IETF-115

I2NSF Hackathon Project

November 5-6, 2022

Champions: Jaehoon (Paul) Jeong and Patrick Lingga

Members: Jiyong Uhm, Jeonghyeon Kim, and Mose Gu

Sungkyunkwan University

I2NSF (Interface to Network Security Functions) Framework Project

Champion: Jaehoon (Paul) Jeong

I2NSF Hackathon Project

Professors:

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

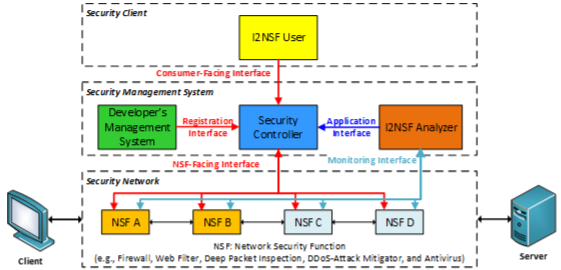
Researchers:

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

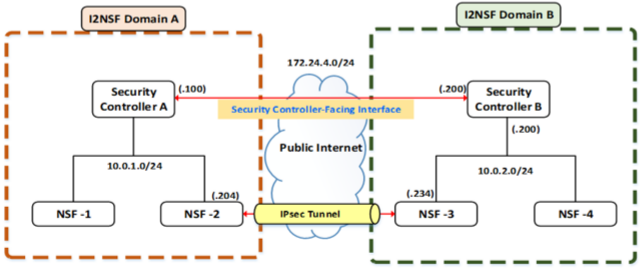
Students:

- Patrick Lingga (SKKU)
- Jeonghyeon Kim (SKKU)
- Jiyong Uhm (SKKU)
- Mose Gu (SKKU)

I2NSF Framework



Multiple I2NSF Domains



Where to get Code and Demo Video Clip

- Github – Source Code
  - ✓ https://github.com/jaehoonpaul/i2nsf-framework
  - ✓ https://github.com/patrick8link/i2nsf-ipsec
  - ✓ https://www.youtube.com/watch?v=l-bSMxOs7zw

What to pull down to set up an environment

https://github.com/patrick8link/docker-i2nsf-ipsec

- OS: Ubuntu 14.04
- DockerHub: sysrepo/sysrepo-netopeer2:legacy
- Libyang v1.0.184
- Strongswan v5.5.0

Manual for Operation Process

- README.md contains detailed description about operation process. It can be found in the GitHub.

