I2NSF BOF
Internet-Draft
Intended status: Standards Track
Expires: December 7, 2015

S. Hares
Huawei
H. Moskowitz
HTT Consulting
H. Rozanak

D. Zhang
June 5, 2015

Analysis of Existing work for I2NSF
draft-zhang-gap-analysis-02.txt

Abstract

This document analysis the status of the arts in industries and the existing IETF work/protocols that are relevant to I2NSF.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 7, 2015.

Copyright Notice

Table of Contents

1. Introduction

This document provides an analysis of the gaps in state of the art two industry efforts, IETF and Network Virtualized Functions (NFV) with Software Defined Network (SDN) that I2NSF proposed fills. I2NSF proposes an interoperable means of passing NSF provisioning rules and orchestration information between I2NSF client (security policy decision point), to I2NSF agent (security policy enforcement). An interoperable I2NSF protocol to will aid the orchestration of the provisioning services among different network security functions/ devices.

There are many network security functions being deployed and new ones are popping up with business and application demands. In order to have a concrete context for the protocols discussion, we start with the following network security related functions:

- Firewall
- DDOS/Anti-DOS
- Access control/Authorization/Authentication
- Remote identity management
- Secure Key management
- Intrusion Detection System/ Intrusion Prevention System (IDS/IPS)

It is envisioned that clients of the I2NSF interfaces include management applications, service orchestration systems, network controllers, or user applications that may solicit network security resources.

Various aspects to I2NSF protocol include:

- mechanisms to pass provisioning rules and orchestration information in a common interoperable format,
- The mechanism for clients (applications) to request security filters/provisioning from the I2NSF Agent, write security filters/ provisioning to the I2NSF Agent, and validate information installed on the physically located on I2NSF Agent,
- a means to get change interrupts when security filters change, and
- a means to provide logging of changes to provision information and filters.

2. Terms and Definitions

2.1. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119, BCP 14 [RFC2119] and indicate requirement levels for compliant CoAP.

2.2. Definitions

- Cloud DC: The data centers that are not on premises of enterprises yet have the compute/storage resources that can be requested or purchased by the enterprises. What the enterprises actually get is Virtual Data Centers.

- DC: Data Center

- Domain: The term Domain in this draft has different connotations in different scenarios:

  * Client--Provider relationship, i.e. client requesting some network functions from its provider;

  * Domain A - Domain B relationship, i.e. one operator domain requesting some network functions from another operator domain; or

  * Applications -- Network relationship, i.e. an application (e.g. cluster of servers) requesting some functions from network, etc.

- NSF - Network Security function

- I2NSF agent - a piece of software in a device that implements a network security function which receives security provisioning and filters across the I2NSF protocol in order provision and control the network security function.

- I2NSF client - A security client software that utilizes the I2NSF protocol to read, write or change the provisioning and filters in network security device via software interface using the I2NSF protocol (denoted as I2RS Agent)

- I2NSF SPDP - I2NSF client which serves as a collections and distribution point for security provisioning and filter data.

- I2NSF SEP - I2NSF agent which services as a insertion point for the security provisioning and filters in a NSF.

- Virtual Security Function: a security function that can be requested by one domain but may be owned or managed by another domain.

- Cloud-based security functions: NSF hosted and managed by service providers or different administrative entity.

3. Summary of Gap Analysis Points

On early focus on ACL policy enforcement on traffic entering a network is the 1990s COPS design (PEP and PDP) as shown in figure 1. The Policy decision point kept network-wide policy (E.g. ACLs) and sent it to Policy enforcements who then would control what data flows between the two These decision points controlled flow from PEP to PEP.

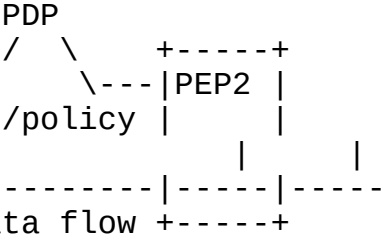

Figure 1

Today's security devices in 2015 replicate the same concept. The I2NSF Security provisioning policy/filter decision point (SPDP) and the I2NSF Agent Security Enforcement point (SEP) still look to control this flow through secure devices (see figure 2.).

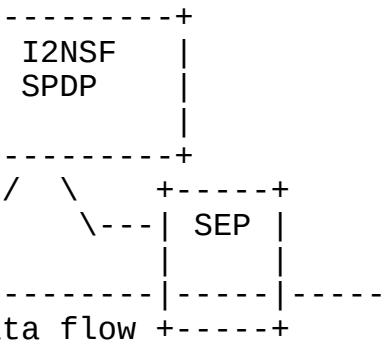

Figure 2

The other security protocols work to interact to create the additional pieces of security the flows for users as follows (see figure 3):

- SACM - examines if policy and reality match. It asks the questions "Have proper policies been pushed to the proper place", and "Has any policy been compromised?"

- MILE - looks at events that go Bump in night in the Security 2015. MILE examines when events need to be reported or correlated. A MILE configuration is a policy pushed out by the SPDP

- DOTS - picks up security the flow for when things go really wrong during security attacks. In this case, an SEP needs to be able to SCREAM for help, to get other SEP to ease its pain. DOTS policy is pushed out via SPDP.

- I2NSF may connect to all of these devices to gather information about the security policy that is pushes down to I2NSF agent. I2NSF provides a common interface between an I2NSF client as a SPDP and the NSF security boxes with SEP agents (which may also DOTS agent or Mile agent).

