Interface to Network Security Functions
Problem Statement

July 2015

Linda Dunbar (linda.dunbar@huawei.com)

Myo Zarny (Myo.Zarny@gs.com)

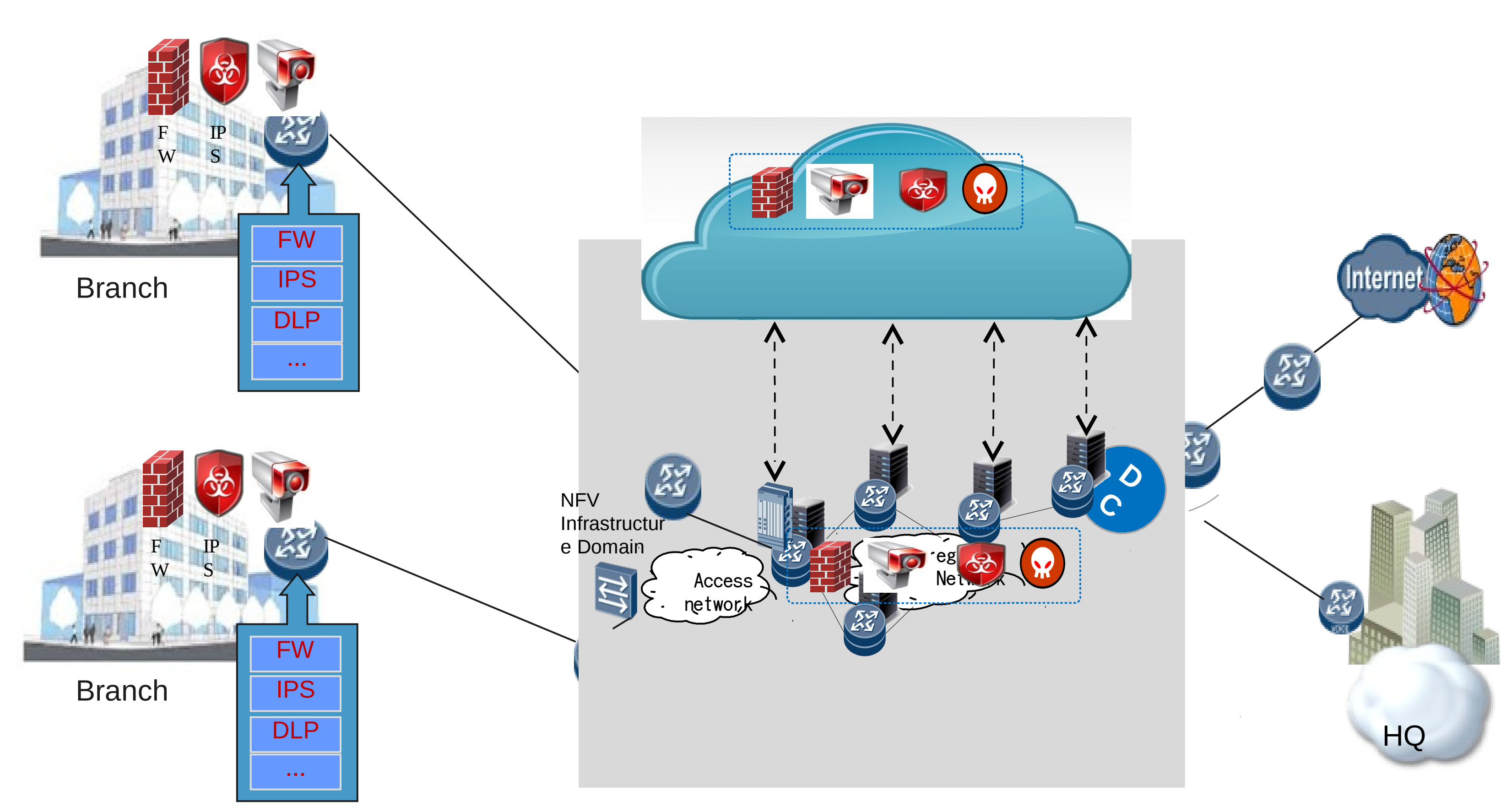
Christian Jacquenet (Christian.jacquenet@orange.com)

Mohamed Boucadair (mohamed.boucadair@orange.com)