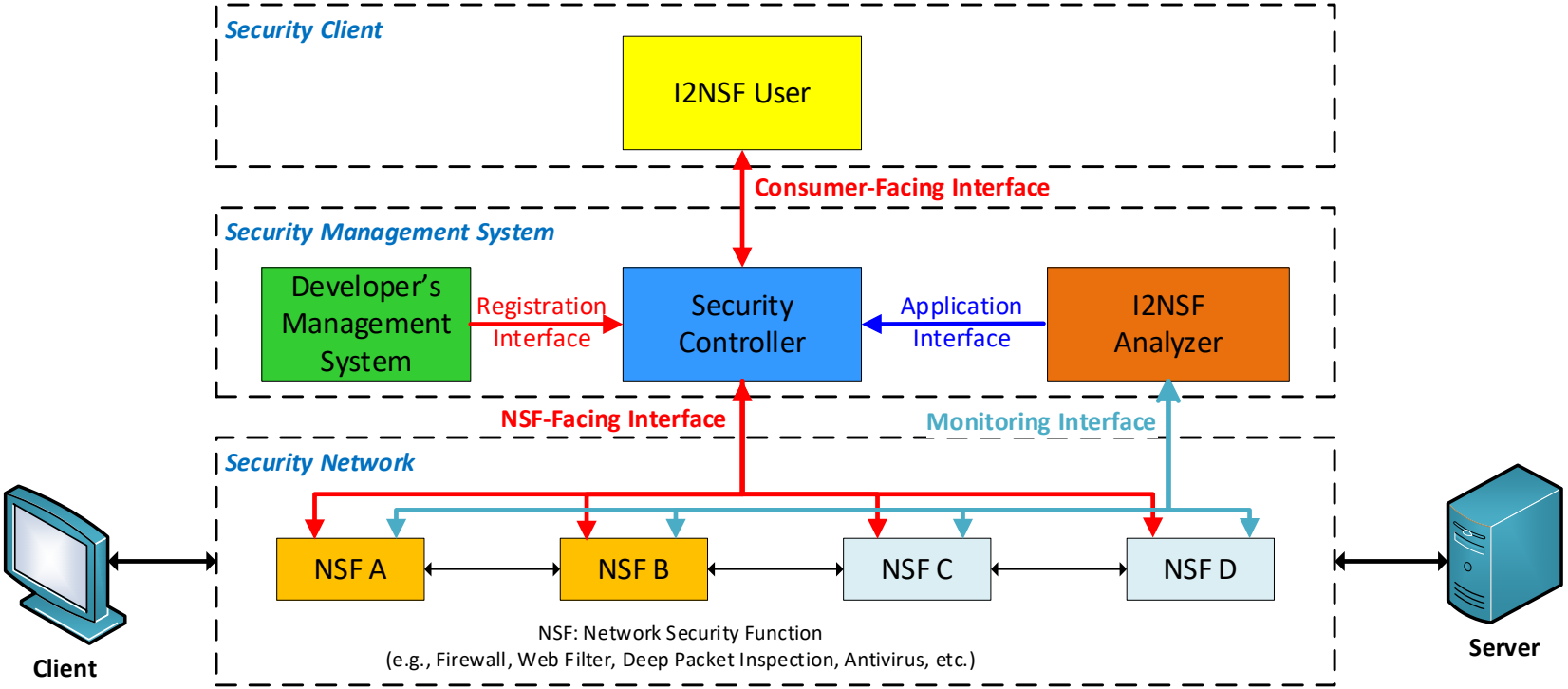
Contents of Implementation

- IPsec Flow Protection based on SDN for I2NSF Framework
  - ✓ SPD, PAD, IKE parameters for IPsec Configuration according to RFC 9061
  - ✓ Interactive client for Security Controller
  - ✓ IPsec tunnel configuration using IKEv2 protocol
  - ✓ Console-based Developer's Management System
  - ✓ I2NSF Framework in Docker Container
  - ✓ I2NSF Capability YANG Data Model
  - ✓ IPsec SA establishment through Security Controller via NETCONF/YANG
  - ✓ Registration Interface via NETCONF/YANG
  - ✓ NSF-Facing Interface via NETCONF/YANG
- West/Eastbound Interface (Security Controller-Facing Interface)
  - ✓ IPsec SA establishment across different Domains
  - ✓ IPsec tunnel configuration between two Security Controllers via NETCONF/YANG

docker Sysrepo cesnet

성균관대학교 SUNG KYUN KWAN UNIVERSITY ETRI openstack. HYPERLEDGER FABRIC ConfD tail-f a Cisco company SURICATA

Hackathon Plan (1/2)

