I2NSF
Internet-Draft
Intended status: Standards Track
Expires: June 22, 2016

S. Hares
L. Dunbar
Huawei
A. Pastor
D. Lopez
Telefonica I+D
M. Zarny
Goldman Sachs
N. Leymann
Deutsche Telekom
M. Georgiades
Prime Tel
M. Qi
China Mobile
M. Boucadair
C. Jacquenet
France Telecom
S. Chakrabarty
US Ignite
December 20, 2015

I2NSF Problem Statement and Use cases
draft-hares-i2nsf-merged-problem-use-cases-00.txt

Abstract

This document describes the problem statement for Interface to Network Security Functions (I2NSF) and summary of the I2NSF use cases.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document describes the problem statement for Interface to Network Security Functions (I2NSF) and summary of the I2NSF use cases. A summary of the I2NSF state of the art in the industries and IETF which is relevant to I2NSF work is contained in [I-D.hares-i2nsf-gap-analysis].

The growing challenges and complexity in maintaining a secure infrastructure, complying with regulatory requirements, and controlling costs are enticing enterprises into consuming network security functions hosted by service providers. The hosted security service is especially attractive to small and medium size enterprises who suffer from a lack of security experts to continuously monitor, acquire new skills and propose immediate mitigations to ever increasing sets of security attacks.

According to [Gartner-2013], the demand for hosted (or cloud-based) security services is growing. Small and medium-sized businesses (SMBs) are increasingly adopting cloud-based security services to replace on-premises security tools, while larger enterprises are deploying a mix of traditional and cloud-based security services.

To meet the demand, more and more service providers are providing hosted security solutions to deliver cost-effective managed security services to enterprise customers. The hosted security services are primarily targeted at enterprises (especially small/medium ones), but could also be provided to any kind of mass-market customer. As the result, the Network security functions (NSFs) are provided and consumed in increasingly diverse environments. Users of NSFs may consume network security services hosted by one or more providers, which may be their own enterprise, service providers, or a combination of both. This document also briefly describes the following use cases summarized by [I-D.pastor-i2nsf-merged-use-cases]:

- o [I-D.pastor-i2nsf-access-usecases] (I2NSF-Access),
- o [I-D.zarny-i2nsf-data-center-use-cases](I2NSF-DC), and
- o [I-D.qi-i2nsf-access-network-usecase] (I2NSF-Mobile).

2. Terminology

ACL: Access Control List

B2B: Business-to-Business

Bespoke: Something made to fit a particular person, client or company.

Bespoke security management: Security management which is make to fit a particular customer.

DC: Data Center

FW: Firewall

IDS: Intrusion Detection System

IPS: Intrusion Protection System

NSF: Network security function. An NSF is a function that that detects unwanted activity and blocks/mitigates the effect of such unwanted activity in order to support availability of a network. In addition, the NSF can help in supporting communication stream integrity and confidentiality.

Flow-based NSF: A NSF which inspects network flows according to a policy intended for enforcing security properties. Flow based security also means that packets are inspected in the order they are received, and without modification to the packet due to the inspection process (MAC rewrites, TTL decrement action, or NAT inspection or changes).

Virtual NSF: A NSF which is deployed as a distributed virtual device.

VNFPool: Pool of Virtual Network Functions.

3. Problem Space

The following sub-section describe the problems and challenges facing customers and security service providers when some or all of the security functions are no longer physical hosted by the customer's administrative domain.

Security service providers can be internal to the company or external security service providers. For example, an internal IT Security group within a large enterprise could act as a security service provider for the enterprise. In contrast, an enterprise could outsource all security services to an external security service provider in a global service provider. In document, the security service provider function whether it is internal or external, will be denoted as "service provider".

The "Customer-Provider" relationship may be between any two parties. The parties can be in different firms or different domains of the same firm. Contractual agreements may be required in such contexts to formally document the customer's security requirements and the provider's guarantees to fulfill those requirements. Such agreements may detail protection levels, escalation procedure, alarms reporting, etc. There is currently no standard mechanism to capture those requirements.

A service provider may be a customer of another service provider.

3.1. Facing Security Service Providers

3.1.1. Diverse Types of Security Functions

There are many types of NSFs. NSFs by different vendors can have different features and have different interfaces. NSFs can be deployed in multiple locations in a given network, and perhaps have different roles.

