Network Security as a Service (NSaaS)

July 2014

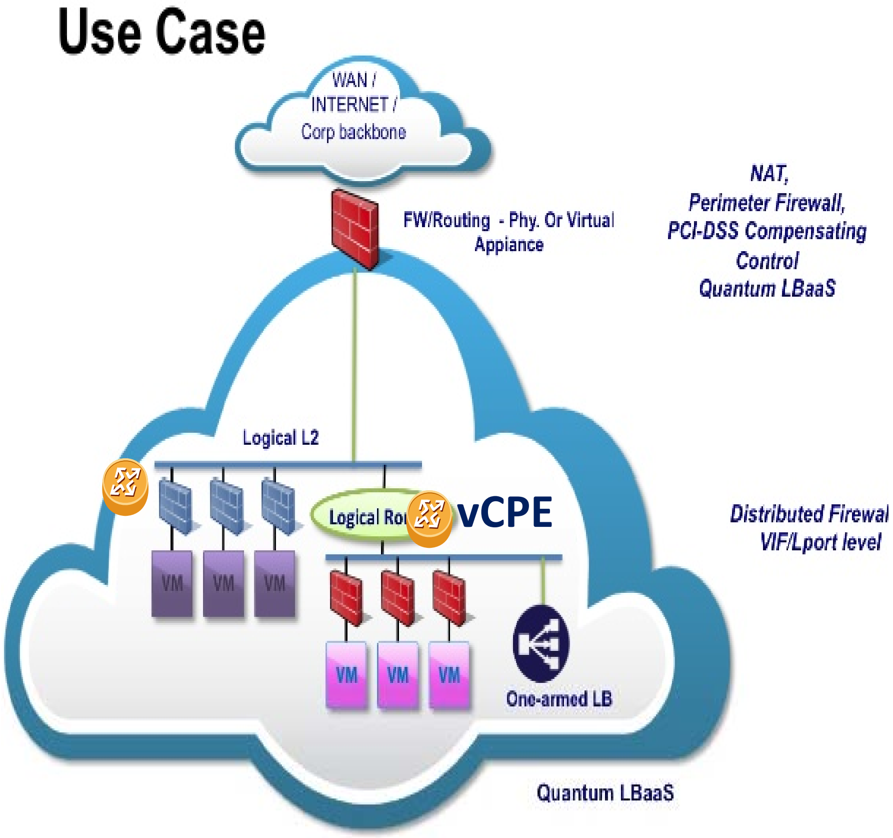
Linda Dunbar (linda.dunbar@huawei.com)

Myo Zarny (Myo.Zarny@gs.com)

Christian Jacquenet (Christian.jacquenet@orange.com)

Shaibal Chakrabarty (shaibalc@us-ignite.org)

Use Case: Virtual Firewall Function for vCPE in multi-tenant DC


Use Case
WAN / INTERNET / Corp backbone
FW/Routing - Phy. Or Virtual Appiance
NAT, Perimeter Firewall, PCI-DSS Compensating Control Quantum LBaaS
Logical L2
Logical Rou
vCPE
VM
VM
VM
VM
VM
VM
One-armed LB
Distributed Firewal VIF/Lport level
Quantum LBaaS