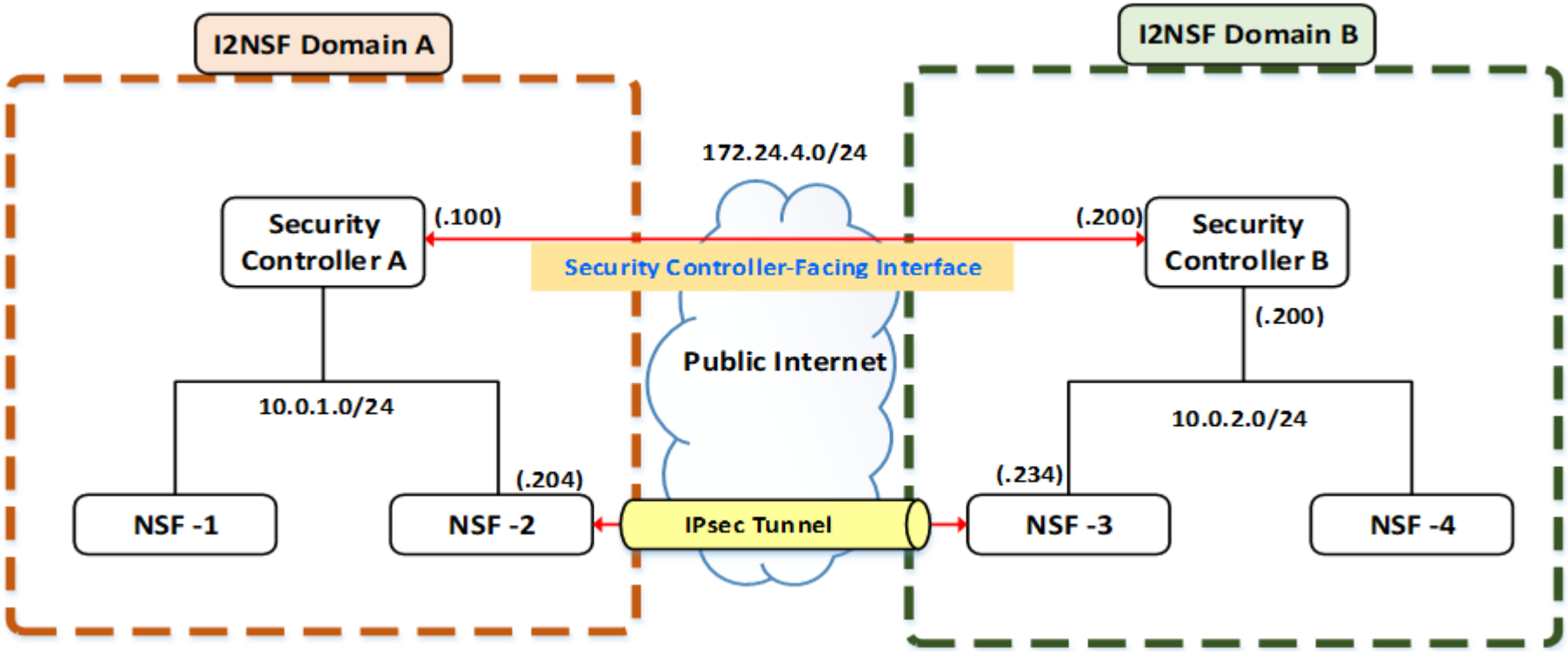


Hackathon Plan (2/2)

- **Implementation of IPsec Flow Protection based on SDN for I2NSF Framework:**
  - RFC 9061: A YANG Data Model for IPsec Flow Protection Based on Software-Defined Networking (SDN)
    - https://datatracker.ietf.org/doc/rfc9061/

- **Implementation of West/Eastbound Interface (Security Controller-Facing Interface) for I2NSF Framework:**
  - draft-kim-i2nsf-security-controller-interface-dm-00

What got done (1/4)



The Security Controller is in charge of provisioning the NSF with the required information in the SPD and PAD (e.g., IKE credentials) and the IKE protocol itself (e.g., parameters for the IKE_SA_INIT negotiation).