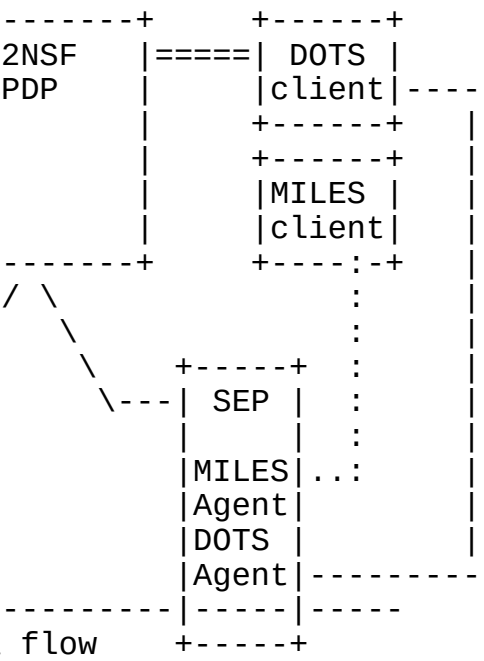

Figure 3

4. Analysis of NFV Status of the Arts in Industry

Network Function Virtualization (NFV) provides the service providers with flexibility, cost effective and agility to offer their services to customers. One such service is the network security function which guards the exterior of a service provider or its customers.

The flexibility and agility of NFV encourages service providers to provide different products to address business trends in their market to provide better service offerings to their end user. A traditional

product such as the network security function (NSF) may be broken into multiple virtual devices each hosted from another vendor. In the past, network security devices may have been single sourced from a small set of vendors - but in NFV version of NSF devices, this reduced set of sources will not provide a competitive edge. Due to this market shift, the network security device vendors are realizing that the proprietary provisioning protocols and formats of data may be a liability. Out of the NFV work has arisen a desire for a single interoperable network security device provisioning and control protocol.

The I2NSF will be deployed along networks using other security and NFV technology. As section 3 described, the NFV NSF security is deployed along side other security functions (AAA, DOTS, MILE, SCREAM devices) or deep-packet-inspection. The I2NSF will be deployed with routing functions that are configured by NETCONF/RESTCONF or I2RS which control the provisioning and management of the L1, L2, l3 and service pathways through the network.

In the NFV-related productions, the current architectures does not have a protocol to maintain an interoperability provisioning from I2NSF client to I2NSF agent. The result is that service providers have to manage the interoperability between private protocols. In response to this problem, the device manufacturers and the service providers have begun to discuss an I2NSF protocol for interoperable passing of provisioning and filter in formation.

Open source work (such as OPNFV) provides a common code base for providers to start their NFV work from. However, this code base faces the same problem. There is no defacto standard protocol.

5. Comparison of Current IETF Works

The following sections describes compares the current work in the IETF with the I2NSF. To provide an easier way of reviewing this work, the working groups in the IETF are addressed via Areas of work. The work of each working group (WG) is summarize and compared with I2NSF.

5.1. Network Management and Operations

5.1.1. Anima

Summary of Anima

ANIMA (Autonomic Networking Integrated Model and Approach) introduces a control paradigm where network processes, driven by objectives (or intent), coordinate their local decisions, autonomically translate

them into local actions, and adapt them automatically according to various sources of information including external information and protocol information bases.

ANIMA first step to is develop the platform that these autonomic network processes can run on.

ANIMA will develop protocols to achieve auto discovery among management system and devices. The listed drafts proposed include:

- The configuration discovery and negotiation protocol designed to be a generic platform, which is independent from the negotiation contents. There are also security aspects being discussed in the ANIMA drafts such as secure messages and keys which are passed among the discovered parties.

Diagram of Anima: (TBD)

Anima drafts

- Anima has no WG drafts

Why I2NSF is different than ANIMA

I2NSF is to develop application /user oriented policies (the attributes, the profiles, or the descriptors) of the network security functions that clients can request/query from 3rd party providers.

5.1.2. COPS

COPS had a design of Policy Enforcement Points (PEP), and policy Decision Points (PDP) as shown in figure 3. These decision points controlled flow from PEP to PEP.

Why COPS is no longer used

Security in the network in 2015 uses specific devices (IDS/IPS, NAT firewall, etc) with specific policies and profiles for each types of device. No common protocol or policy format exists between the policy manager (PDP) and security enforcement points. As described above, the security policy enforcement has security policy decision points (SPDP) and security enforcement points (SEP). Today’s security Policy Decision points exist where policy and services come together in a convenient place to push out SEP.

COPs RFCs: [RFC4261], [RFC2940], , [RFC3084], , [RFC3483]

Why I2NSF is different COPS

COPS was a protocol for all policy (security, flow, and others). I2NSF creates a common protocol between security policy decision points (SPDP) and security enforcement points (SEP). Today's security devices currently only proprietary protocols. Manufacturers wold like a security specific policy enforcement protocol rather than a generic policy protocol.

