I2NSF Project @ IETF-100 Hackathon

IETF 100, Singapore
November 14, 2017

Champions: Jaehoon Paul Jeong, Sangwon Hyun (Presenter), and Jinyong Tim Kim
Sungkyunkwan University
Why Did We Do this Project?

- **I2NSF: Use NETCONF, RESTCONF, YANG Data Models**
  - Is I2NSF reasonable for management of network security functions?
  - Can we implement I2NSF using open source software?

- **This work is a student project!!**
  - 7 graduate students at Sungkyunkwan University
  - Source Code on Github
Champions: Jaehoon Paul Jeong, Sangwon Hyun, and Jinyong Tim Kim (SKKU)

Professors
- Jaehoon (Paul) Jeong (Sungkyunkwan)
- Hyoungshick Kim (Sungkyunkwan)
- Sangwon Hyun (Sungkyunkwan)

Collaborators
- Jongsu Park (ETRI)
- Tae-Jin Ahn (Korea Telecom)

Students
- Jinyong Tim Kim
- Daeyoung Hyun
- Eunsoo Kim
- Dongjin Hong
- Tae-Kyun Roh
- Sarang Wi
- Seungjin Lee

Where to get code
- Github – Source code
  - https://github.com/kimjinyong/i2nsf-framework/tree/master/Hackathon-100

What to pull down to set up an environment
- OS: Ubuntu 14.04TL
- Conf 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
- XSLT (Extensible StyleSheet Languages Transformations)
- Jetconf: Jetconf is a python Open API for RESTCONF.

Manual for Operation Process
- README.txt

Contents of Implementation
- I2NSF Framework for provisioning Network Security Functions (NSF)
  - Consumer-Facing Interface via RESTCONF/YANG
  - NSF-Facing Interface via NETCONF/YANG
  - Registration Interface via NETCONF/YANG
- Network Security Functions
  - Firewall using SDN and Suricata
  - Deep Packet Inspection (DPI) using Suricata
- Advanced Functions
  - NSF-triggered Traffic Steering using SFC
  - YANG Data Modeling for NSF Monitoring
  - Dynamic Policy Configuration
I2NSF Framework is extended with

1. **Consumer-facing interface** based on RESTCONF and YANG.

2. **Registration interface** based on NETCONF and YANG.

3. **Service Function Chaining (SFC)** based on Network Service Header (NSH) and tunneling protocol (e.g., GRE)
Build Environment

1. OS
   - Ubuntu 14.04TL

2. Netconfd
   - 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
Network Configuration for Hackathon

- **I2NSF User**
- **Security Controller**
- **NSF1 (Firewall)**
- **SFF**
- **NSF2 (Web Filter)**
- **SDN Controller**
- **Gateway (NAT)**
- **Internet**

Block the access to Facebook from 9am to 6pm

Enterprise Network

Host 10.0.0.1 (Employee)
Consumer-Facing Interface Implementation

**Consumer-Facing Interface based on RESTCONF & YANG**

I2NSF User

RESTCONF server

Security Controller

Firewall

Web filter

"Block the access of staff to Facebook from 9am to 6pm"

```xml
<i2nsf:Policy_web>
  <rule>
    <rule-id>1</rule-id>
    <rule-name>example</rule-name>
    <event>
      <time-information>
        <start-time>09:00</start-time>
        <end-time>18:00</end-time>
      </time-information>
    </event>
    <condition>
      <source>Staff</source>
      <destination>www.facebook.com</destination>
    </condition>
    <action>
      <action-name>drop</action-name>
    </action>
  </rule>
</i2nsf:Policy_web>
```

draft-jeong-i2nsf-consumer-facing-interface-dm-04

*JetConf for RESTCONF*
Registration Interface Implementation (1/2)

I2NSF User

Security Controller

Capability (e.g., IP & port inspection)
YANG data of Firewall in XML

Registration Interface
ConfD for NETCONF/YANG

Developer’s Mgmt System

Firewall

Web Filter

* Confd for NETCONF
* draft-hares-i2nsf-capability-data-model-05
Registration Interface Implementation (2/2)

I2NSF User

Security Controller

Capability (e.g., URL Inspection) YANG data of Web Filter in XML

Registration Interface Based on NETCONF/YANG

Developer’s Mgmt System

Firewall

Web Filter

* Confd for NETCONF
* draft-hares-i2nsf-capability-data-model-05
Network Security Functions (NSF) – Triggered Steering

Security Controller

High-Level ▶ Low-Level
(Staff–Facebook–Block, Staff, Facebook, 09:00, 18:00, Reject)
(Staff–Facebook–Block, 10.0.0.2, facebook, 09, 18, reject)

Service Function Forwarder (Classifier)

Switch Controller

Switch

Host packet forwarding (Firewall → Web Filter)

Switch

Internet

Server (Web Site)

NSH Proxy

Outer IP

Tunneling

NSH

Facebook

www.facebook.com

Mininet Topology

Cloud for NSFs

Steering

Mininet

VoIP/VoLTE

Web Filter

DDoS-attack mitigation

Attach

VoIP

VoLTE
Information of I2NSF Hackathon Project

Github for I2NSF Hackathon

- Documents and Source Code
  https://github.com/kimjinyong/i2nsf-framework
Proof of Concept (POC) of I2NSF Framework using Open Sources:

- **Confd** for I2NSF NSF-Facing and Registration Interface
- **JetConf** for Consumer-Facing Interface
- **Suricata** for NSFs (i.e., Firewall and Web Filter)
- **OpenDaylight** for SDN Controller
- **NSH and tunneling** for packet steering over NSFs
- **Mininet** for SDN Network
IETF 100

❖ Best Student Project Award!!