What got done (2/4)

```
<pad>
  <pad-entry>
    <name>Host1</name>
    <ipv4-address>192.168.123.100</ipv4-address>
    <auth-protocol>ikev2</auth-protocol>
    <peer-authentication>
      <auth-method>pre-shared</auth-method>
      <pre-shared>
        <secret>73:65:63:72:65:74:6F:5F:63:6F:6D:70:61:72:74:69:64:6F</secret>
      </pre-shared>
    </peer-authentication>
  </pad-entry>
  <pad-entry>
    <name>Host2</name>
    <ipv4-address>192.168.123.200</ipv4-address>
    <auth-protocol>ikev2</auth-protocol>
    <peer-authentication>
      <auth-method>pre-shared</auth-method>
      <pre-shared>
        <secret>73:65:63:72:65:74:6F:5F:63:6F:6D:70:61:72:74:69:64:6F</secret>
      </pre-shared>
    </peer-authentication>
  </pad-entry>
</pad>
```

```
<name>gateway1</name>
<autostartup>start</autostartup>
<version>ikev2</version>
<initial-contact>false</initial-contact>
<fragmentation><enabled>false</enabled></fragmentation>
<ike-sa-lifetime-soft>
  <rekey-time>30</rekey-time>
  <reauth-time>60</reauth-time>
</ike-sa-lifetime-soft>
<ike-sa-lifetime-hard>
  <over-time>10</over-time>
</ike-sa-lifetime-hard>
<!--AUTH_HMAC_SHA2_512_256-->
<ike-sa-intr-alg>14</ike-sa-intr-alg>
<!--ENCR_AES_CBC - 128 bits-->
<ike-sa-encr-alg>
  <id>1</id>
</ike-sa-encr-alg>
<!--8192-bit MODP Group-->
<dh-group>18</dh-group>
<half-open-ike-sa-timer>30</half-open-ike-sa-timer>
<half-open-ike-sa-cookie-threshold>
  15
</half-open-ike-sa-cookie-threshold>
<local>
  <local-pad-entry-name>Host1</local-pad-entry-name>
</local>
<remote>
  <remote-pad-entry-name>Host2</remote-pad-entry-name>
</remote>
```

```
<spd-entry>
  <name>gateway1</name>
  <ipsec-policy-config>
    <anti-replay-window-size>64</anti-replay-window-size>
    <traffic-selector>
      <local-prefix>192.168.201.0/24</local-prefix>
      <remote-prefix>192.168.202.0/24</remote-prefix>
      <inner-protocol>6</inner-protocol>
    </traffic-selector>
    <processing-info>
      <action>protect</action>
      <ipsec-sa-cfg>
        <pfp-flag>false</pfp-flag>
        <ext-seq-num>true</ext-seq-num>
        <seq-overflow>false</seq-overflow>
        <stateful-frag-check>false</stateful-frag-check>
        <mode>tunnel</mode>
        <protocol-parameters>esp</protocol-parameters>
        <esp-algorithms>
          <!-- AUTH_HMAC_SHA1_96 -->
          <integrity>2</integrity>
          <encryption>
            <!-- ENCR_AES_CBC -->
            <id>1</id>
            <algorithm-type>12</algorithm-type>
            <key-length>128</key-length>
          </encryption>
          <encryption>
            <!-- ENCR_3DES-->
            <id>2</id>
            <algorithm-type>3</algorithm-type>
          </encryption>
          <tfc-pad>false</tfc-pad>
        </esp-algorithms>
        <tunnel>
          <local>192.168.123.100</local>
          <remote>192.168.123.200</remote>
          <df-bit>clear</df-bit>
          <bypass-dscp>true</bypass-dscp>
        </tunnel>
      </ipsec-sa-cfg>
    </processing-info>
  </ipsec-policy-config>
</spd-entry>
```

PAD, IKE, and SPD parameters according to RFC 9061 .