5.1.3. NETCONF/RESTCONF

Summary of IETF NETCONF WG

IETF NETCONF working group has developed the basics of the NETCONF protocol focusing on secure configuration and querying operational state. The NETCONF protocol [RFC6241] may be run over TLS [RFC6639] or SSH ([RFC6242]. NETCONF can be expanded to defaults [RFC6243], handling events ([RFC5277] and basic notification [RFC6470], nd filtering writes/reads based on network access control models (NACM, [RFC6536]). The NETCONF configuration must be committed to a configuration data store (denoted as config=TRUE). Yang models identify nodes within a configuration datastore or an operational data store using a XPath expression (document root ---to --- target source). NETCONF uses an RPC model and provides protocol for handling configs (get-config, edit-config, copy-config, delete-config, lock, unlock, get) and sessions (close-session, kill-session). The NETCONF Working Group has developed RESTCONF which is an HTTP-based protocol that provides a programmatic interface for accessing data defined in YANG, using the datastores defined in NETCONF.

RESTCONF supports "two edit condition detections" - time stamp and entity tag. RESTCONF uses a URI encoded path expressions. RESTCONF provides operations to get remote servers options (OPTIONS), retrieve data headers (HEAD), get data (GET), create resource/invoke operation (POST), patch data (PATCH), delete resource (DELETE), or query.

At this time, RESTCONF does not handle the ephemeral datastore proposed by I2RS (see Routing Area) at this time (see I2RS working group for details on I2RS). RESTCONF also does not promise to provide the real-time programmatic interface I2RS requires.

NETMOD developed initial Yang models for interfaces [RFC7223]), IP address ([RFC7277]), IPv6 Router advertisement ([RFC7277]), IP Systems ([RFC7317]) with system ID, system time management, DNS resolver, Radius client, SSH, syslog ([I-D.ietf-netmod-syslog-model]), ACLS ([I-D.ietf-netmod-acl-model]), and core routing blocks ([I-D.ietf-netmod-routing-cfg] The routing working group (rtgwg) has begun to examine policy for routing and tunnels.

Protocol specific Working groups have developed yang models for ISIS ([I-D.ietf-isis-yang-isis-cfg]), OSPF ([I-D.ietf-ospf-yang]), and BGP (merge of [I-D.shaikh-idr-bgp-model] and [I-D.zhdankin-idr-bgp-cfg] with the bgp policy proposed multiple Working groups (idr and rtgwg)). BGP Services yang models have been proposed for PPB EVPN ([I-D.tsingh-bess-pbb-evpn-yang-cfg]), EVPN ([I-D.zhuang-bess-evpn-yang]), L3VPN ([I-D.zhuang-bess-l3vpn-yang]), and multicast MPLS/BGP IP VPNs ([I-D.liu-bess-mvpn-yang]).

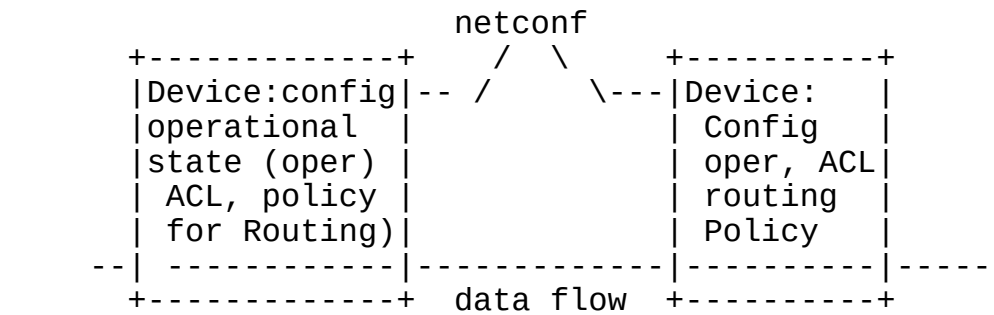

Figure 4

NETCONF and RESTCONF manage device layer yang models. However as figure 5 shows, there are multiple levels of device levels, network-wide level, and application level yang modules.

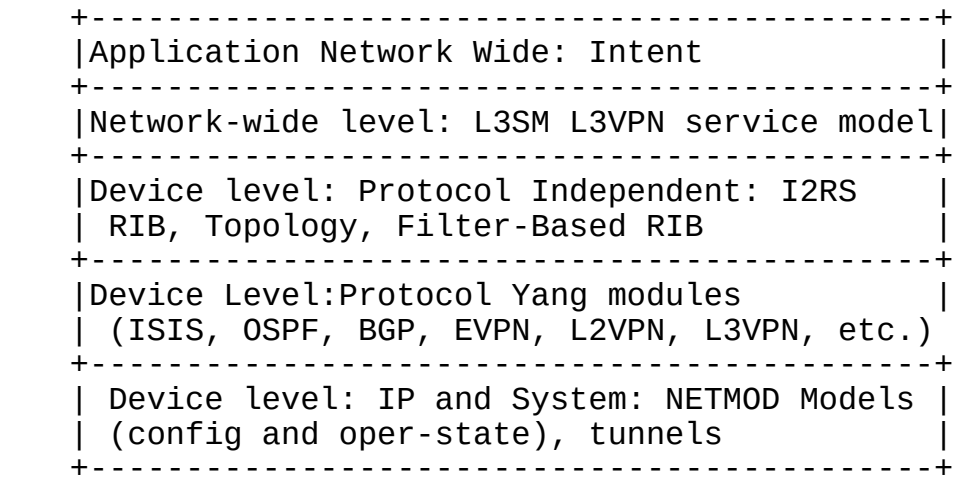

Figure 5 levels of Yang modules

RFCs for NETCONF

- NETCONF [RFC6242]
- NETCONF monitoring [RFC6022]

- NETCONF over SSH [RFC6242]
- NETCONF over TLS [RFC5539]
- NETCONF system notification> [RFC6470]
- NETCONF access-control (NACM) [RFC6536]
- RESTCONF [I-D.ietf-netconf-restconf]
- NETCONF-RESTCONF call home [I-D.ietf-netconf-call-home]
- RESTCONF collection protocol [I-D.ietf-netconf-restconf-collection]
- NETCONF Zero Touch Provisioning [I-D.ietf-netconf-zerotouch]

How I2NSF is different than NETCONF

NETCONF and RESTCONF are protocol for configuration of routing and IP devices, and monitoring of operational state. I2NSF seeks to create an interoperable protocol to pass security provisioning and filtesr.

