

# IETF-109 Hackathon

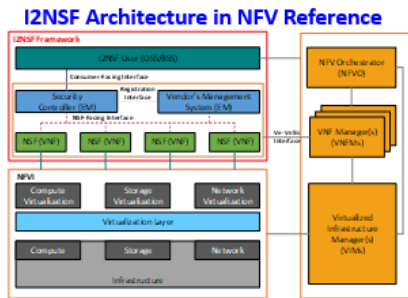
## I2NSF Framework Project

**November 9-13, 2020**  
**Online**

**Champion:** Jaehoon Paul Jeong (SKKU)  
Computer Science & Engineering  
Sungkyunkwan University  
[pauljeong@skku.edu](mailto:pauljeong@skku.edu)

# I2NSF (Interface to Network Security Functions) Framework Project

Champion: Jaehoon (Paul) Jeong (SKKU)



## Where to get Code and Demo Video Clip

- Github – Source Code
  - ✓ <https://github.com/jaehoonpaul/i2nsf-framework>
- Youtube – Demo Video Clip
  - ✓ <https://youtu.be/dAA1WTGhIXE>

## What to pull down to set up an environment

- OS: Ubuntu 16.04 LTS
- Confd for NETCONF: 6.6 Version
- Jetconf for RESTCONF
- OpenStack: Queens version
- NSF: Suricata

## Professors:

- Jaehoon (Paul) Jeong (SKKU)
- Younghan Kim (SSU)

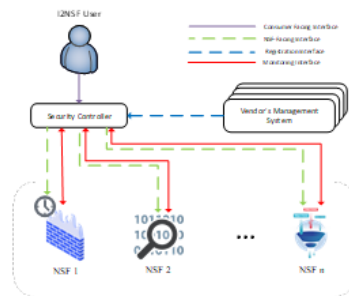
## Researchers:

- Jung-Soo Park (ETRI)
- Yunchul Choi (ETRI)

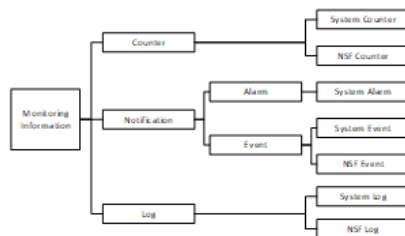
## Students:

- Patrick Lingga (SKKU)
- Yoseop Ahn (SKKU)
- Mose Gu (LU)
- Hyunsik Yang (SSU)
- Kyungsik Kim (KNU)

