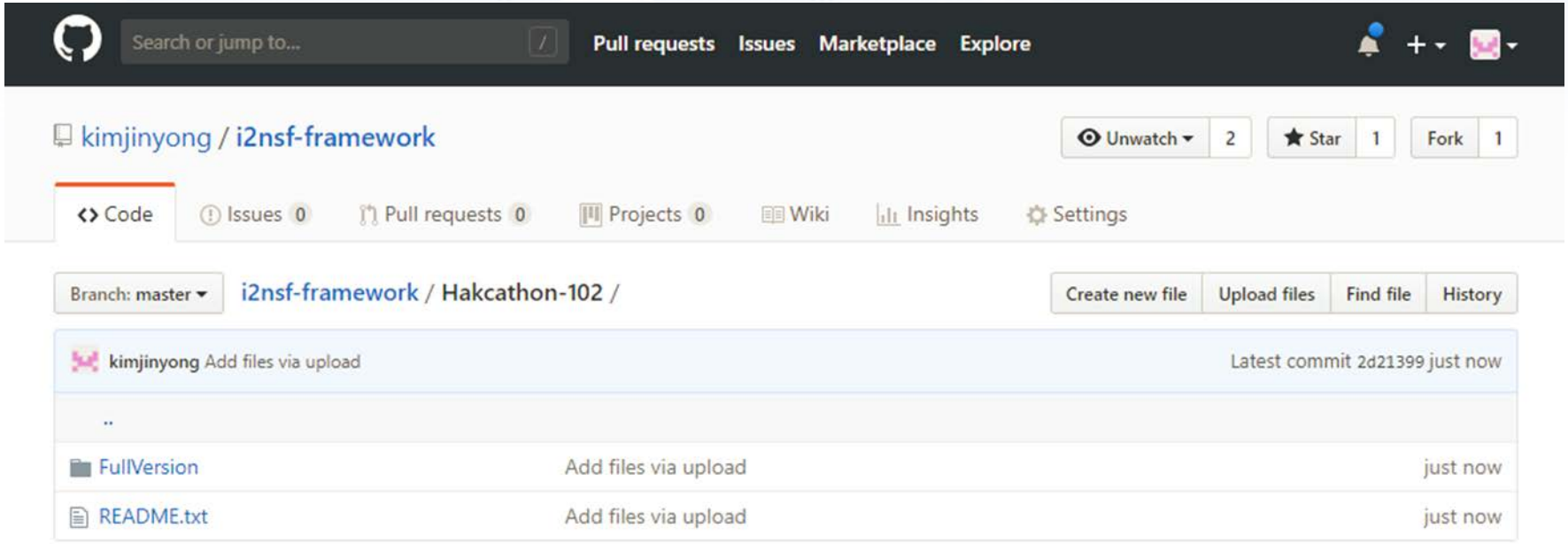


5. Mininet
   - 2.2.1 Version
6. OpenDaylight
   - Distribution-karaf-0.4.3-Beryllium-SR3
7. Suricata
   - 3.2.1 RELEASE
8. Jetconf
   - Python Open API for RESTCONF