What I2NSF can use from NETCONF

I2NSF should consider using NETCONF/RESTCONF protocol for capability layer to communicate the security data models to the designated security functions.

5.1.4. IETF L3SM

Beyond the device level yang models for network elements, protocol's configuration, operational status, or ephemeral state (I2RS), there is the goal of a full system configuration allows deployment of services across networks. Services are built from a combination of network element and protocol configuration, but are specified to service users in more abstract terms. The Layer Three Virtual Private Network Service Model (L3SM) working group is a short-lived WG tasked to create a YANG data model that describes a L3VPN service (a L3VPN service model) that can be used for communication between customers and network operators, and to provide input to automated control and configuration applications. This L3VPN service model is not an L3VPN configuration model. That is, it does not provide details for configuring network elements or protocols. Instead it contains the characteristics of the service, as discussed between the operators and their customers. A separate process is responsible for mapping this L3VPN service model (see figure 4) onto the protocols and network elements depending on how the network operator chooses to

realize the service. The starting point for this L3VPN model is [I-D.l3vpn-service-yang].

Status and Relevance

IETF L3SM working is an approved IETF working group with a draft written by authors who are operators at BT, Orange, Verizon, and ATT. This network-wide service model is at a network-wide level of service.

5.1.5. NEMO BOF

NeMo provides a simple transaction based Intent-based NBI, enabling applications to create, modify and takedown virtual networks built on virtual nodes with policy-controlled flows. The NeMo Intent NBI allows an application to communicate with a controller, providing the following group of commands:

- entity group: (un)node (un)link, (un)flow
- capabilities: (un)policy, query, notification, connect, disconnect, commit, and withdraw,
- model: Node Model, Link Model, and Link model.

An application exchanges NeMo commands, using the REST Protocol to a controller running a Nemo language processing engine, to instruct the controller to set up a virtual network of nodes and links with flow policy to control the data flows across the network links. NeMo uses an application’s view of the compute, storage, and network to allow an application to set any grouping of compute, storage, or network as a virtual node. This allows the application to decide what constitutes a compute node and what constitute a link and a flow. From the application’s viewpoint, it intends to connect two or more nodes in a network. It does not matter to the application if the node is a single virtual machine (VM) or a cluster of interconnected compute and storage devices with many network connections. NeMo’s NBI API hides this complexity, making the application’s commands prescriptive and simple. The

Nemo language engine in the controller is associated with a model that allows a group of applications to have a set of pre-loaded definitions (model semantics) for nodes, flows, or policy. For example, a company nodes could be defined along with the necessary flows for accounting traffic or big-data transfers.

NEMO Documents

- Intent Common Information Model (and definitions) [I-D.xia-ibnemo-icim],

- NEMO (NEtwork MOdeling Language) [I-D.xia-sdnrg-nemo-language],

- Yang Data Model for Intent-Based NEMO [I-D.zhou-netmod-intent-nemo]

- Requirements for Intent language(description, not title) [I-D.xia-sdnrg-service-description-language]

Relevance to I2NSF

The Intent-based or Declarative policy may be an aspect of the I2NSF customer requests. It is not directly related to the I2NSF Client to I2NSF Agent protocol passing provisioning work.

Status of Nemo

In 2014, the NEMO project provided an early proof-of-concept code demos (Layer123, CNV2015, IETF92) for an Intent-Based interface that uses a domain specific language. Nemo is moving this work into two open Source projects (ODL Nemo, OPNFV Movie) and work at IETF’s open-source projects.

5.1.6. SUPA BOF

The IETF SUPA (Simplified Use of Policy Abstractions) BOF is proposing an IETF Working Group to develop a set of information models for defining standardized policy rules at different levels of abstraction, and will show how to map these (technology-independent) forms into YANG data models. The BOF introduces the concepts of multi-level (multiple levels of abstraction) (similar to figure 5) and multi-technology (e.g., IP, VPNs, MPLS) network abstractions to address the current separation between development and deployment operations. Multiple levels of abstraction enable common concepts present in different technologies and implementations to be represented in a common manner. This facilitates using diverse components and technologies to implement a network service.

Three information models are envisioned:

- A generic information model that defines concepts needed by policy management independent of the form and content of the policy.

- A more specific information model that refines the generic information model to specify how to build policy rules of the event-condition-action paradigm.

- o A more specific information model that refines the generic information model to specify how to build policy rules that declaratively specify what goals to achieve (but not how to achieve those goals).

The set of generic policy information models in SUPA's work will be mapped to a set of concrete YANG data models. These data models will provide a set of core YANG modules that define how to manage and communicate policies, expressed using the event-condition-action paradigm or the declarative goal-oriented paradigm, between systems.