## I2NSF Framework



## Monitoring Data Model Information



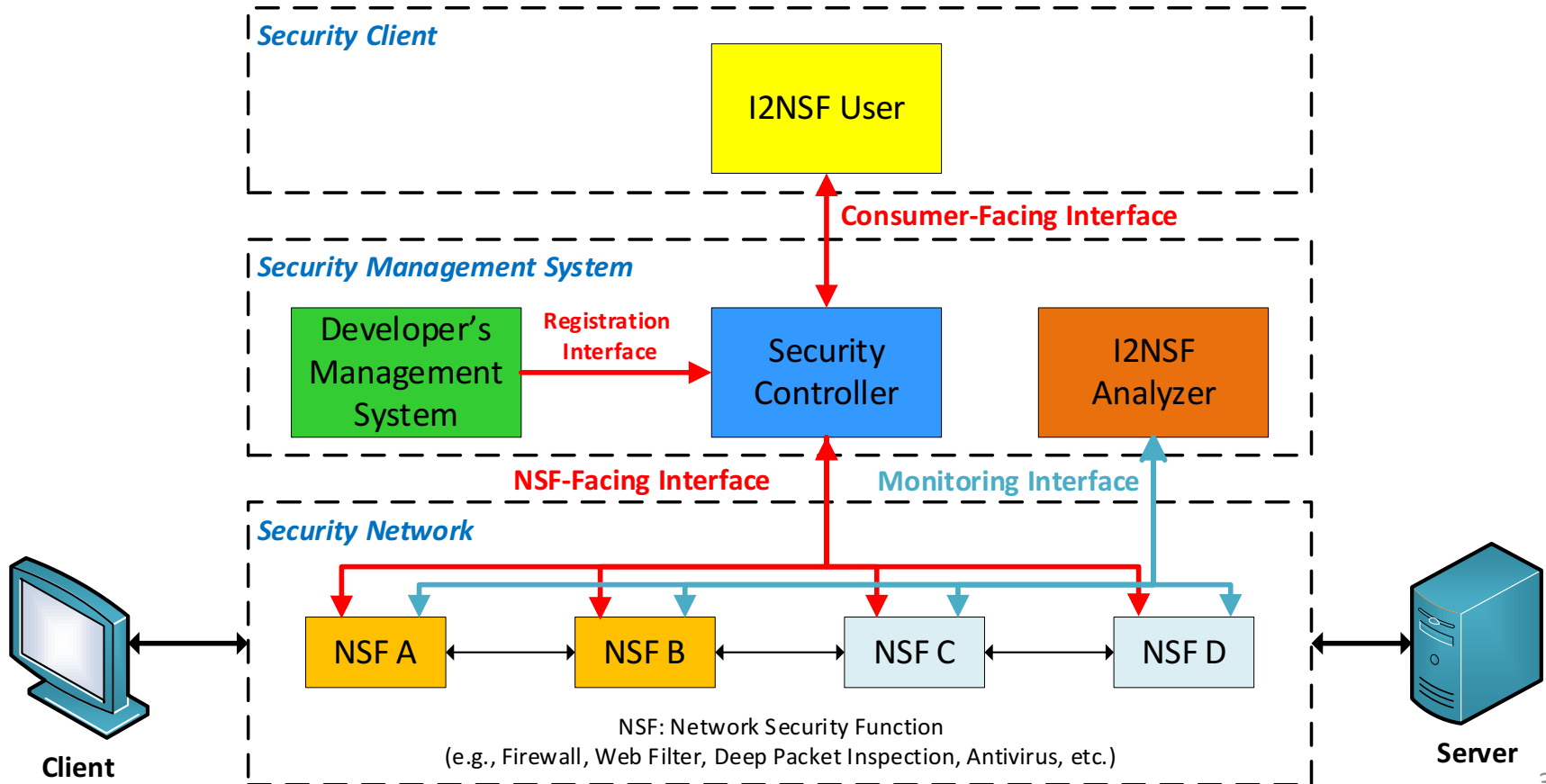
## Manual for Operation Process

- Manual.txt contain detailed description about operation process. (It can be found in Open Source Project folder.)

## Contents of Implementation

- Cloud-based Security Service System using I2NSF Framework
  - ✓ Web-based I2NSF User
  - ✓ Console-based Security Controller
  - ✓ Console-based Developer's Management System
  - ✓ I2NSF Framework in OpenStack NFV Environment
  - ✓ I2NSF Capability YANG Data Model
  - ✓ Registration Interface via NETCONF/YANG
  - ✓ Consumer-Facing Interface via RESTCONF/YANG
  - ✓ NSF-Facing Interface via NETCONF/YANG
  - ✓ Monitoring Interface via NETCONF/YANG
- Network Security Functions
  - ✓ Firewall and Web-filter using Suricata
- Advanced Function
  - ✓ Security Policy Translation
  - ✓ Security Policy Provisioning

# I2NSF for Security Management Automation



# I2NSF Framework: Interfaces

- Registration Interface

- Developer's Management System (DMS) registers an NSF with Security Controller.