Resource Model

Firewalls - A logical instance of a firewall embodying a Firewall Policy

Firewall Policies - An ordered collection of Firewall Rules

Firewall Rules - N-tuple that generically models firewall rules

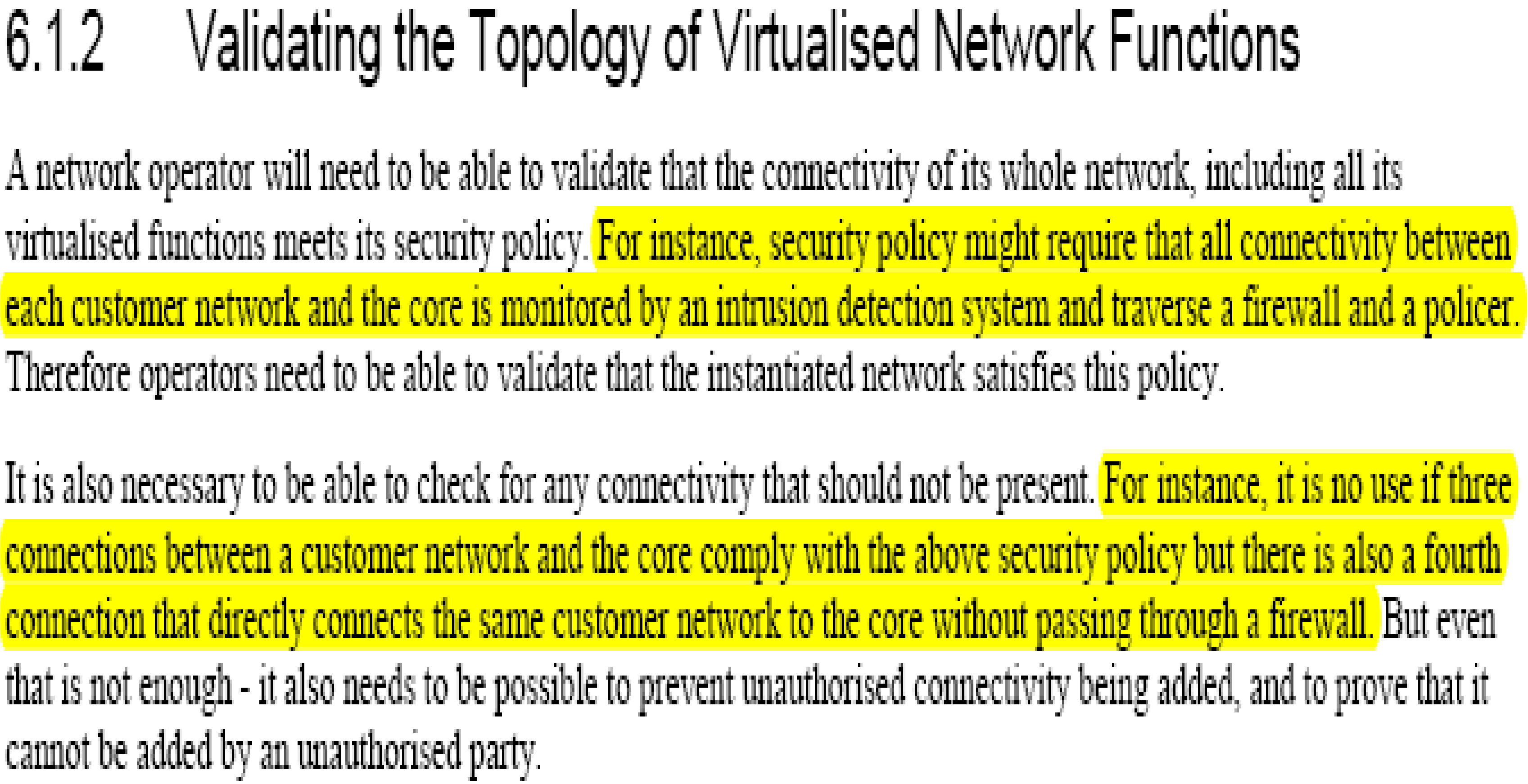
Workflow

Firewall Rules are defined and Firewall Policy is composed

Firewall Policy is audited (audit process in not modeled here)