The SUPA BOF/WG plans to focus in the first phase of its work on completing the set of information models required to construct an extensible, policy-based framework. These information models will lead to a set of core YANG data models for a policy-based management framework to monitor and control network services.

The working group will use the distributed data center (DDC) use case, which includes the dynamic policy-driven provisioning and operation of inter-datacenter (inter-dc) virtual private networks (VPNs) of various types, as a means to validate that the generic policy-framework is implementable and usable.

I2NSF versus SUPA BOF work

I2NSF is focus on passing policies between I2NSF client and I2NSF Agent in an interoperable format. The SUPA policies are more generic policies (Prescriptive Event-Condition-Action and declarative/Intent-based. The protocol between the I2NSF Client and I2NSF agent is specific to the security policies. If SUPA was completed now, it might provide wisdom for the I2NSF interoperable protocol. With SUPA running in parallel, the generic models may or may not provide timely advise to structure I2NSF protocol.

5.2. Internet

5.2.1. PCP

As indicated by the name, the Port Control Protocol (PCP) enables an IPv4 or IPv6 host to flexibly manage the IP address and port mapping information on Network Address Translators (NATs) or firewalls, to facilitate communication with remote hosts.

PCP RFCs:

[RFC6887]

[RFC7225]

[I-D.ietf-pcp-authentication]

[I-D.ietf-pcp-optimize-keepalives]

[I-D.ietf-pcp-proxy]

Why is I2NSF different from PCP:

Here are some aspects that I2NSF is different from PCP:

- PCP only support the management of port and address information rather than any other security functions
- We must cover the proxy, firewall and NAT box proposals in I2NSF

5.2.2. Midcom

Midcom Summary:

summary TBD

MidCom RFCs:

RFCs

Why I2NSF is different than Midcom

TBD

explanation of differences

5.3. Routing

5.3.1. I2RS

Summary of I2RS

The IETF I2RS Working group is working on an interface to the routing system that facilitates a real-time or event driven interaction with the routing system through a collection of protocol-based control or management interfaces. These allow information, policies, and operational parameters to be injected into and retrieved (as read or by notification) from the routing system while retaining data consistency and coherency across the routers and routing infrastructure, and among multiple interactions with the routing system. The I2RS interfaces co-exist with existing configuration and management systems and interface that focus on configuring, managing, or monitoring information on the routing system in a device.

A short description of the problem that I2RS is trying to solve can be found in [I-D.ietf-i2rs-problem-statement] It is envisioned that users of the I2RS interfaces will be management applications, network controllers, and user applications that make specific demands on the network. The use case requirements are described in [I-D.ietf-i2rs-usecase-reqs-summary] for protocol independent RIBs, topologies, and filter-rules and for protocol dependent use cases for BGP, OSPF, ISIS, CCNE, SFC, traffic steering, MPLS-TE, MPLS-LDP, Mobile Backhaul(MBB) uses, large data flows, large data collection systems, and CDNI. The I2RS Architecture [I-D.ietf-i2rs-architecture] states the I2RS will be data-model driven.

I2RS has three protocol independent models:

- I2RS RIB [I-D.ietf-i2rs-rib-data-model] ([I-D.ietf-i2rs-rib-info-model],
- I2RS Topology models (generic, L1, L2, L3, and service topology)
  - Generic topolgoy [I-D.ietf-i2rs-yang-network-topo]
  - L1 topology [I-D.zhang-i2rs-l1-topo-yang-model],
  - L2 Topology [I-D.ietf-i2rs-yang-l2-network-topology]",
  - L3 Topology (draft-ietf-i2rs-yang-l3-topo-00"), and
  - service topology model [I-D.hares-i2rs-info-model-service-topo].
- Filter-Based RIB topology [I-D.kini-i2rs-fb-rib-info-model].

The I2RS WG has a policy of re-use of existing technology where possible. One of the potential re-uses is the enhancement of the NETCONF protocol [RFC6241], or RESTCONF [I-D.ietf-netconf-restconf], and the use of the netmod (RFC6020) for the data models. In June 2015, I2RS is finalizing the requirements for changes in the netconf protocol. Existing requirements include:

- requirements for I2RS’s ephemeral state [I-D.haas-i2rs-ephemeral-state-reqs] that provides writing/reading of real-time state,
- requirements for traceability framework and information model described in [I-D.ietf-i2rs-traceability],

- requirements for subscriptions to datastores [I-D.ietf-i2rs-pub-sub-requirements], and

- mutual authentication requirements and transport requirements (draft pending).

I2RS modules have been proposed for ephemeral state for protocol dependent units for OSPF, ISIS, BGP, MPLS-TE, MPLS-LDP, SFC forwarding, and SFC filter-based rules.

Pre-standard implementations of I2RS protocol exist in Juniper and other vendors.

Why I2NSF is different than I2NSF

I2NSF focus is on an interoperable protocol that passes policy between the I2NSF client and the I2NSF AGent. The I2RS client passes ephemeral state for configuration and operational state for protocol and protocol-independent yang modules. A part of this state may be the routing policy that applies to a routing agent. The specific policies for a network security devices are not consider in I2RS at this time.

What I2NSF can use from I2RS

I2NSF may want to use I2RS ephemeral state (configuration and operational) as it manages, monitors, or handles NSF devices. The I2NSF may want to re-use I2RS protocol or modules to pass this ephemeral state.