- Consumer-Facing Interface

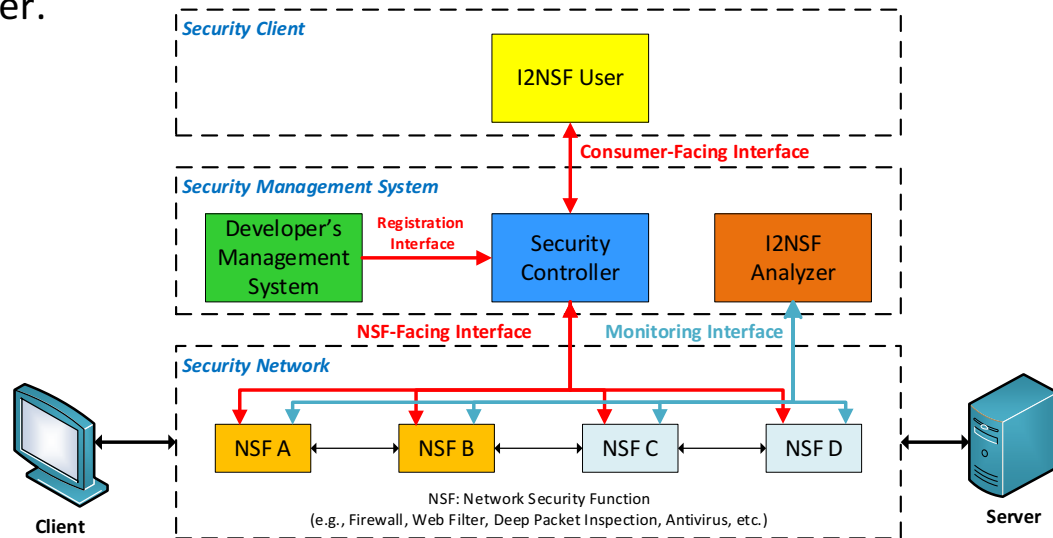
- I2NSF User delivers a high-level security policy to Security Controller.

- NSF-Facing Interface

- Security Controller delivers a low-level security policy to an NSF.

- Monitoring Interface

- An NSF delivers its monitoring data to I2NSF Analyzer.