Tenant creates Firewall instance using Firewall Policy

Problems identified by ETSI NFV

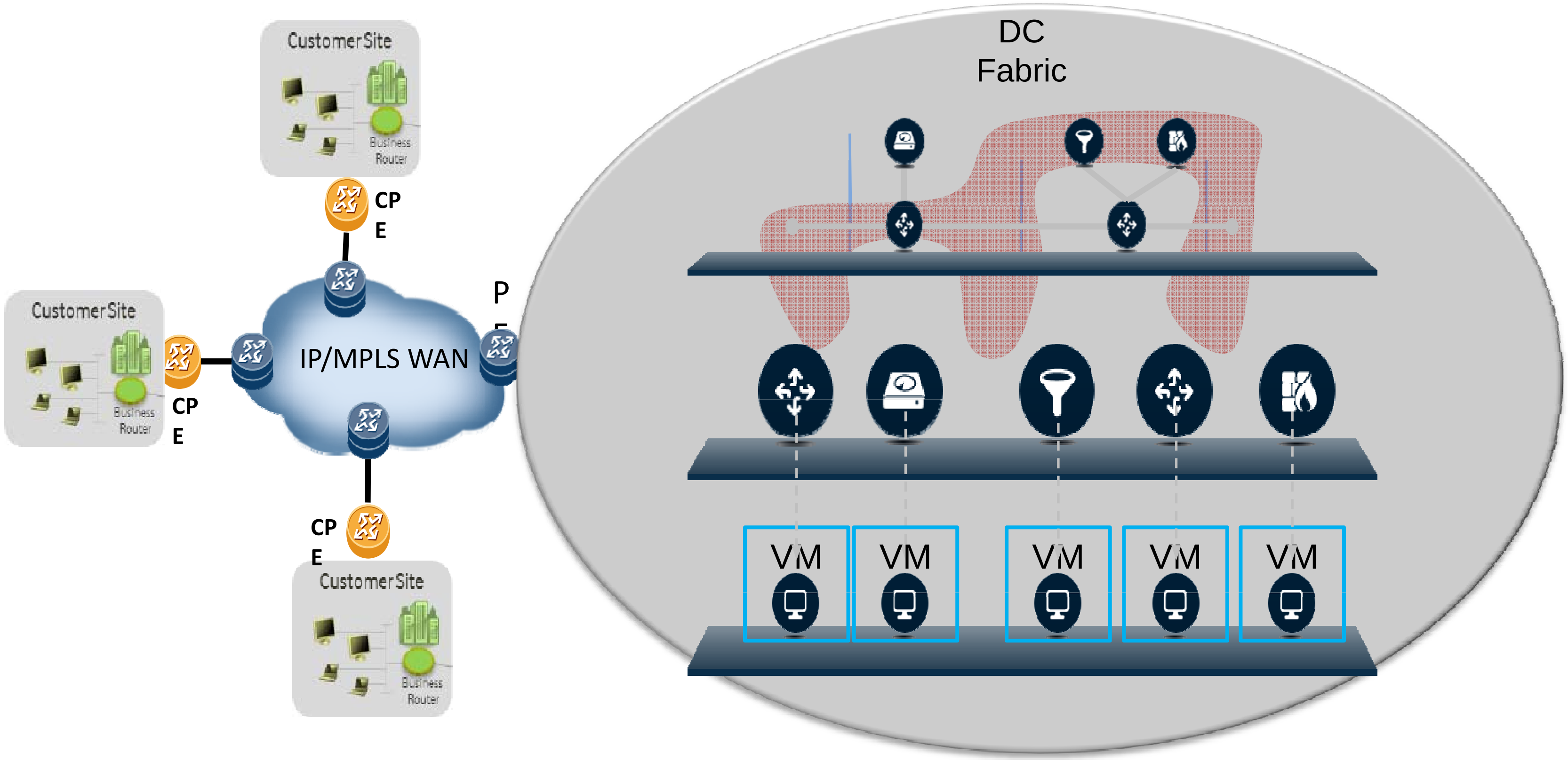


6.1.2 Validating the Topology of Virtualised Network Functions

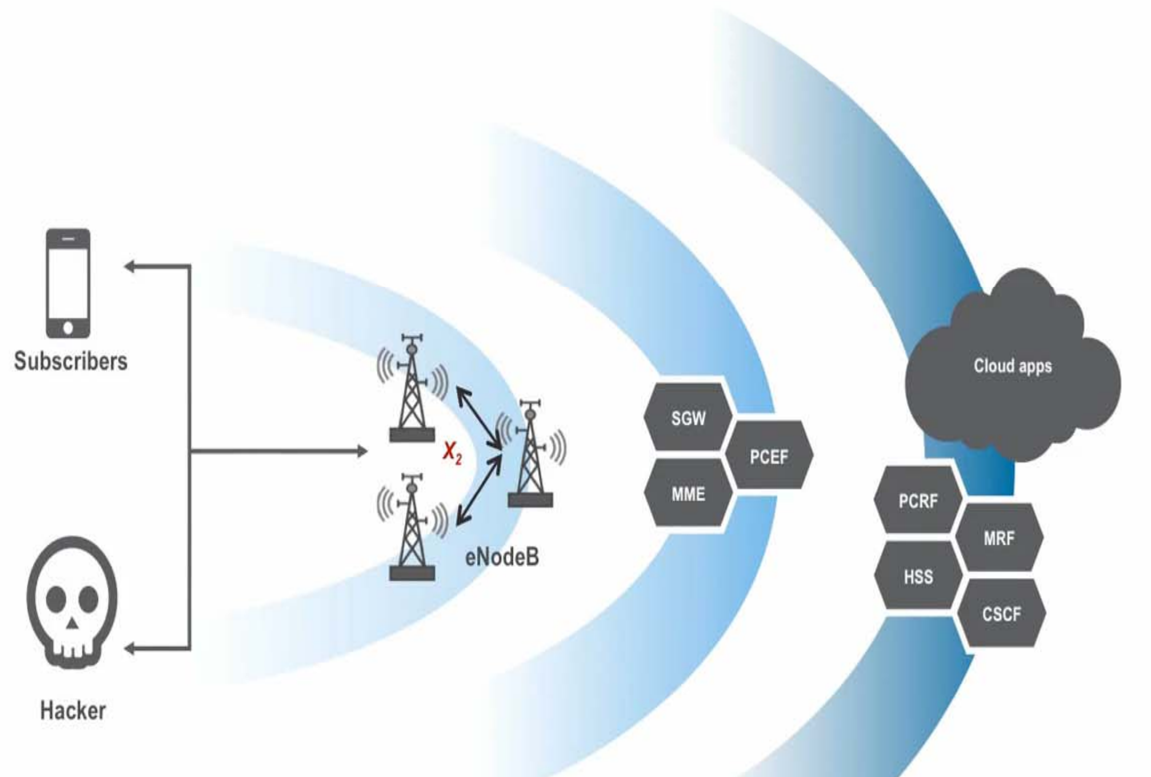
A network operator will need to be able to validate that the connectivity of its whole network, including all its virtualised functions meets its security policy. For instance, security policy might require that all connectivity between each customer network and the core is monitored by an intrusion detection system and traverse a firewall and a policer. Therefore operators need to be able to validate that the instantiated network satisfies this policy.