I2RS Status

Status and Relevance IETF I2RS is nearing the end of its initial definition cycle for protocol independent yang models and its protocol requirements for NETCONF Working Group. If protocol additions to netconf’s protocol and netmod’s yang modules for the I2RS ephemeral state can be finalized in June, then early implementation of the I2RS code may appear in the summer with the IETF hack-a-thon. Movement of I2RS code is possible into ODL, Cisco, Juniper, Ericsson, Huawei, Brocade, Dell and PacketDesign as authors from these companies have joined together to create the I2RS drafts. An I2RS interface into all routers will provide a programmatic interface for many routing stacks.

5.3.2. SFC

Summary of SFC:

IETF SFC is about mechanism of chaining together service functions; IETF SFC treats all those Service Functions as black box. This means that the SFC mechanism do not care what actions those functions are performing. SFC defines the SFC header to carry Metadata with payload to those functions. But SFC mechanism do not specify what content is encoded in the metadata.

diagram of SFC: TBD

SFC RFCs (TBD)

Why I2NSF is different:

I2NSF is targeted to define the descriptor (the actual rules and policies) of the network security functions needed and the negotiation scheme.

5.4. Transport Area

5.4.1. NSIS - Next steps in Signalling

NSIS is for standardizing an IP signaling protocol (RSVP) along data path for end points to request its unique QoS characteristics, unique FW policies or NAT needs (RFC5973) that are different from the FW/NAT original setting. The requests are communicated directly to the FW/ NAT devices. NSIS is like east-west protocols that require all involved devices to fully comply to make it work.

NSIS is path-coupled, it is possible to message every participating device along a path without having to know its location, or its location relative to other devices (this is particularly a pressing issue when you’ve got one or more NATs present in the network, or when trying to locate appropriate tunnel endpoints).

A diagram should be added here showing I2NSF and NSIS

Why I2NSF is different than NSIS:

- The I2NSF requests form clients do not go directly to network security devices, but instead to controller or orchestrator that can translate the application/user oriented policies to the involved devices in the interface that they support.

- The I2NSF request does not require all network functions in a path to comply, but it is a protocol between the I2NSF client and the I2NSF Agent in the controller and orchestrator

- I2NSF defines clients (applications) oriented descriptors (profiles, or attributes) to request/negotiate/validate the network security functions that are not on the local premises.

Why we belief I2NSF has a higher chance to be deployed than NSIS:

- Open Stack already has a proof-of-concept/preliminary implementation, but the specification is not complete. IETF can play an active role to make the specification for I2NSF complete. IETF can complete and extend the OpenStack implementation to provide an interoperable specification that can be needs and requirements of operators that is workable for suppliers of the technology. The combination of an carefully designed interoperable IETF specification with an open-source code development Open Stack will leverage the strengths of the two communities, and expand the informal ties between the two groups. A software development cycle has the following components: architecture, design specification, coding, and interoperability testing. The IETF can take ownership of the first two steps, and provide expertise and a good working atmosphere (in hack-a-thons) in the last two steps for OpenSTack or other open-source coders.

- IETF has the expertise in security architecture and design for interoperable protocols that span controllers/routers, middle-boxes, and security end-systems.

- IETF has a history of working on interoperable protocols or virtualized network functions (L2VPN, L3VPN) that are deployed by operators in large scale devices. IETF has a strong momentum to create virtualized network functions (see SFC WG in routing) to be deployed in network boxes. [Note: We need to add SACM and others here].

5.4.2. VNFPool BOF

VNFpool is about the reliability and availability of the virtualized network functions. But none of them address how service functions are requested, or how service functions are fulfilled.

drawing for VNF-Pool

RFCs for VNF-Pool

Why I2NSF is different than the VNFPool BOF Proposal

VNFpool does not cover the protocol for provisioning a NSF (e.g. rules for the requested FW) from the I2NSF clients to I2NSF Agent. VNFPool examined a way to provide an interoperable protocol manage the VNF pools from different vendors. With VNFpool (as well as SFC), NSF functions (such as Firewall function) are treated as a black box, that is treated in same way as Video Optimization function.

6. IANA Considerations

No IANA exist for this document.

7. Security Considerations

No security considerations are involved with a gap analysis.

8. References

8.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2. Informative References

[I-D.haas-i2rs-ephemeral-state-reqs]
Haas, J., "I2RS Ephemeral State Requirements", draft-haas-i2rs-ephemeral-state-reqs-00 (work in progress), May 2015.

[I-D.hares-i2rs-info-model-service-topo]
Hares, S., Wu, W., Wang, Z., and J. You, "An Information model for service topology", draft-hares-i2rs-info-model-service-topo-03 (work in progress), January 2015.

[I-D.ietf-i2rs-architecture]
Atlas, A., Halpern, J., Hares, S., Ward, D., and T. Nadeau, "An Architecture for the Interface to the Routing System", draft-ietf-i2rs-architecture-09 (work in progress), March 2015.

[I-D.ietf-i2rs-problem-statement]
Atlas, A., Nadeau, T., and D. Ward, "Interface to the Routing System Problem Statement", draft-ietf-i2rs-problem-statement-06 (work in progress), January 2015.

[I-D.ietf-i2rs-pub-sub-requirements]
Voit, E., Clemm, A., and A. Prieto, "Requirements for Subscription to YANG Datastores", draft-ietf-i2rs-pub-sub-requirements-02 (work in progress), March 2015.