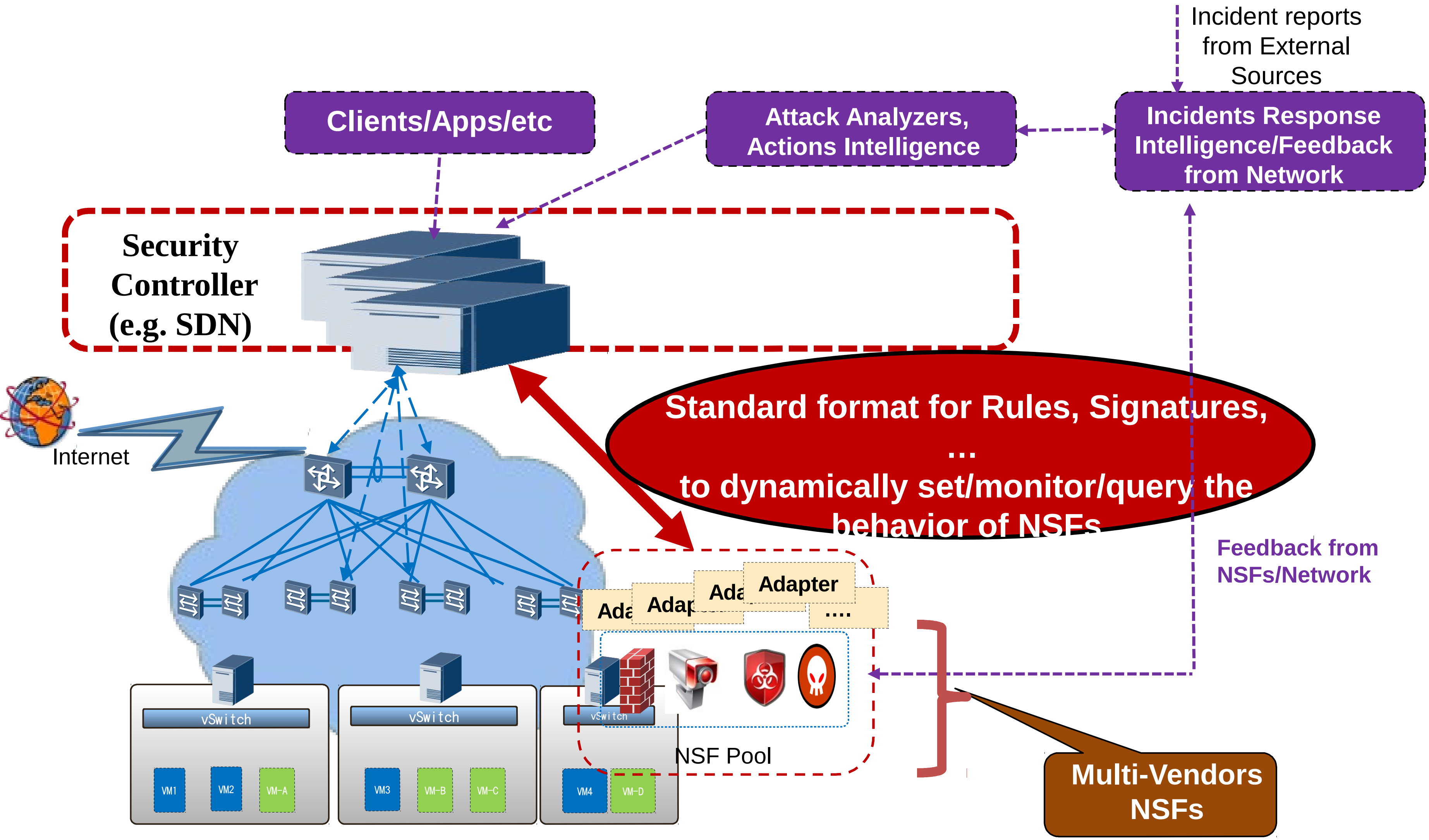
Shaibal Chakrabarty (shaibalc@us-ignite.org)

Multi-vendor & Multi-Types of NSFs

To be managed



Automation of the NSFs’ control & monitor



Incident reports from External Sources

Clients/Apps/etc

Attack Analyzers, Actions Intelligence

Incidents Response Intelligence/Feedback from Network

Security Controller (e.g. SDN)