Below are a few examples of security functions and locations or contexts in which they are often deployed:

External Intrusion and Attack Protection: Examples of this function are firewall/ACL authentication, IPS, IDS, and endpoint protection.

Security Functions in a DMZ: Examples of this function are firewall/ACLs, IDS/IPS, authentication and authorization services, NAT, forwarding proxies, application, and AAA services. These functions may be physically on-premise in a server provider's network at the DMZ spots or at "virtual" DMZ.

Internal Security Analysis and Reporting: Examples of this function are security logs, event correlation, and forensic analysis.

Internal Data and Content Protection: Examples of this function are encryption, authorization, and public/private key management for internal database.

Given the diversity of security functions, the contexts in which these functions can be deployed, and the constant evolution of these functions, standardizing all aspects of security functions is challenging, and most probably not feasible. Fortunately, it is not necessary to standardize all aspects. For example, from an I2NSF perspective, there is no need to standardize on how a firewall filters are created or applied.

What is needed is having a standardized interface to control and monitor the rule sets that NSFs use to treat packets traversing through. And standardizing interfaces will provide an impetuous for standardizing established security functions.

3.1.2. Diverse Interfaces to Control NSFS

To provide effective and competitive solutions and services, Security Service Providers may need to utilize multiple security functions from various vendors to enforce the security policies desired by their customers.

Since no widely accepted industry standard security interfaces exists today, management of NSFs (device and policy provisioning, monitoring, etc.) tends to be bespoke security management offered by product vendors. As a result, automation of such services, if it exists at all, is also bespoke. Thus, even in the traditional way of deploying security features, there is a gap to coordinate among implementations from distinct vendors. This is the main reason why mono-vendor security functions are often deployed and enable in a particular network segment.

A challenge for monitoring is that an NSF cannot monitor what it cannot view. Therefore, enabling a security function (e.g., firewall [I-D.ietf-opsawg-firewalls]) does not mean that a network is protected. As such, it is necessary to have a mechanism to monitor and provide execution status of NSFs to security and compliance management tools. There exist various network security monitoring vendor specific interfaces for forensics and troubleshooting.

3.1.3. Diverse Interface to monitor the behavior of NSFs

Obviously, enabling a security function (e.g., firewall [I-D.ietf-opsawg-firewalls] does not mean that a network is protected. Therefore, it is necessary to have a mechanism to monitor the execution status of NSFs.

3.1.4. More Distributed NSFs and vNSFs

The security functions which are invoked to enforce a security policy can be located in different equipment and network locations.

The European Telecommunications Standards Institute (ETSI) Network Function Virtualization (NFV) initiative creates new management challenges for security policies to be enforced by distributed, virtual, and network security functions (vNSF).

A vNSF has higher risk of failure, migrating, and state changes as their hosting VMs being created, moved, or decommissioned.

3.1.5. More Demand to Control NSFs Dynamically

In the advent of SDN (see [I-D.jeong-i2nsf-sdn-security-services]), more clients, applications or application controllers need to dynamically update their communication policies that are enforced by NSFs. The Security Service Providers have to dynamically update control requests to NSFs upon receiving the requests from their clients

3.1.6. Demand for multi-tenancy to control and monitor NSFs

Service providers may require having several operational units to control and monitor the NSFs, especially when NSFs become distributed and virtualized.

3.1.7. Lack of Characterization of NSFs and Capability Exchange

To offer effective security services, service providers need to activate various security functions in NSFs or vNSFs manufactured by multiple vendors. Even within one product category (e.g., firewall), security functions provided by different vendors can have different features and capabilities. For example, filters that can be designed and activated by a firewall may or may not support IPv6 depending on the firewall technology.

The service provider’s management system (or controller) needs a way to retrieve the capabilities of service functions by different vendors so that it could build an effective security solution. These service function capabilities can be documented in a static manner (e.g. a file) or via an interface which access a repository of security function capabilities which the NSF vendors dynamically update.

A dynamic capability registration is useful for automation because security functions may be subject to software and hardware updates. These updates may have implications on the policies enforced by the NSFs.

Today, there is no standard method for vendors to describe the capabilities of their security functions. Without a common technical framework to describe the capabilities of security functions, service providers cannot automate the process of selecting NSFs by different vendors to accommodate customer's requirements.

3.1.8. Lack of Mechanism for NSFs to utilize external profiles

Many security functions depend on signature files or profiles to perform (e.g. IPS/IDS signatures, DOTS filters). Different policies might need different signatures or profiles. Today, the construction and use of black databases can be win-win strategy for all parties involved. There might be Open Source provided signature/profiles (e.g. by Snort or others) in the future.

