I2NSF Capability YANG Data Model
(draft-hares-i2nsf-capability-data-model-01)

IETF 98, Chicago, US
Mar. 27, 2017

Susan hares, Robert Moskowitz, Liang Xia,
Jinyong Tim Kim, and Jaehoon Paul Jeong.
Security Controller doesn't know where NSFs is located

Capability YANG Data Model

- Network Control (10.0.0.3)
  - Firewall
- DDoS Mitigator (10.0.0.5)
  - Syn Flood
  - ICMP Flood
- Content Control (10.0.0.6)
  - Game Filter
  - P2P Filter
- DDoS Mitigator (10.0.0.7)
  - UDP Flood
  - IP-Frag Flood
Capability YANG Data Model (2/2)

Security Controller

Query based on the Capability description

NSF's information (e.g., NSF2)

Capability-based NSF discovery

Capability-based packet forwarding request

PFH includes
- Capability description

PFH: Packet Forwarding Header
NSFF: NSF Forwarder
Introduction

• This draft is an updated version from draft-hares-i2nsf-capability-yang-00.

• This draft introduces YANG data model for security controller to express and discover the capabilities of NSF devices.

• This YANG model can also be used by the list of I2NSF capabilities that can be controlled by security controller.
Update of Version 01

- Types of IP Addresses used by NSF devices
  - IPv4 address
  - IPv6 address

- Enhanced Content Security Control
  - dns filter
  - ftp filter
  - games filter
  - rpc filter
  - sql filter
  - telnet filter
  - tftp filter
Types of IP Addresses used by NSF devices

OLD

module : ietf-i2nsf-capability
  +=rw sec-ctl-capabilities
  +=rw nsf-capabilities
  +=rw nsf* [nsf-name]
  +=rw nsf-name_string
  +=rw nsf-address inet:ipv4-address
  +=rw net-sec-control-capabilities
  | +=rw i2nsf-net-sec-control-caps
  +=rw con-sec-control-capabilities
  | +=rw i2nsf-con-sec-control-caps
  +=rw attack-mitigation-capabilities
  | +=rw i2nsf-attack-mitigation-control-caps
  +=rw it-resource
  | +=rw i2nsf-it-resources

NEW

module : ietf-i2nsf-capability
  +=rw sec-ctl-capabilities
  +=rw nsf-capabilities
  +=rw nsf* [nsf-name]
  +=rw nsf-name_string
  +=rw nsf-address inet:ipv4-address
  +=rw (nsf-address-type)?
  | +=rw ipv4-address inet:ipv4-address
  | +=rw ipv6-address inet:ipv6-address
  +=rw net-sec-control-capabilities
  | +=rw i2nsf-net-sec-control-caps
  +=rw con-sec-control-capabilities
  | +=rw i2nsf-con-sec-control-caps
  +=rw attack-mitigation-capabilities
  | +=rw i2nsf-attack-mitigation-control-caps
  +=rw it-resource
  | +=rw i2nsf-it-resources
Enhanced Content Security Control

```plaintext
+++rw dns-filter
   ++rw dns-filter-support?  boolean
   +++rw dns-filter-fcn*  [dns-filter-name]
      +++rw dns-filter-fcn-name  string  //std or vendor name
++rw ftp-filter
   ++rw ftp-filter-support?  boolean
   +++rw ftp-filter-fcn*  [ftp-filter-fcn-name]
      +++rw ftp-filter-fcn-name  string  //std or vendor name
++rw games-filter
   ++rw games-filter-support?  boolean
   +++rw games-filter-fcn*  [games-filter-fcn-name]
      +++rw games-filter-fcn-name  string  //std or vendor name
++rw p2p-filter
   ++rw p2p-filter-support?  boolean
   +++rw p2p-filter-fcn*  [p2p-filter-fcn-name]
      +++rw p2p-filter-fcn-name  string  //std or vendor name
++rw rpc-filter
   ++rw rpc-filter-support?  boolean
   +++rw rpc-filter-fcn*  [rpc-filter-fcn-name]
      +++rw rpc-filter-fcn-name  string  //std or vendor name
++rw sql-filter
   ++rw sql-filter-support?  boolean
   +++rw sql-filter-fcn*  [sql-filter-fcn-name]
      +++rw sql-filter-fcn-name  string  //std or vendor name
++rw telnet-filter
   ++rw telnet-filter-support?  boolean
   +++rw telnet-filter-fcn*  [telnet-filter-fcn-name]
      +++rw telnet-filter-fcn-name  string  //std or vendor name
++rw tftp-filter
   ++rw tftp-filter-support?  boolean
   +++rw tftp-filter-fcn*  [tftp-filter-fcn-name]
      +++rw tftp-filter-fcn-name  string  //std or vendor name
```
Next Step

• We will implement and test a prototype to use the enhanced data YANG model:
  – Types of IP Addresses for NSFVs,
  – Content Security Control, and
  – Attack Mitigation Control.