[I-D.ietf-i2rs-rib-data-model]
Wang, L., Ananthakrishnan, H., Chen, M., amit.dass@ericsson.com, a., Kini, S., and N. Bahadur, "A YANG Data Model for Routing Information Base (RIB)", draft-ietf-i2rs-rib-data-model-00 (work in progress), April 2015.

[I-D.ietf-i2rs-rib-info-model]
Bahadur, N., Folkes, R., Kini, S., and J. Medved, "Routing Information Base Info Model", draft-ietf-i2rs-rib-info-model-06 (work in progress), March 2015.

[I-D.ietf-i2rs-traceability]
Clarke, J., Salgueiro, G., and C. Pignataro, "Interface to the Routing System (I2RS) Traceability: Framework and Information Model", draft-ietf-i2rs-traceability-03 (work in progress), May 2015.

[I-D.ietf-i2rs-usecase-reqs-summary]
Hares, S. and M. Chen, "Summary of I2RS Use Case Requirements", draft-ietf-i2rs-usecase-reqs-summary-01 (work in progress), May 2015.

[I-D.ietf-i2rs-yang-l2-network-topology]
Dong, J. and X. Wei, "A YANG Data Model for Layer-2 Network Topologies", draft-ietf-i2rs-yang-l2-network-topology-00 (work in progress), April 2015.

[I-D.ietf-i2rs-yang-network-topo]
Clemm, A., Medved, J., Varga, R., Tkacik, T., Bahadur, N., and H. Ananthakrishnan, "A Data Model for Network Topologies", draft-ietf-i2rs-yang-network-topo-00 (work in progress), April 2015.

[I-D.ietf-isis-yang-isis-cfg]
Litkowski, S., Yeung, D., Lindem, A., Zhang, J., and L. Lhotka, "YANG Data Model for ISIS protocol", draft-ietf-isis-yang-isis-cfg-02 (work in progress), March 2015.

[I-D.ietf-netconf-call-home]
Watsen, K., "NETCONF Call Home and RESTCONF Call Home", draft-ietf-netconf-call-home-06 (work in progress), May 2015.

[I-D.ietf-netconf-restconf]
Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", draft-ietf-netconf-restconf-04 (work in progress), January 2015.

[I-D.ietf-netconf-restconf-collection]
Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Collection Resource", draft-ietf-netconf-restconf-collection-00 (work in progress), January 2015.

[I-D.ietf-netconf-zerotouch]
Watsen, K., Clarke, J., and M. Abrahamsson, "Zero Touch Provisioning for NETCONF Call Home (ZeroTouch)", draft-ietf-netconf-zerotouch-02 (work in progress), March 2015.

[I-D.ietf-netmod-acl-model]
Bogdanovic, D., Sreenivasa, K., Huang, L., and D. Blair, "Network Access Control List (ACL) YANG Data Model", draft-ietf-netmod-acl-model-02 (work in progress), March 2015.

[I-D.ietf-netmod-routing-cfg]
Lhotka, L. and A. Lindem, "A YANG Data Model for Routing Management", draft-ietf-netmod-routing-cfg-19 (work in progress), May 2015.

[I-D.ietf-netmod-syslog-model]
Wildes, C. and K. Sreenivasa, "SYSLOG YANG model", draft-ietf-netmod-syslog-model-03 (work in progress), March 2015.

[I-D.ietf-ospf-yang]
Yeung, D., Qu, Y., Zhang, J., Bogdanovic, D., and K. Sreenivasa, "Yang Data Model for OSPF Protocol", draft-ietf-ospf-yang-00 (work in progress), March 2015.

[I-D.ietf-pcp-authentication]
Wasserman, M., Hartman, S., Zhang, D., and T. Reddy, "Port Control Protocol (PCP) Authentication Mechanism", draft-ietf-pcp-authentication-09 (work in progress), May 2015.

[I-D.ietf-pcp-optimize-keepalives]
Reddy, T., Patil, P., Isomaki, M., and D. Wing, "Optimizing NAT and Firewall Keepalives Using Port Control Protocol (PCP)", draft-ietf-pcp-optimize-keepalives-06 (work in progress), May 2015.

[I-D.ietf-pcp-proxy]
Perreault, S., Boucadair, M., Penno, R., Wing, D., and S. Cheshire, "Port Control Protocol (PCP) Proxy Function", draft-ietf-pcp-proxy-08 (work in progress), May 2015.

[I-D.kini-i2rs-fb-rib-info-model]
Kini, S., Hares, S., Ghanwani, A., Krishnan, R., Wu, Q., Bogdanovic, D., Tantsura, J., and R. White, "Filter-Based RIB Information Model", draft-kini-i2rs-fb-rib-info-model-00 (work in progress), March 2015.

[I-D.l3vpn-service-yang]
Litkowski, S., Shakir, R., Tomotaki, L., and K. D'Souza, "YANG Data Model for L3VPN service delivery", draft-l3vpn-service-yang-00 (work in progress), February 2015.

[I-D.liu-bess-mvpn-yang]
Liu, Y. and F. Guo, "Yang Data Model for Multicast in MPLS/BGP IP VPNs", draft-liu-bess-mvpn-yang-00 (work in progress), April 2015.