It is also necessary to be able to check for any connectivity that should not be present. For instance, it is no use if three connections between a customer network and the core comply with the above security policy but there is also a fourth connection that directly connects the same customer network to the core without passing through a firewall. But even that is not enough - it also needs to be possible to prevent unauthorised connectivity being added, and to prove that it cannot be added by an unauthorised party.

Virtual Network Functions: requested by clients based on demand



Evolving Threats to Mobile Networks



Device

Battery drain attacks
Mobile malware and bots

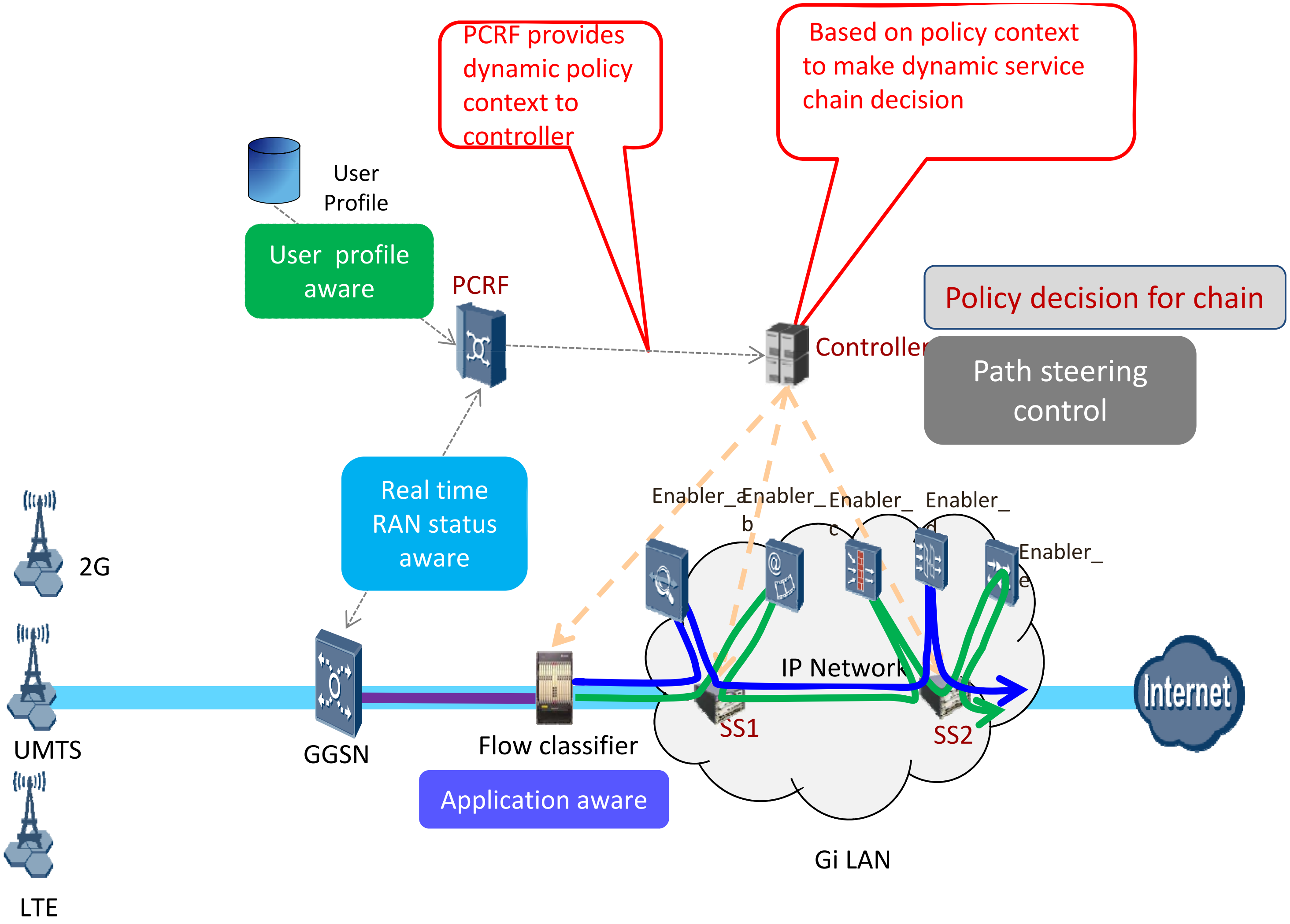
Network

RAN resource exhaustion
Revenue leakage – weak APN controls
Terms & Conditions Violations