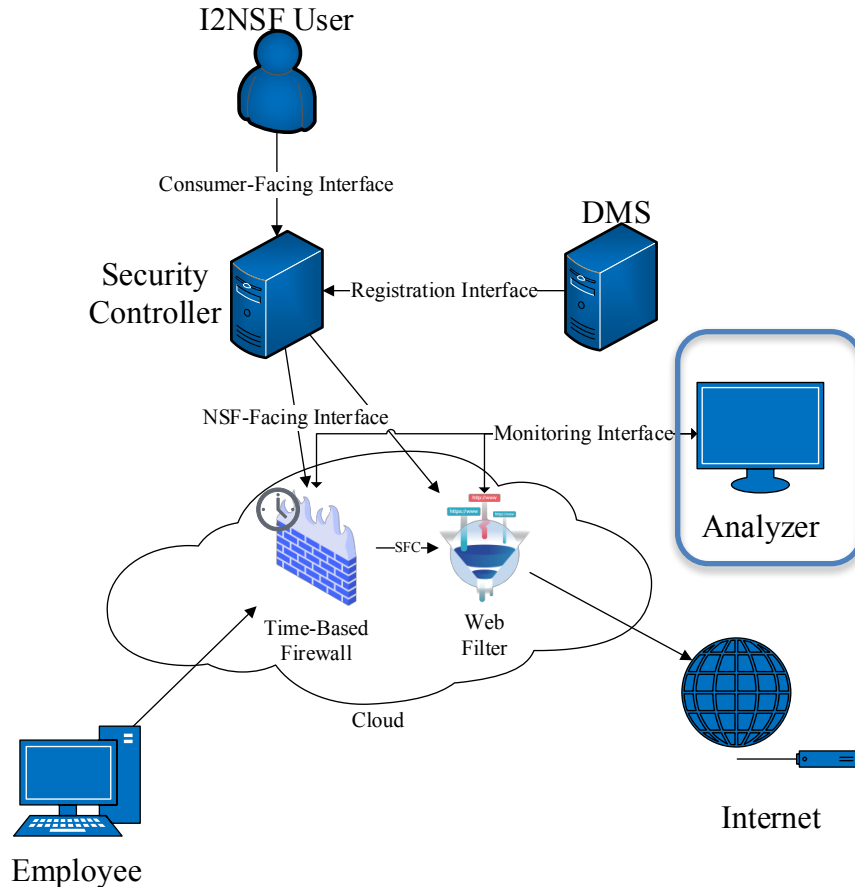


# Hackathon Plan

## ❖ The Implementation of the Internet Drafts for the I2NSF System for Cloud-based Security Services:

- draft-ietf-i2nsf-capability-data-model-13
- draft-ietf-i2nsf-consumer-facing-interface-dm-12
- draft-ietf-i2nsf-nsf-facing-interface-dm-10
- draft-ietf-i2nsf-registration-interface-dm-09
- draft-ietf-i2nsf-nsf-monitoring-data-model-04
- draft-yang-i2nsf-security-policy-translation-06
- draft-jeong-i2nsf-security-management-automation-00

# Network Topology for Hackathon Project



- **Open Source:**

- ✓ **OS:** Ubuntu 16.04 LTS
- ✓ **ConfD:** 6.6 Version
- ✓ **MySQL:** 14.14 Version
- ✓ **OpenStack:** Queens
- ✓ **Suricata:** 3.2.1 RELEASE
- ✓ **RestConf:** JETCONF Server

- **Minimum Specification for OpenStack:**

- ✓ **RAM:** 4 ~ 8 GB
- ✓ **Storage:** 10 GB
- ✓ **CPU:** 2 ~ 4 cores @ 2.4 GHz

# What got done (1/3)

- I2NSF Framework on top of OpenStack.
  - Web-based I2NSF User.
  - Console-based Security Controller and DMS.
  - Security Policy Translator in Security Controller.
  - Security Policy Provisioning
- NSFs Monitoring using I2NSF Monitoring Interface.
  - Pulling of NSF's Resources Data (CPU, Memory, Disk, Interface)

# What got done (2/3)

The screenshot shows a web browser window with the URL `172.244.12/rule_create.php`. The page has a green border and a red asterisk indicating a required field. The form is titled "Policy" and contains the following fields:

- Policy name:** `security_policy_for_blocking_sns`
- Rule name:** `block_access_to_sns_during_office_hours`
- Condition:**
  - Source Target:** `employees`
  - Destination Target:** `sns-websites`
  - Start time (YYYY-MM-DDThh:mm:ssZ, ex: 2020-07-20T09:00:00Z):** `2020-07-20T09:00:00Z`
  - End time (YYYY-MM-DDThh:mm:ssZ, ex: 2020-07-20T18:00:00Z):** `2020-07-20T18:00:00Z`
  - Frequency:** `weekly`
  - Day:** ☒ Monday ☒ Tuesday ☒ Wednesday ☒ Thursday ☒ Friday ☐ Saturday ☐ Sunday
- Actions:** `Drop`

A green "Submit" button is at the bottom of the form.

Web-based I2NSF User

Security Controller

Developer's  
Management  
System (DMS)

The screenshot shows a terminal window with the following output:

```
Test: url_filtering
Test: voip_volte_filter
Test: http_and_https_flood_mitigation
Current Num: 4
TRACE CDB_GET i2nsf-nsf-capability-registration[4]/nsf-name --> CONFID_OK
nsf-name: voip_volte_filter
Test: general_firewall
Test: time_based_firewall
Test: url_filtering
Test: voip_volte_filter
Test: http_and_https_flood_mitigation
Success
Read new config, updating dhcpd config
TRACE CDB_SYNC_SUB CDB_DONE_PRIORITY --> CONFID_OK

ubuntu@dms: ~
Created NSF Information
NSF Name: time_based_firewall
NSF IP: 10.0.0.25

[35] Creating NSF...
[36] Creating NSF...
[37] Creating NSF...
[38] Creating NSF...
[39] Creating NSF...
[40] Creating NSF...
[41] Creating NSF...
[42] Creating NSF...
[43] Creating NSF...
[44] Creating NSF...

Create NSF!!!
Created NSF Num: 2
Created NSF Information
NSF Name: url_filtering
NSF IP: 10.0.0.12

##### Complete creation of all NSFs #####
```