What got done (3/4)

```
DEBUG: [PAD][IMPORTANT] CURRENT PAD NAME:
DEBUG: [SPD][TRAFFIC-SELECTOR] local-prefix: 192.168.201.0/24
DEBUG: [SPD][TRAFFIC-SELECTOR] remote-prefix: 192.168.202.0/24
DEBUG: [SPD][TRAFFIC-SELECTOR] inner-protocol: 6
DEBUG: [SPD][PROCESSING-INFO] action: protect
DEBUG: [SPD][PROCESSING-INFO] mode: tunnel
DEBUG: [SPD][PROCESSING-INFO] satype: esp
DEBUG: [SPD][PROCESSING-INFO] mode tunnel src_tunnel: 192.168.123.100
DEBUG: [SPD][PROCESSING-INFO] mode tunnel dst_tunnel: 192.168.123.200
DEBUG: [PAD][IMPORTANT] CURRENT PAD NAME: Host1
DEBUG: [PAD] ipv4-address: 192.168.123.100
DEBUG: [PAD] auth_protocol: ikev2
DEBUG: [PAD] Auth Method: pre-shared
DEBUG: [PAD] ssecret: 73:65:63:72:65:74:6f:5f:63:6f:6d:70:61:72:74:69:64:6f
DEBUG: [PAD][IMPORTANT] CURRENT PAD NAME: Host2
DEBUG: [PAD2] ipv4-address: 192.168.123.200
DEBUG: [PAD2] auth_protocol: ikev2
DEBUG: [PAD2] Auth Method: pre-shared
DEBUG: [PAD2] ssecret: 73:65:63:72:65:74:6f:5f:63:6f:6d:70:61:72:74:69:64:6f
DEBUG: Exiting addIPSEC_conn_entry
INFO: ipsec-conn-entry added
DEBUG:  ========== END OF CHANGES ======================================
######################## SENDING RPC TO 10.0.1.204:830#########################
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:f
ee7bd5c-d671-422e-8558-b8c61c8e12e4"><nc:ok/></nc:rpc-reply>
```

Security Controller A receiving IPsec configuration for NSF-2 from Security Controller B.

RPC reply from Gateway 1 to Controller 1

What got done (4/4)

```
root@gw2:/home/netconf/i2nsf-ipsec# tcpdump -i eth0 esp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
04:10:13.123432 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc28048be,seq=0x1), length 136
04:10:13.123712 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc5aca945,seq=0x1), length 136
04:10:14.147374 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc28048be,seq=0x2), length 136
04:10:14.147405 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc5aca945,seq=0x2), length 136
04:10:15.171403 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc28048be,seq=0x3), length 136
04:10:15.171438 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc5aca945,seq=0x3), length 136
04:10:16.195378 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc28048be,seq=0x4), length 136
04:10:16.195411 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc5aca945,seq=0x4), length 136
04:10:17.219383 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc28048be,seq=0x5), length 136
04:10:17.219415 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc5aca945,seq=0x5), length 136
04:10:18.243378 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc28048be,seq=0x6), length 136
04:10:18.243410 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc5aca945,seq=0x6), length 136
04:10:19.267519 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc28048be,seq=0x7), length 136
04:10:19.267569 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc5aca945,seq=0x7), length 136
04:10:20.291375 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc6c70587,seq=0x1), length 136
04:10:20.291408 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc6b8b84e,seq=0x1), length 136
04:10:21.315642 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc6c70587,seq=0x2), length 136
04:10:21.315703 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc6b8b84e,seq=0x2), length 136
04:10:22.339776 IP c2c_gw1_1.c2c_gw1_gw2_data > gw2: ESP(spi=0xc6c70587,seq=0x3), length 136
04:10:22.339810 IP gw2 > c2c_gw1_1.c2c_gw1_gw2_data: ESP(spi=0xc6b8b84e,seq=0x3), length 136
```

TCP dump of ESP packets from IPSEC configuration between NSF-2 and NSF-3.

What we learn

- IPsec SA establishment between NSFs is possible through the Security Controller. Establishing IPsec tunnel is possible with minimal intervention from the network administrator.

- In a case of multiple domains, it is possible to create IPsec tunnel by exchanging the SPD and PAD parameters between the Security Controllers.