Application

Server-side malware
Application DDoS

Mobile Network requests Security functions from Backhaul provider



PCRF provides dynamic policy context to controller

Based on policy context to make dynamic service chain decision

Policy decision for chain

Path steering control

User profile aware

Real time RAN status aware

Application aware

Security Functions under consideration:

- **The wide acceptance of security functions that are not running on customer premises. For example:**
  - Security as a Service: https://cloudsecurityalliance.org/research/secaas/#_get-involved
  - Firewall as a Service : http://docs.openstack.org/admin-guide-cloud/content/fwaas.html
  - Security has the sense of “long lasting services”. So we don’t have to deal with “On-Demand” oscillation issues.

- **Here are the network functions under consideration:**
  - **Firewall**
  - DDOS/AntiDoS
  - IPS/IDS
  - Access control/Authorization/Authentication
  - Secure Key management
  - Intrusion detection system/ Intrusion prevention system

Objective of the proposed work in IETF:

- The goal of the proposed work is to establish a standardized protocols for clients (or one domain) to view/request/verify security functions from provider (or another domain).
- The proposed protocols between requester and provider can be used for the following scenarios:
  - A Client requests a certain network security function from a provider
  - The provider fulfills the request for example, by instantiating an instance of the service in question, or configures an additional rule in an already provisioned service.

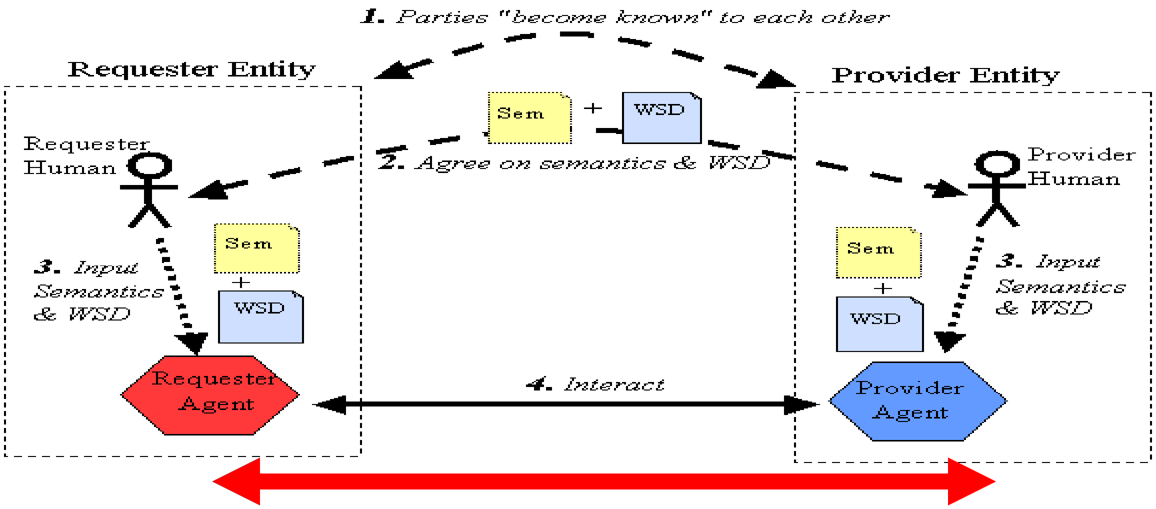
The communication is bidirectional

- ❑ Client not only needs to specify what functions are needed, their associated policies, but also need to receive periodic update from the virtual security functions.
- ❑ such as policies exchanges, steering conditions for higher level security functions

Problem Space

- Issues of the current Cloud-based security solutions:
  - Requiring clients to peer with vendor provided functions hosted in the cloud → Hairpin traffic to far away DC, difficult to maintain consistent software, ..
  - Leave Service providers out of the control
- No available tool to handle: validation of distributed vFW, distributed IPS, or

Candidate Solutions
WebService: common shell



Semantics of Security related functions, e.g. FW

- Need a policy language (YANG model ?) to describe the semantics between requesters and providers.
  - Service function discovery
  - Service function migration policy verification, conflict checking

Other Candidate Solutions

- YANG model, via NetConf protocol, with potential extension.