Standard format for Rules, Signatures, …

to dynamically set/monitor/query the behavior of NSFs

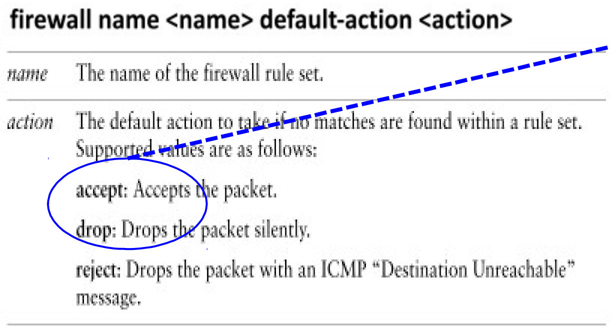
Feedback from NSFs/Network

Multi-Vendors NSFs

It doesn’t require NFV, it doesn’t require provider domain. I2NSF is to facilitate automation

Different vendor → Different Provisioning Formats

Vendor A



firewall name <name> default-action <action>

name The name of the firewall rule set.

action The default action to take if no matches are found within a rule set. Supported values are as follows:

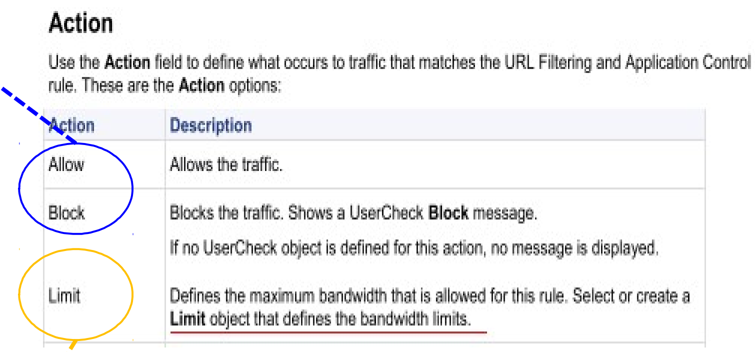
accept: Accepts the packet.

drop: Drops the packet silently.

reject: Drops the packet with an ICMP "Destination Unreachable" message.

same function , Different name

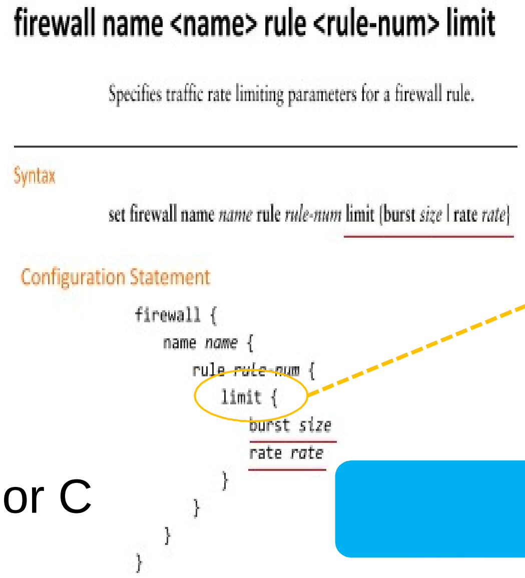
Vendor B



Action

Use the **Action** field to define what occurs to traffic that matches the URL Filtering and Application Control rule. These are the **Action** options:

Action	Description
Allow	Allows the traffic.
Block	Blocks the traffic. Shows a UserCheck **Block** message. If no UserCheck object is defined for this action, no message is displayed.
Limit	Defines the maximum bandwidth that is allowed for this rule. Select or create a **Limit** object that defines the bandwidth limits.



firewall name <name> rule <rule-num> limit

Specifies traffic rate limiting parameters for a firewall rule.

Syntax

set firewall name *name* rule *rule-num* limit [burst *size* | rate *rate*]

Configuration Statement

```
firewall {
    name name {
        rule rule-num {
            limit {
                burst size
                rate rate
            }
        }
    }
}
```