Next Step

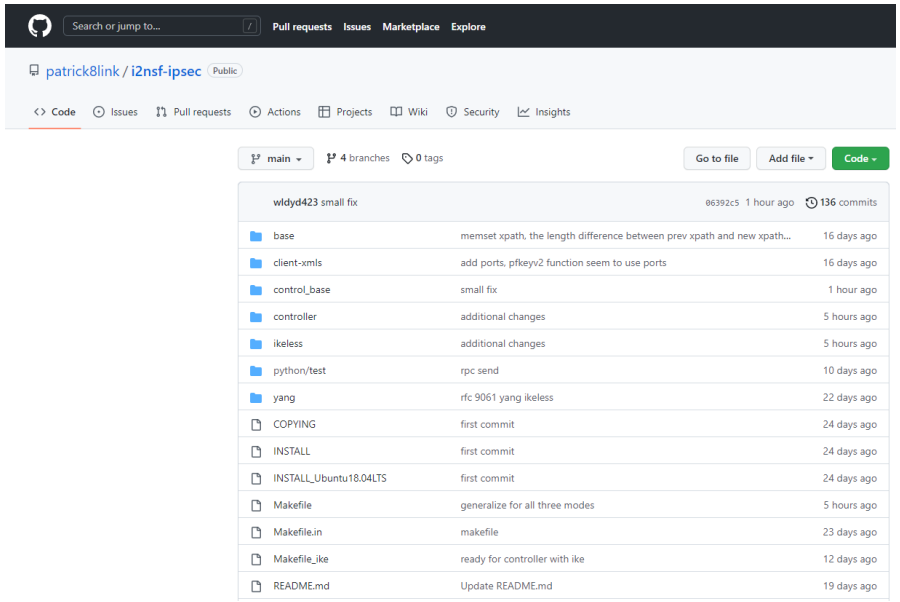
- For NSFs where IKEv2 is not available, IKE-less case is possible. As discussed in RFC 9061, this moves the task of managing SAD from IKEv2 to Security Controllers.

- Implementation of IKE-less case for the West/Eastbound Interface (Security Controller Facing Interface) will be done.

Open-Source Project at GitHub

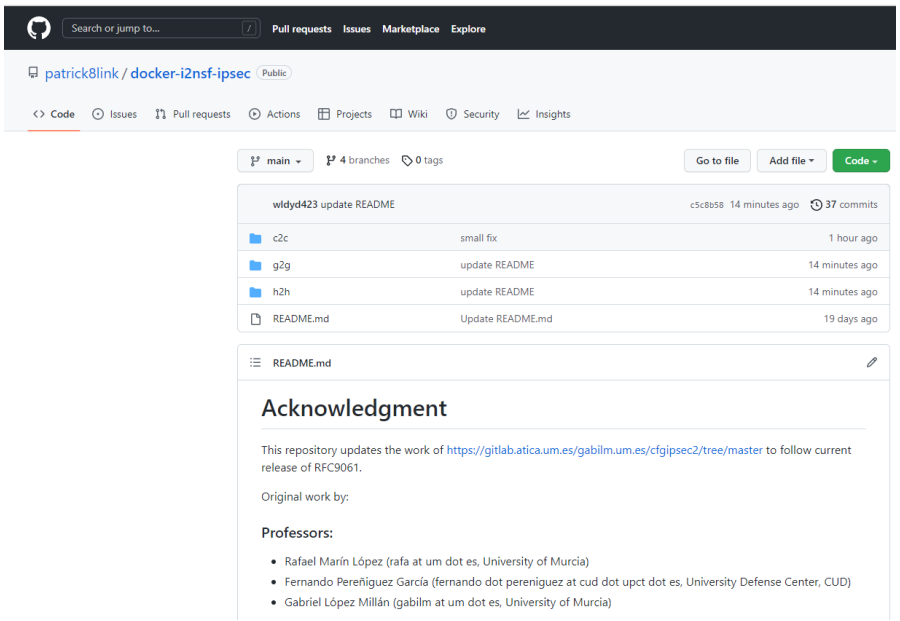
URL:

https://github.com/patrick8link/i2nsf-ipsec



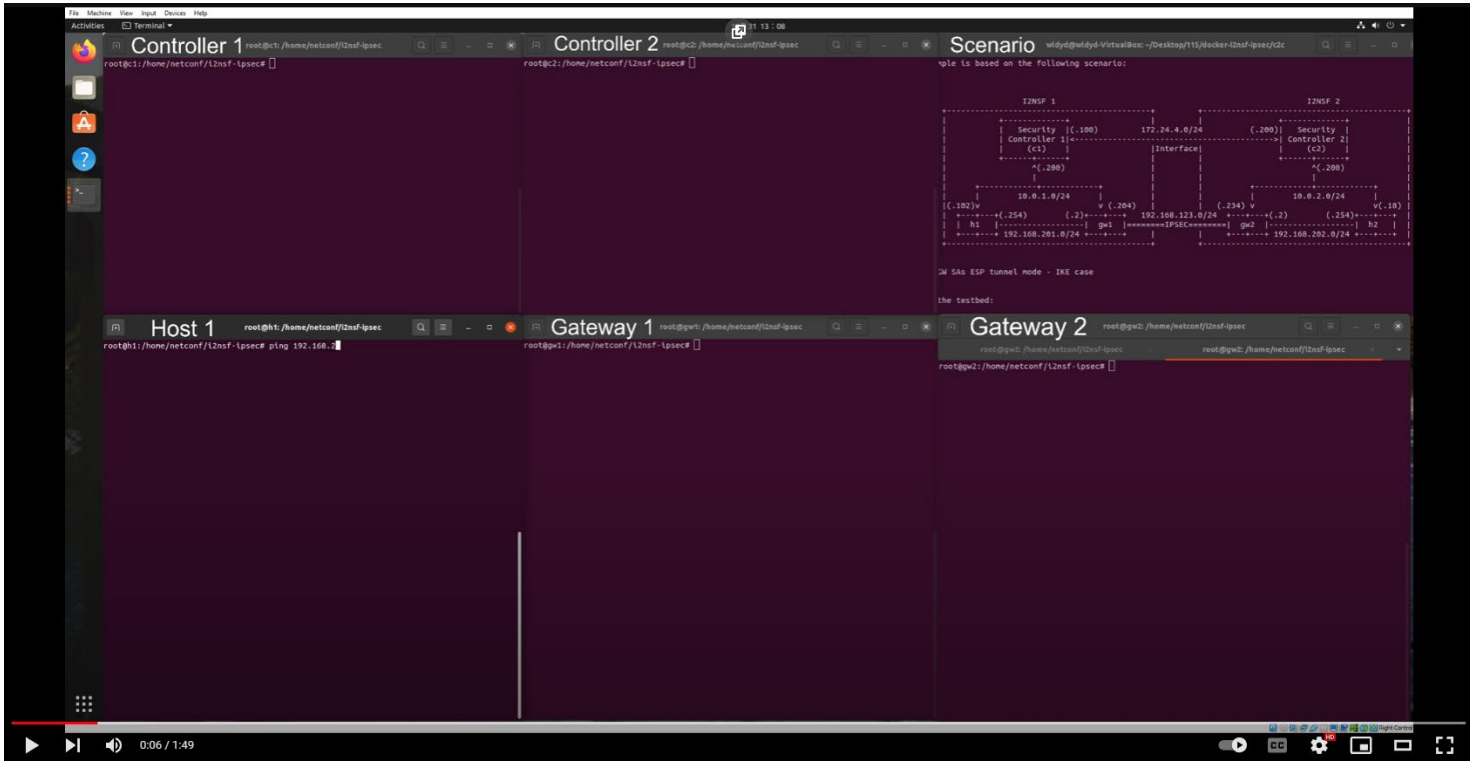
URL:

https://github.com/patrick8link/docker-i2nsf-ipsec



Demonstration Video Clip at YouTube

URL: https://www.youtube.com/watch?v=l-bSMxOs7zw



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)**
- **Jeonghyeon Kim (SKKU)**
- **Jiyong Uhm (SKKU)**
- **Mose Gu (SKKU)**

Hackathon Team Photo