Console-based Security  
Controller and DMS



# What got done (3/3)

## Monitoring of NSFs

NSF (Web Filter)



**Monitoring  
Interface  
(JSON/RESTCONF)**

I2NSF Analyzer

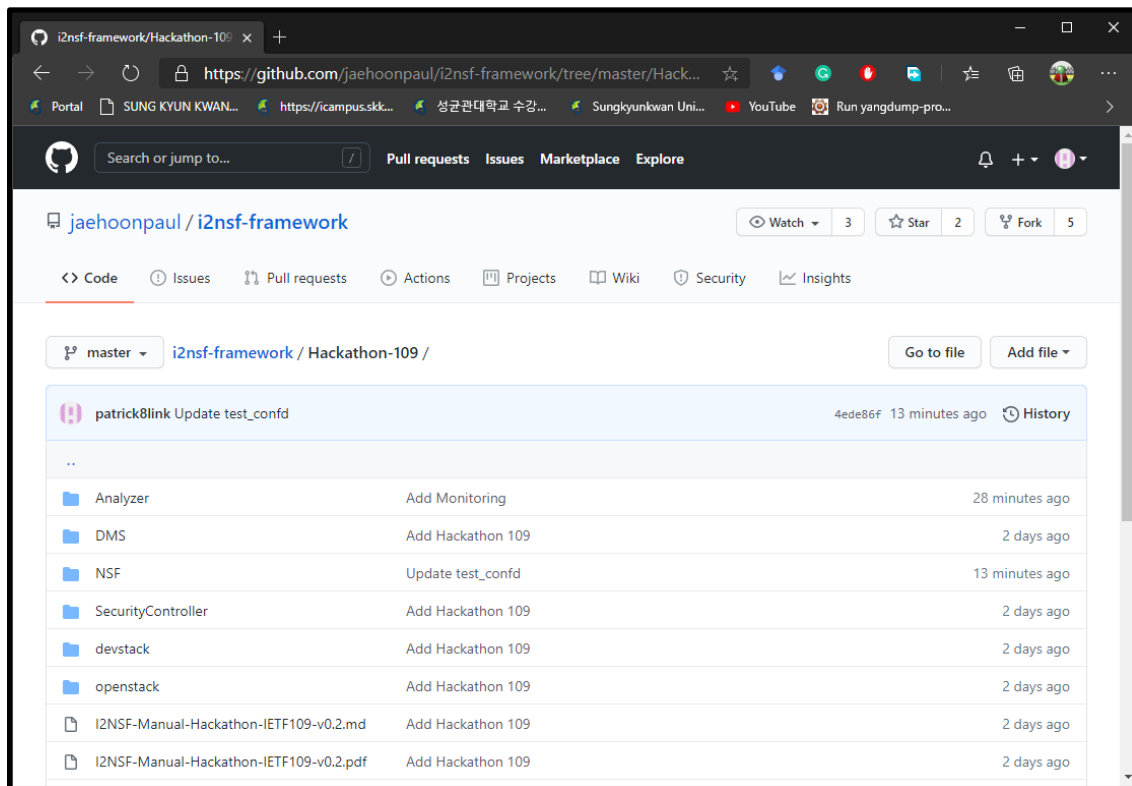
```
{
  "counters": {
    "system-interface": [
      {
        "in_total_traffic_bytes": 3082,
        "interface-name": "ens3",
        "nsf-name": "url_filtering",
        "out_total_traffic_bytes": 2185
      }
    ]
  },
  "system-res-util-log": {
    "cpu_usage": 1.9,
    "disk-left": 17384,
    "disk-usage": 2388,
    "memory-usage": 22.6
  }
}
```

# What we learned

- We realized the feasibility of NSF monitoring via RESTCONF or NETCONF for Security Management.
- Next Steps:
  - Support of NETCONF for NSF Monitoring Interface.
  - Enhancement of Security Policy Translator for Automatic Setup.
  - Implementation of Application Interface for Security Management Automation.