Open-Source Project for I2NSF Framework

Github for I2NSF Framework Project

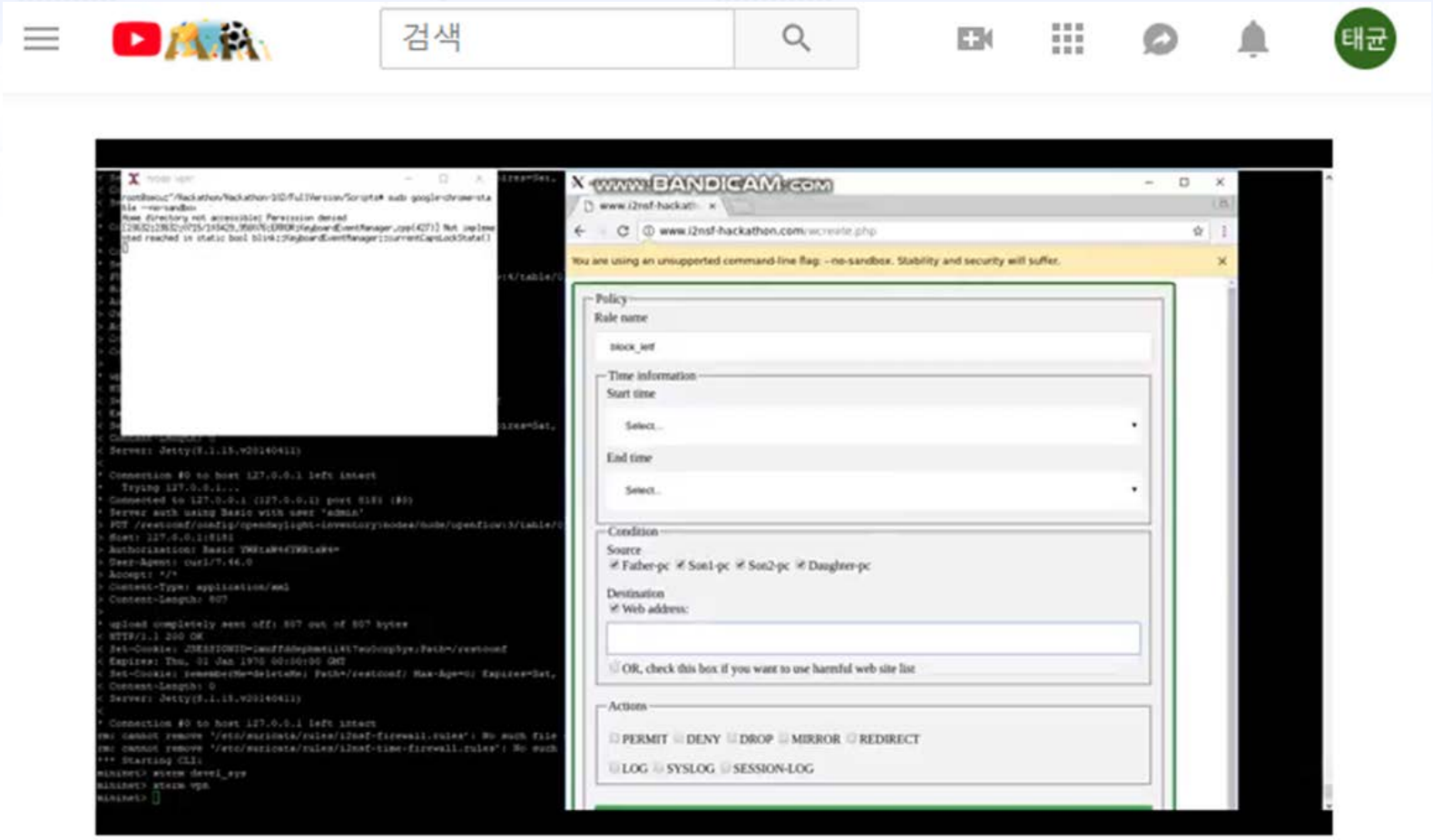
- **Documents and Source Code**
  https://github.com/kimjinyong/i2nsf-framework/tree/master/Hackathon-102



Youtube of I2NSF Hackathon Project

Youtube for I2NSF Framework Project

- https://www.youtube.com/watch?v=txkCmlgUvpc&feature=youtu.be



Lessons from the Implementation @ Hackathon

- **Proof of Concept (POC) of I2NSF Framework**
- **YANG Data Models for I2NSF Interfaces**
  - **Registration Interface**
- **I2NSF Security Policy Translation**
  - **Security Policy Translator**
  - **Automatic Policy Provisioning**
- **Open Source-Based I2NSF Framework**
  - **Confd for NSF-Facing and Registration Interfaces**
  - **JetConf for Consumer-Facing Interface**
  - **Suricata for NSFs (i.e., Firewall, Web/Mail Filters)**
  - **OpenDaylight for SDN Switch Controller**
  - **Mininet for SDN Network**