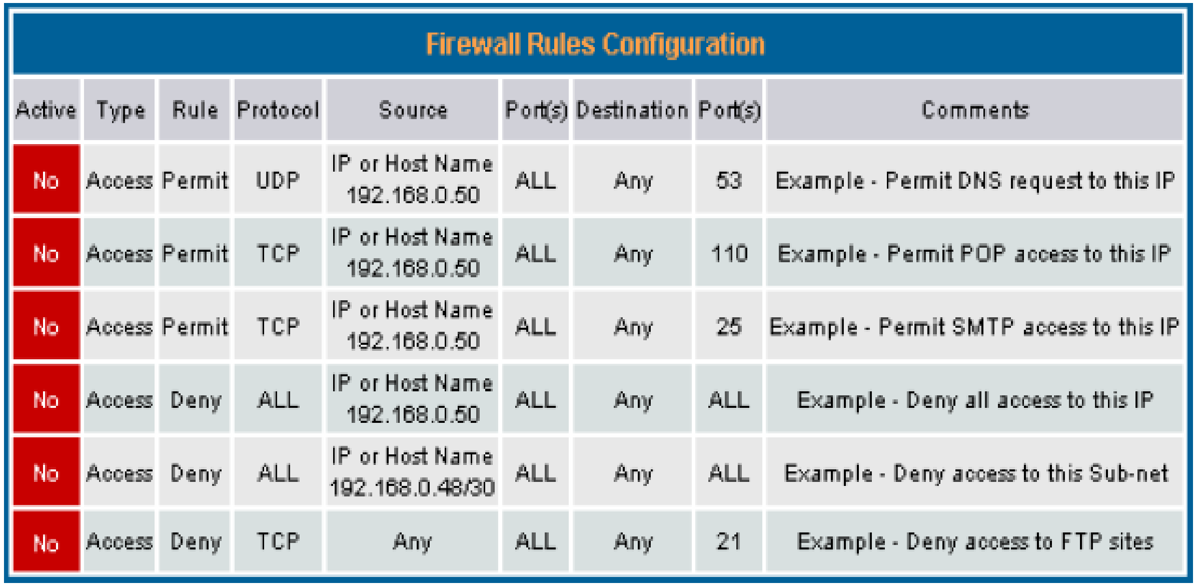
Vendor C

same parameter , different Settings

Difficult to achieve automated deployment.

FW configuration: ports & links based

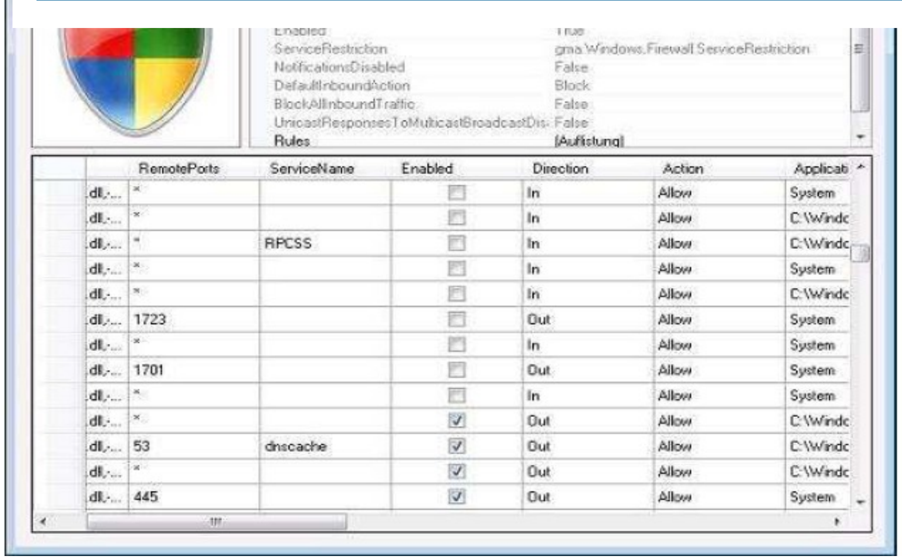
Virtual Networks Needs Group Policies & Abstraction. Need standard format for automation