[I-D.shaikh-idr-bgp-model]
Shaikh, A., D'Souza, K., Bansal, D., and R. Shakir, "BGP Model for Service Provider Networks", draft-shaikh-idr-bgp-model-01 (work in progress), March 2015.

[I-D.tsingh-bess-pbb-evpn-yang-cfg]
Tiruveedhula, K., Singh, T., Sajassi, A., Kumar, D., and L. Jalil, "YANG Data Model for PBB EVPN protocol", draft-tsingh-bess-pbb-evpn-yang-cfg-00 (work in progress), March 2015.

[I-D.xia-ibnemo-icim]
Xia, Y., Zhou, T., Zhang, Y., Hares, S., Aranda, P., Lopez, D., Crowcroft, J., and Y. Zhang, "Intent Common Information Model", draft-xia-ibnemo-icim-00 (work in progress), May 2015.

[I-D.xia-sdnrg-nemo-language]
Xia, Y., Jiang, S., Zhou, T., and S. Hares, "NEMO (NEtwork MOdeling) Language", draft-xia-sdnrg-nemo-language-02 (work in progress), May 2015.

[I-D.xia-sdnrg-service-description-language]
Xia, Y., Jiang, S., and S. Hares, "Requirements for a Service Description Language and Design Considerations", draft-xia-sdnrg-service-description-language-02 (work in progress), May 2015.

[I-D.zhang-i2rs-l1-topo-yang-model]
Zhang, X., Rao, B., and X. Liu, "A YANG Data Model for Layer 1 Network Topology", draft-zhang-i2rs-l1-topo-yang-model-01 (work in progress), March 2015.

[I-D.zhdankin-idr-bgp-cfg]
Alex, A., Patel, K., Clemm, A., Hares, S., Jethanandani, M., and X. Liu, "Yang Data Model for BGP Protocol", draft-zhdankin-idr-bgp-cfg-00 (work in progress), January 2015.

[I-D.zhou-netmod-intent-nemo]
Zhou, T., Liu, S., Xia, Y., and S. Jiang, "YANG Data Models for Intent-based NEtwork MOdel", draft-zhou-netmod-intent-nemo-00 (work in progress), February 2015.

[I-D.zhuang-bess-evpn-yang]
Zhuang, S. and Z. Li, "Yang Model for Ethernet VPN", draft-zhuang-bess-evpn-yang-00 (work in progress), December 2014.

[I-D.zhuang-bess-l3vpn-yang]
Zhuang, S. and Z. Li, "Yang Data Model for BGP/MPLS IP VPNs", draft-zhuang-bess-l3vpn-yang-00 (work in progress), December 2014.

[RFC2940] Smith, A., Partain, D., and J. Seligson, "Definitions of Managed Objects for Common Open Policy Service (COPS) Protocol Clients", RFC 2940, October 2000.

[RFC3084] Chan, K., Seligson, J., Durham, D., Gai, S., McCloghrie, K., Herzog, S., Reichmeyer, F., Yavatkar, R., and A. Smith, "COPS Usage for Policy Provisioning (COPS-PR)", RFC 3084, March 2001.

[RFC3483] Rawlins, D., Kulkarni, A., Bokaemper, M., and K. Chan, "Framework for Policy Usage Feedback for Common Open Policy Service with Policy Provisioning (COPS-PR)", RFC 3483, March 2003.

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

[RFC4261] Walker, J. and A. Kulkarni, "Common Open Policy Service (COPS) Over Transport Layer Security (TLS)", RFC 4261, December 2005.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.

[RFC5277] Chisholm, S. and H. Trevino, "NETCONF Event Notifications", RFC 5277, July 2008.

[RFC5539] Badra, M., "NETCONF over Transport Layer Security (TLS)", RFC 5539, May 2009.

[RFC6022] Scott, M. and M. Bjorklund, "YANG Module for NETCONF Monitoring", RFC 6022, October 2010.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, June 2011.

[RFC6243] Bierman, A. and B. Lengyel, "With-defaults Capability for NETCONF", RFC 6243, June 2011.

[RFC6436] Amante, S., Carpenter, B., and S. Jiang, "Rationale for Update to the IPv6 Flow Label Specification", RFC 6436, November 2011.

[RFC6470] Bierman, A., "Network Configuration Protocol (NETCONF) Base Notifications", RFC 6470, February 2012.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, March 2012.

[RFC6639] King, D. and M. Venkatesan, "Multiprotocol Label Switching Transport Profile (MPLS-TP) MIB-Based Management Overview", RFC 6639, June 2012.

[RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

[RFC7223] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 7223, May 2014.

[RFC7225] Boucadair, M., "Discovering NAT64 IPv6 Prefixes Using the Port Control Protocol (PCP)", RFC 7225, May 2014.

[RFC7277] Bjorklund, M., "A YANG Data Model for IP Management", RFC 7277, June 2014.

[RFC7317] Bierman, A. and M. Bjorklund, "A YANG Data Model for System Management", RFC 7317, August 2014.

Authors' Addresses

Susan Hares
Huawei
7453 Hickory Hill
Saline, MI 48176
USA

Email: shares@ndzh.com

Bob Moskowitz
HTT Consulting
Oak Park, MI 48237

Email: rgm@labs.htt-consult.com

Hosneih Rozanak
Munich
Germany

Email: ietf@rozanak.com

Dacheng Zhang
Beijing
China

Email: dacheng.zdc@aliabab-inc.com