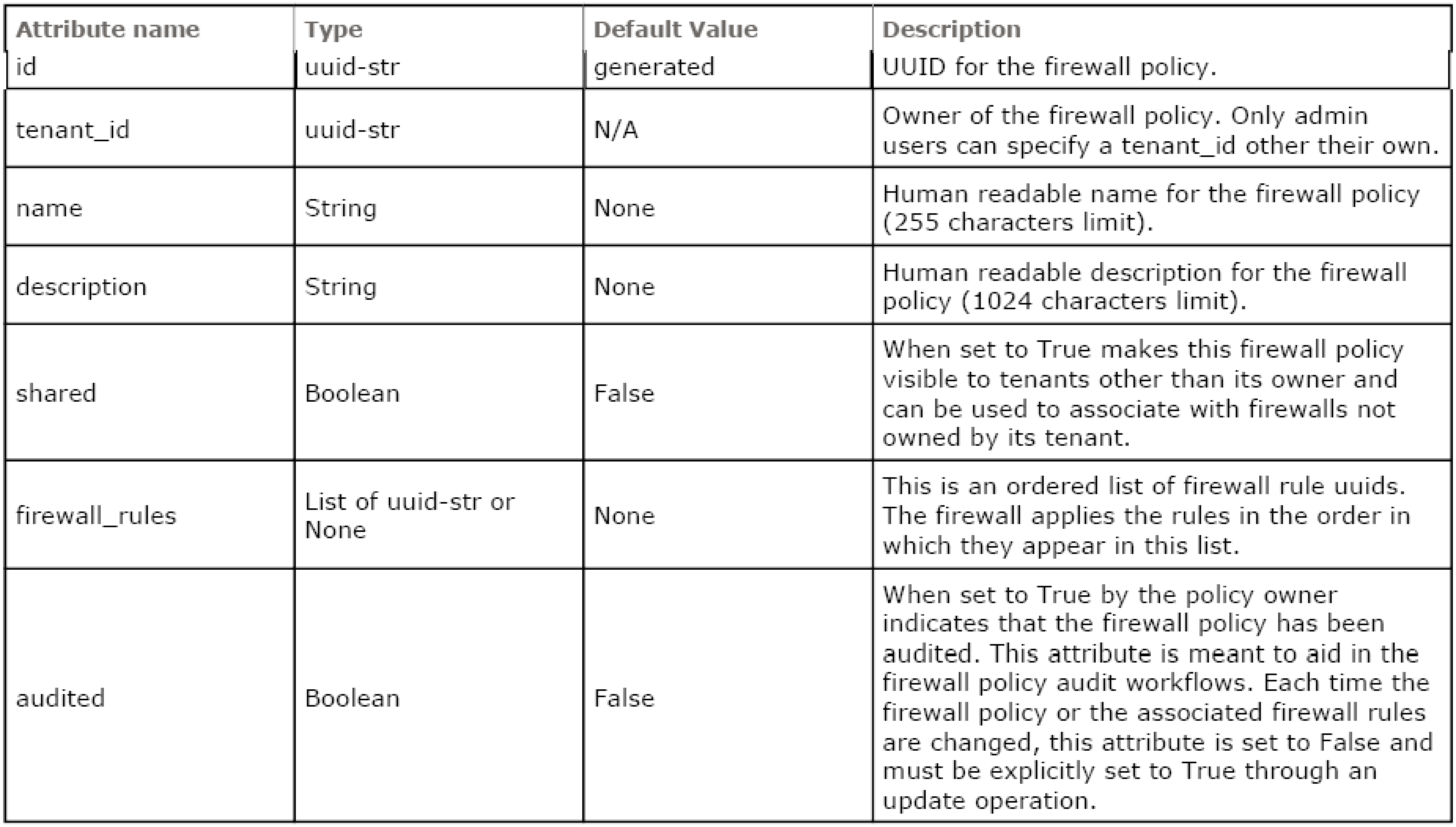
- Diameter, or BGP

- somewhat correlated with RFC7297 “the dynamic service parameter negotiation procedure”.
  - The CPPP template documented in RFC7297 could serve as a basis for the negotiation procedure.
  - the companion CPNP protocol could be a candidate to proceed with the negotiation procedure.

  - The “security as a service” would be a typical example of the kind of (CPP-based) negotiation procedure that could take place between a corporate customer and a service provider.

- Gap analysis is needed.
  - Concrete security specific parameters have to be considered by this proposed work.