Firewall Rules Configuration

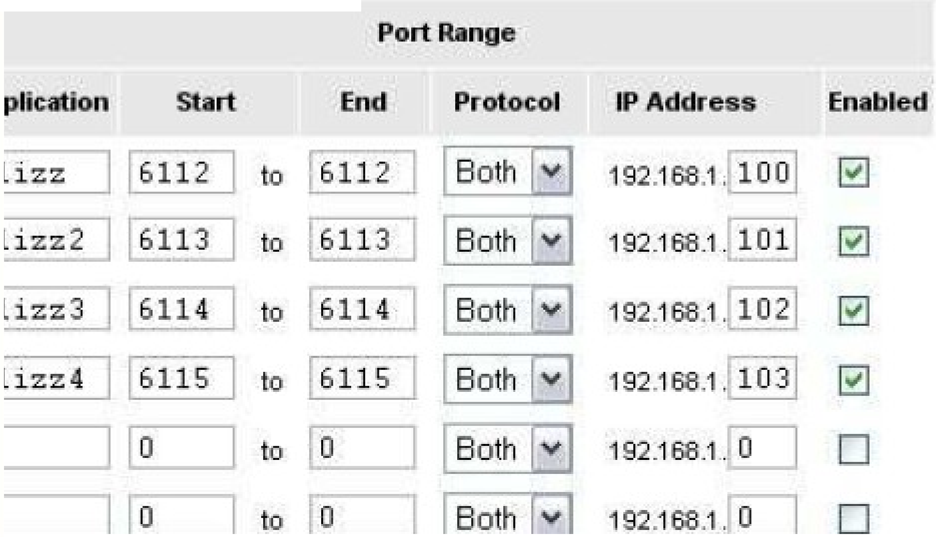
Active	Type	Rule	Protocol	Source	Port(s)	Destination	Port(s)	Comments
No	Access	Permit	UDP	IP or Host Name 192.168.0.50	ALL	Any	53	Example - Permit DNS request to this IP
No	Access	Permit	TCP	IP or Host Name 192.168.0.50	ALL	Any	110	Example - Permit POP access to this IP
No	Access	Permit	TCP	IP or Host Name 192.168.0.50	ALL	Any	25	Example - Permit SMTP access to this IP
No	Access	Deny	ALL	IP or Host Name 192.168.0.50	ALL	Any	ALL	Example - Deny all access to this IP
No	Access	Deny	ALL	IP or Host Name 192.168.0.48/30	ALL	Any	ALL	Example - Deny access to this Sub-net
No	Access	Deny	TCP	Any	ALL	Any	21	Example - Deny access to FTP sites

Need standard method to express commonly used rules for virtual networks and groups



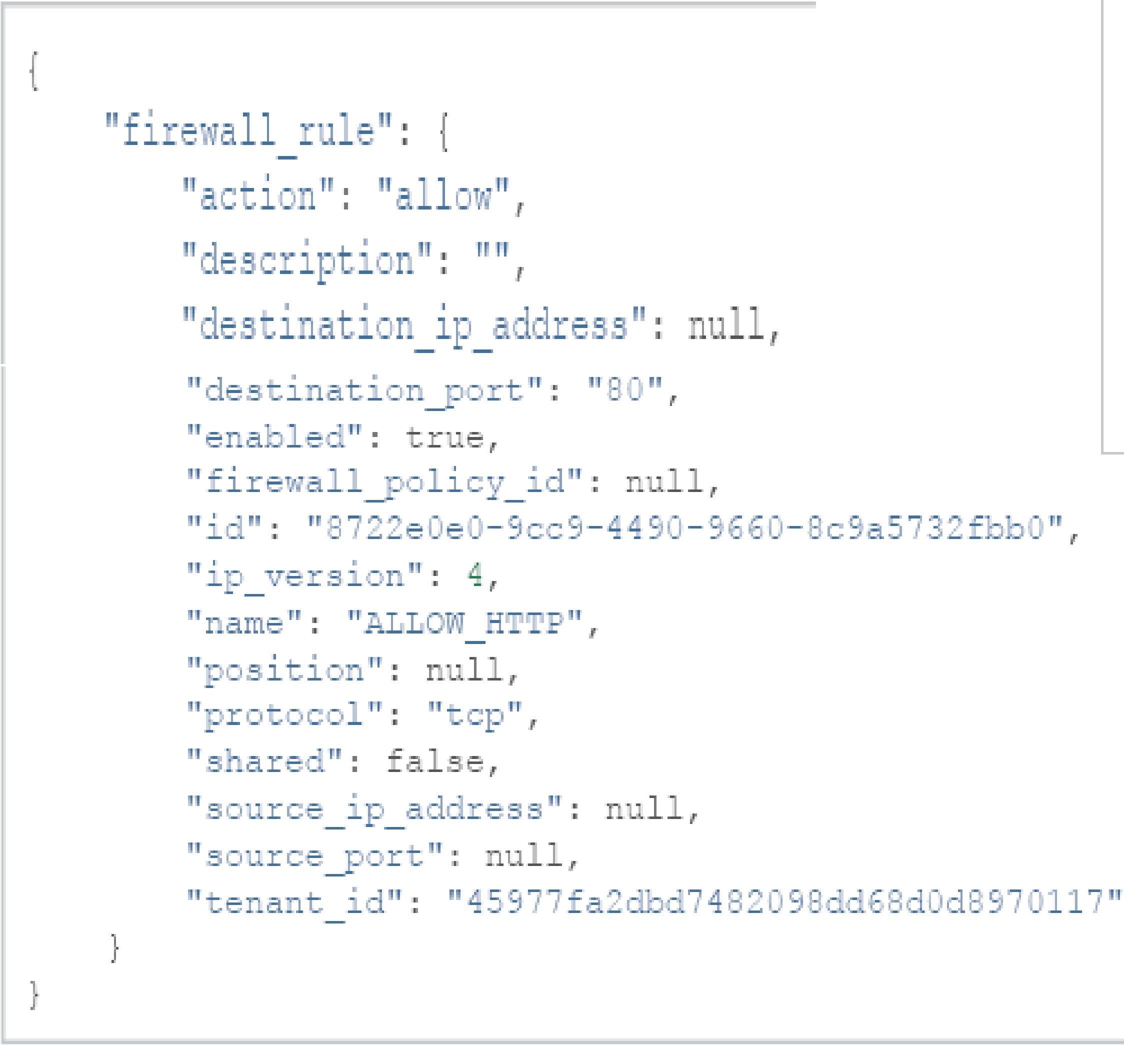
Enabled	True
ServiceRestriction	gma.Windows.Firewall.ServiceRestriction
NotificationsDisabled	False
DefaultInboundAction	Block
BlockAllInboundTraffic	False
UnicastResponsesToMulticastBroadcastDis	False
Rules	(Auflistung)