# Open Source Project at Github

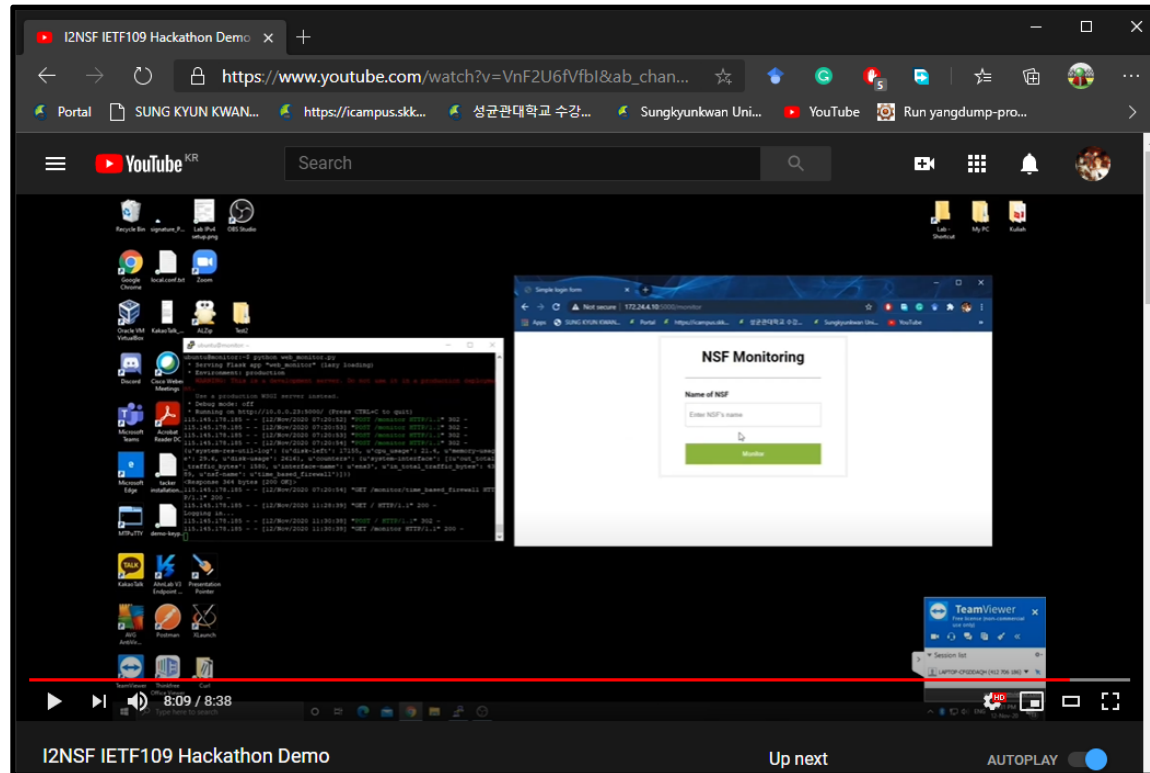
- URL: <https://github.com/jaehoonpaul/i2nsf-framework/tree/master/Hackathon-109>



I2NSF Hackathon Project

# Demo Video Clip at YouTube

- URL: <https://youtu.be/VnF2U6fVfbl>



# Wrap Up

## Hackathon Team

### Champion:

- Jaehoon Paul Jeong (SKKU)

### Professor:

- Younghan Kim (SSU)

### Researchers:

- Jung-Soo Park (ETRI)
- Yunchul Choi (ETRI)

### Students:

- Patrick Lingga (SKKU)
- Yoseop Ahn (SKKU)
- Mose Gu (Liberty University in US)
- Hyunsik Yang (SSU)
- Kyungsik Kim (KNU)



I2NSF hackathon team worked in collaboration with IPWAVE and BMWG teams.