FW as a service: potential attributes



Attribute name	Type	Default Value	Description
id	uuid-str	generated	UUID for the firewall policy.
tenant_id	uuid-str	N/A	Owner of the firewall policy. Only admin users can specify a tenant_id other their own.
name	String	None	Human readable name for the firewall policy (255 characters limit).
description	String	None	Human readable description for the firewall policy (1024 characters limit).
shared	Boolean	False	When set to True makes this firewall policy visible to tenants other than its owner and can be used to associate with firewalls not owned by its tenant.
firewall_rules	List of uuid-str or None	None	This is an ordered list of firewall rule uuids. The firewall applies the rules in the order in which they appear in this list.
audited	Boolean	False	When set to True by the policy owner indicates that the firewall policy has been audited. This attribute is meant to aid in the firewall policy audit workflows. Each time the firewall policy or the associated firewall rules are changed, this attribute is set to False and must be explicitly set to True through an update operation.

Security as a Service: Potential attributes

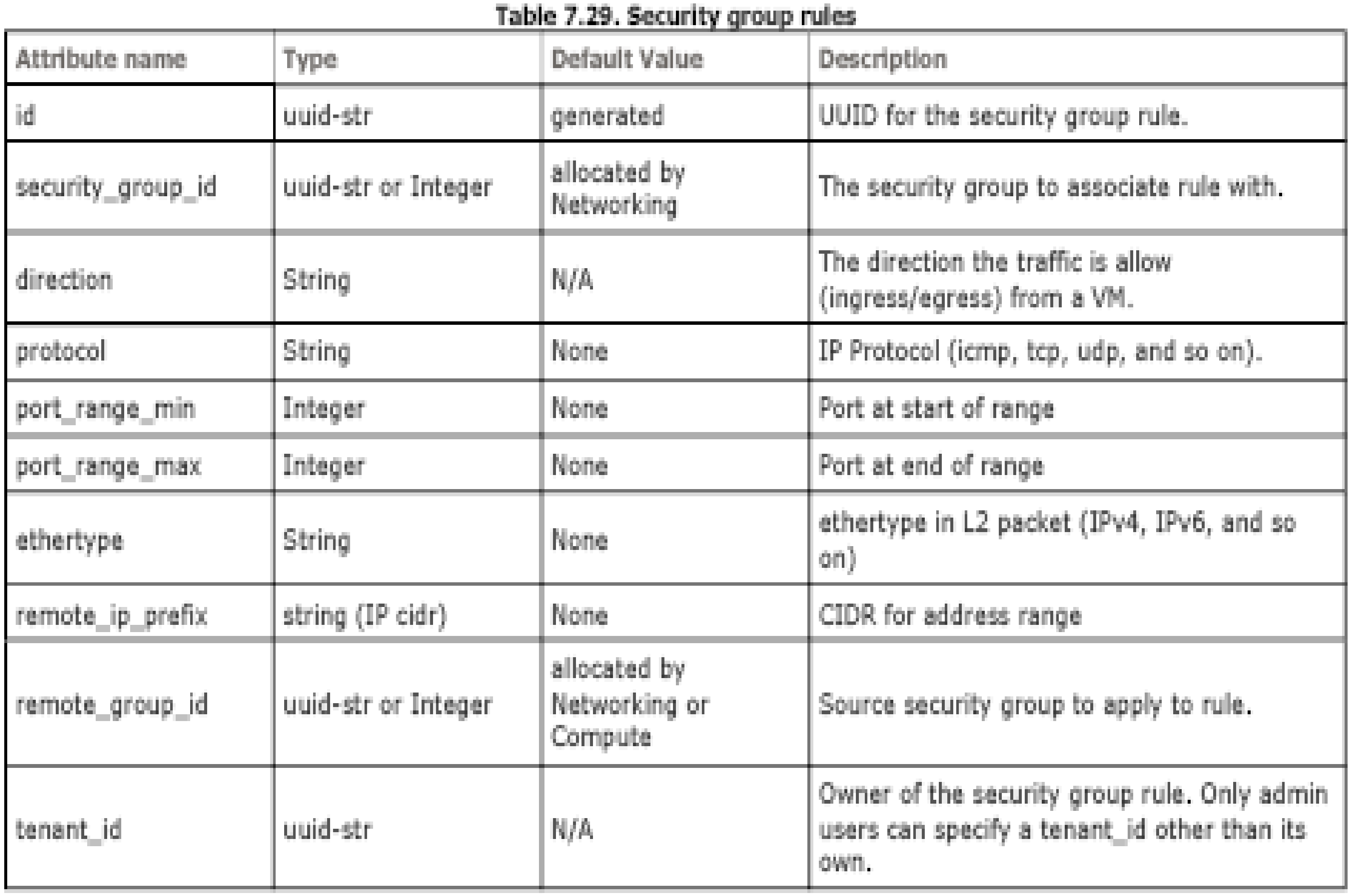


Table 7.29. Security group rules

Attribute name	Type	Default Value	Description
id	uuid-str	generated	UUID for the security group rule.
security_group_id	uuid-str or Integer	allocated by Networking	The security group to associate rule with.
direction	String	N/A	The direction the traffic is allow (ingress/egress) from a VM.
protocol	String	None	IP Protocol (icmp, tcp, udp, and so on).
port_range_min	Integer	None	Port at start of range
port_range_max	Integer	None	Port at end of range
ethertype	String	None	ethertype in L2 packet (IPv4, IPv6, and so on)
remote_ip_prefix	string (IP cidr)	None	CIDR for address range
remote_group_id	uuid-str or Integer	allocated by Networking or Compute	Source security group to apply to rule.
tenant_id	uuid-str	N/A	Owner of the security group rule. Only admin users can specify a tenant_id other than its own.

Relevant Industry initiatives:

- Firewall as a Service by OpenStack
  - OpenStack completed the Firewall as a Service project and specified the set of APIs for Firewall services: http://docs.openstack.org/admin-guide-cloud/content/fwaas_api_abstractions.html