	RemotePorts	ServiceName	Enabled	Direction	Action	Applicat
.dll,-...	*		□	In	Allow	System
.dll,-...	*		□	In	Allow	C:\Windc
.dll,-...	*	RPCSS	□	In	Allow	C:\Windc
.dll,-...	*		□	In	Allow	System
.dll,-...	*		□	In	Allow	C:\Windc
.dll,-...	1723		□	Out	Allow	System
.dll,-...	*		□	In	Allow	System
.dll,-...	1701		□	Out	Allow	System
.dll,-...	*		□	In	Allow	System
.dll,-...	*		☑	Out	Allow	C:\Windc
.dll,-...	53	dnscache	☑	Out	Allow	C:\Windc
.dll,-...	*		☑	Out	Allow	C:\Windc
.dll,-...	445		☑	Out	Allow	System

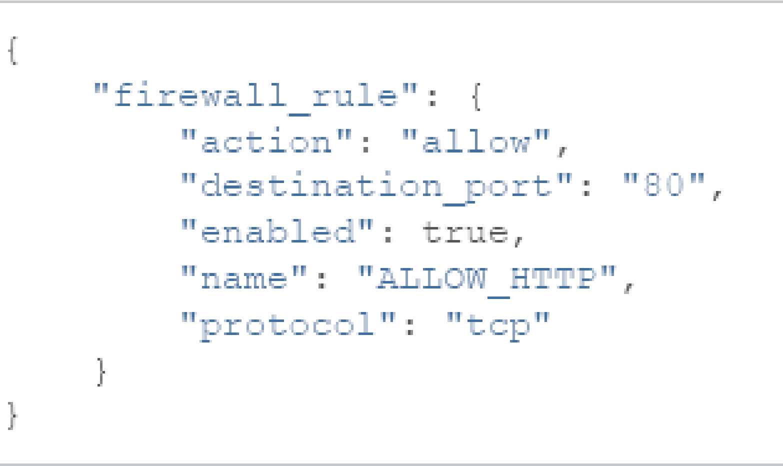


	Port Range					
plication	Start		End	Protocol	IP Address	Enabled
izz	6112	to	6112	Both	192.168.1. 100	☑
izz2	6113	to	6113	Both	192.168.1. 101	☑
izz3	6114	to	6114	Both	192.168.1. 102	☑
izz4	6115	to	6115	Both	192.168.1. 103	☑
	0	to	0	Both	192.168.1. 0	□
	0	to	0	Both	192.168.1. 0	□

OpenStack FWaaS Rules Configuration



```
{
    "firewall_rule": {
        "action": "allow",
        "description": "",
        "destination_ip_address": null,
        "destination_port": "80",
        "enabled": true,
        "firewall_policy_id": null,
        "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
        "ip_version": 4,
        "name": "ALLOW_HTTP",
        "position": null,
        "protocol": "tcp",
        "shared": false,
        "source_ip_address": null,
        "source_port": null,
        "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
}
```



```
{
    "firewall_rule": {
        "action": "allow",
        "destination_port": "80",
        "enabled": true,
        "name": "ALLOW_HTTP",
        "protocol": "tcp"
    }
}
```

Summary of I2NSF Problems

- **3.1. Challenges Facing Security Service Providers**
  - 3.1.1. Diverse types of Security Functions
  - 3.1.2. Diverse Interfaces to Control NSFs
  - 3.1.3. Diverse Interface to monitor the behavior of NSFs
  - 3.1.4. More Distributed NSFs and vNSFs
  - 3.1.5. More Demand to Control NSFs Dynamically
  - 3.1.6. Demand for multi-tenancy to control and monitor NSFs.
  - 3.1.7. Lack of Characterization of NSFs and Capability Exchange
  - 3.1.8. Lack of mechanism for NSFs to utilize external profiles
- 3.2. Challenges Facing Customers
  - 3.2.1. NSFs from heterogeneous administrative domains
  - 3.2.2. Today’s Control Requests are Vendors Specific
  - 3.2.3. Difficulty to Monitor the Execution of Desired Policies
- **3.3. Difficulty to Validate Policies across Multiple Domains**
- **3.4. Lack of Standard Interface to Inject Feedback to NSF**
- **3.5. Lack of Standard Interface for Capability Negotiation**

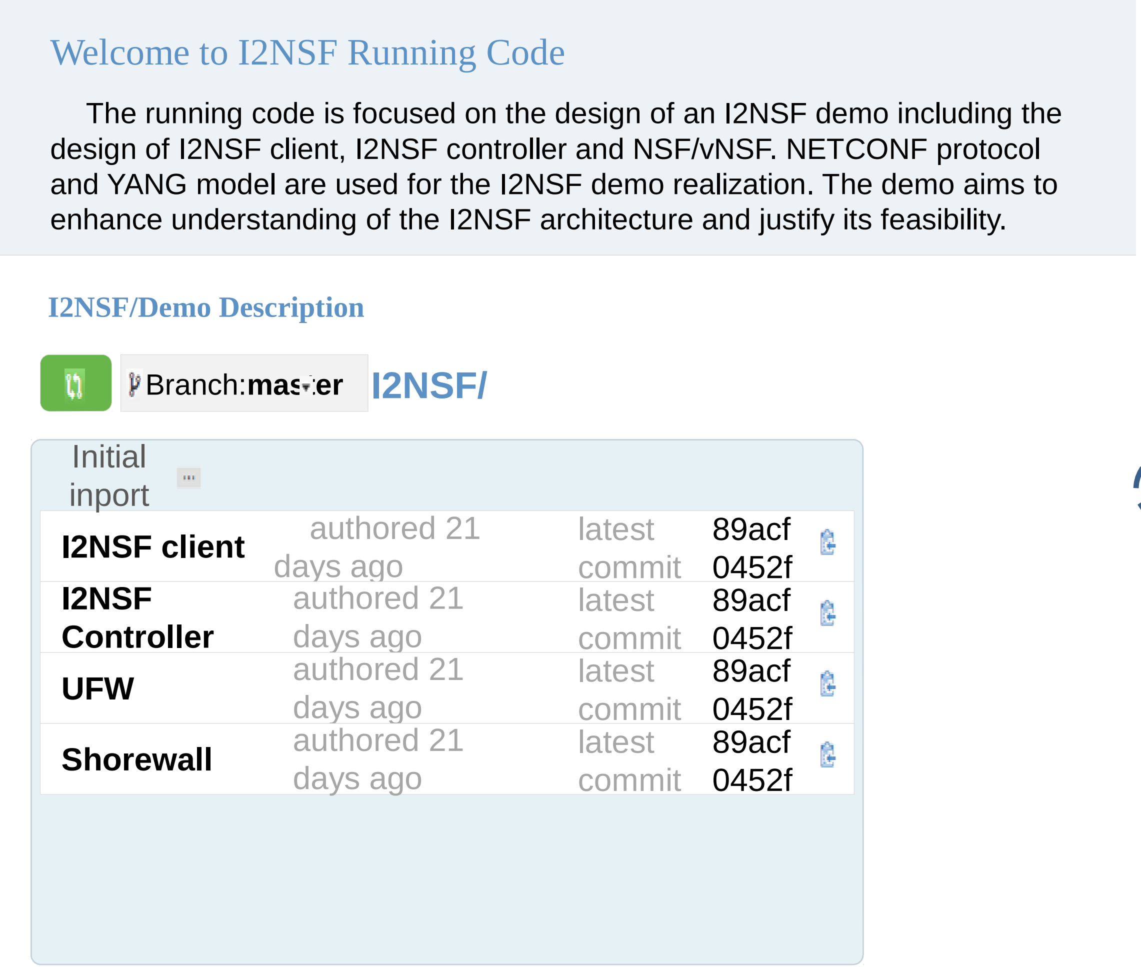
Goal of I2NSF

  - Specify and standardize corresponding information and data models for the dynamic provisioning, querying, monitoring of flow based network security functions