  - OpenStack has defined the APIs for managing Security Groups: http://docs.openstack.org/admin-guide-cloud/content/securitygroup_api_abstractions.html
  - Attributes defined by OpenStack Firewall/Security as a Service will be the basis of the information model for the proposed work at VNFOD IETF initiative.
- Security as a Service by Cloud Security Alliance
  - SaaS by CSA is at the very initiate stage of defining the scope of work.

How NSaaS is different from SACM

SACM:

Security Assessment of End Points

- End points can be routers, switches, clustered DB, installed piece of software
- How to encode that policy in a manner where assessment can be automated
- Example:
  - a Solaris 10 SPARC or Window 7 system used in a environment that requires adherence to a policy of Mission Critical Classified.
  - rules like "The maximum password age must be 30 days" and "The minimum password age must be 1 day"

NSaaS:

Network Security as a Function

- Protocols for edge devices (e.g. vCPE) or clients to request/query/verify Security related functions from Network Providers

  Firewall
  DDOS/Anti-DOS
  Access control/Authorization/Authentication
  Remote identity management
  Secure Key management
  Intrusion Detection System/ Intrusion Prevention System (IDS/IPS)
  Threat detection: Eavesdropping, Trojans, viruses and worms, Malware, etc.

- Example:

  vCPE needs vFW that are hosted in the network.

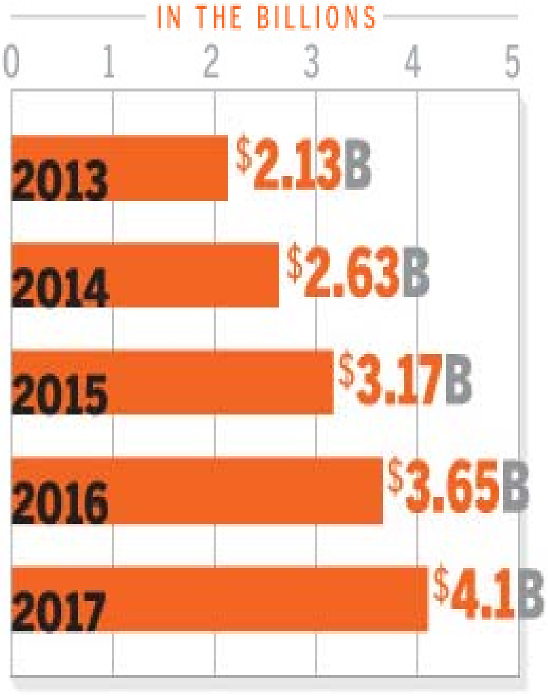
  vCPE provides the “Group Policies” for the vFW, like A can talk to B & C, but B can’t talk to C.

Industry Analysis' opinion:

Gartner is predicting the cloud-based security services market, which includes secure email or web gateways, identity and access management (IAM), remote vulnerability assessment, security information and event management to hit $4.13 billion by 2017.

According to its "Market Trends: Cloud-based Security Services Market, Worldwide, 2014," Gartner is predicting growth is likely to come because of the adoption of these cloud-based security services by small- to-mid-sized business (SMB) in particular. Certain market segments mentioned in the report will see higher overall sales and year-over-year growth.

The cloud-based security services market is rising



SOURCE: GARTNER