  - Define Policy Enforcement Schemes for automated delivery of security services, Design feedback mechanisms for security service fulfillment and assurance purposes

- Other aspects of NSFs, such as device or network provisioning and configuration, are out of scope

Steps towards Open Source



Welcome to I2NSF Running Code

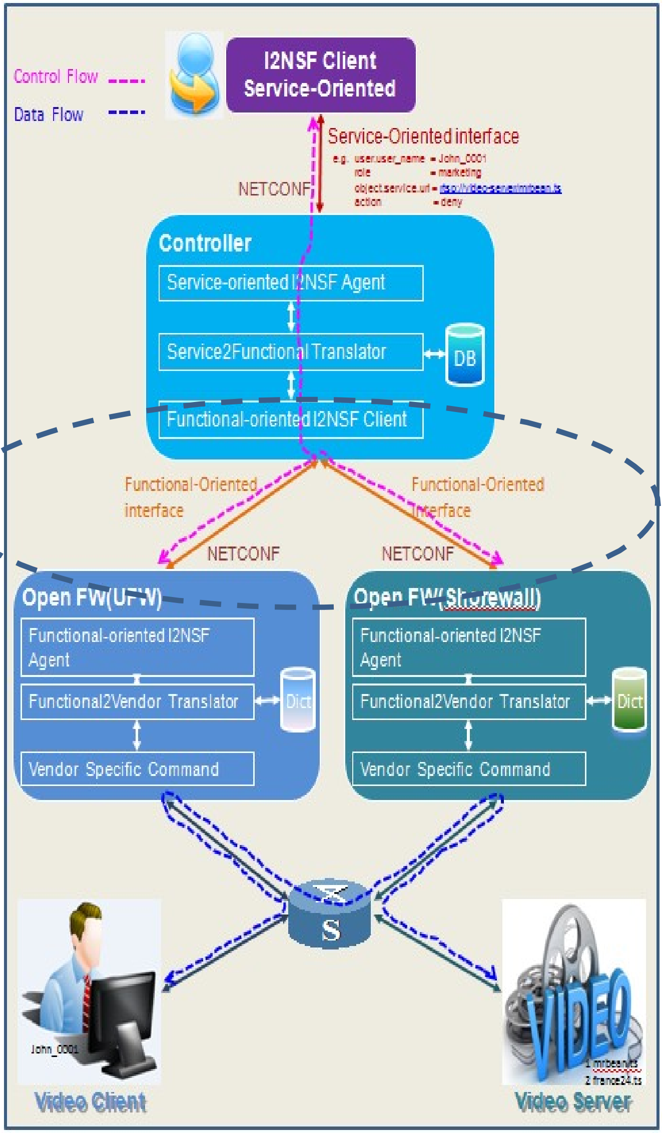
The running code is focused on the design of an I2NSF demo including the design of I2NSF client, I2NSF controller and NSF/vNSF. NETCONF protocol and YANG model are used for the I2NSF demo realization. The demo aims to enhance understanding of the I2NSF architecture and justify its feasibility.

I2NSF/Demo Description

Branch:master I2NSF/

Initial inport …

I2NSF client	authored 21 days ago	latest commit	89acf 0452f
I2NSF Controller	authored 21 days ago	latest commit	89acf 0452f
UFW	authored 21 days ago	latest commit	89acf 0452f
Shorewall	authored 21 days ago	latest commit	89acf 0452f


Control Flow
Data Flow
I2NSF Client Service-Oriented
Service-Oriented interface
e.g. user.user_name = John_0001
role = marketing
object.service.url = rtsp://video-server/mrbean.ts
action = deny
NETCONF
Controller
Service-oriented I2NSF Agent
Service2Functional Translator
DB
Functional-oriented I2NSF Client
Functional-Oriented interface
Functional-Oriented interface
NETCONF
NETCONF
Open FW(UFW)
Functional-oriented I2NSF Agent
Functional2Vendor Translator
Dict
Vendor Specific Command
Open FW(Shorewall)
Functional-oriented I2NSF Agent
Functional2Vendor Translator
Dict
Vendor Specific Command
S
John_0001
Video Client
VIDEO
1 mrbean.ts
2 france24.ts
Video Server