There is a need to have a standard envelop (i.e. the format) to allow NSFs to use external profiles.

3.1.9. Lack of Mechanisms to accept external alerts to trigger automatic configuration changes

NSF can ask the I2NSF security controller to alter network policy. For example, a DDoS alert could trigger a change to routing system to send traffic to a traffic scrubbing service to mitigate the DDoS.

The DDoS protection has the following two parts: a) the configuration of signaling of open threats and b) DDoS mitigation. DOTS controller manages the signaling part of DDoS. I2NSF controller(s) would manage the changing to the network policy. By monitoring the network alerts from DDoS, I2NSF can feed a alerts analytics engine that could recognize attacks and the I2NSF can implement the needed new policies.

DDoS mitigation is enhanced if the provider's network security controller can monitor, analyze, and investigate the abnormal events and provide information to the client or change the network configuration (see section x) for details on the interfaces.

[I-D.zhou-i2nsf-capability-interface-monitoring] provides details on how monitoring aspects of the flow-based Network Security Functions (NSFs) can use the I2NSF interfaces to receive traffic reports and enforce policy.

3.1.10. Lack of mechanism for dynamic key distribution to NSFs

There is a need for controller to distribute various keys to distributed NSFs. To distribute various keys, the keys must be created and managed. While there is many key management methods and key derivation functions (KDF), there is a lack of standard interface to provision and manage keys.

The keys may be used for message authentication and integrity in order to protect data flow. In addition, keys may be used to secure the protocol and messages in the core routing infrastructure.

As of now there is no much focus on an abstraction for keying information that describes the interface between protocols, operators, and automated key management.

The keys may be used for message authentication and integrity in order to protect data flow. In addition, keys may be used to secure the protocol and messages in the core routing infrastructure.

The ability to utilize keys when routing protocols send or receive messages will be enhanced by having an abstract key table maintained by a security services. Conceptually, there must be an interface defined for routing/signaling protocols to make requests of automated key management when it is being used, to notify the protocols when keys become available in the key table.

An abstract key service needs to have three things:

1. I2NSF need to design the key table abstraction, the interface between key management protocols and routing/other protocols, and possibly security protocols at other layers.

2. For each routing/other protocol, I2NSF need to define the mapping between how the protocol represents key material and the protocol-independent key table abstraction. (If protocols share common mechanisms for authentication (e.g. TCP Authentication Option), then the same mapping may be reused.)

3. Automated Key management must support both symmetric keys and group keys via the service provided by items 1 and 2.

3.1.10.1. Background on Core Routing Security

A recommendation from a workshop held by the Internet Architecture Board (IAB) held a workshop on the topic of "Unwanted Internet Traffic" [RFC4948] suggest since a "simple risk analysis" suggests an "ideal attack target of minimal cost but maximal disruption is the core routing infrastructure", it is important to "tightening the security of the core routing infrastructure". One of the ways to tighten security of the core routing infrastructure is to tighten the security of protocol packets on the wire is by protecting the messages by use of keys.

Conceptually, when routing protocols send or receive messages, they might need to look up the key to use in this abstract key table. Conceptually, there must be an interface defined for a protocols to make requests of automated key management when it is being used; when keys become available, they might be made available in the key table.

3.2. Challenges Facing Customers

When customers invoke hosted security services, their security policies may be enforced by a collection of security functions hosted in different domains. Customers may not have the security skills to express sufficiently precise requirements or security policies. Usually these customers express the expectations of their security requirements or the intent of their security policies. These expectations can be considered customer level security expectations. Customers may also desire to express guidelines for security management. Examples of such guidelines are the following:

- o Which critical communications are to be preserved during critical events (DOTS),
- o Which hosts are to continue service even during severe security attacks (DOTS),
- o Reporting of attacks to CERT (MILE),
- o Managing network connectivity of systems out of compliance (SACM),

3.2.1. NSFs from Heterogeneous Administrative Domains

Many medium and large enterprises have deployed various on-premises security functions which they want to continue to deploy. These enterprises want to combine local security functions with remote hosted security functions to achieve more efficient and immediate counter-measures to both Internet-originated attacks and enterprise network-originated attacks.

Some enterprises may only need the hosted security services for their remote branch offices where minimal security infrastructures/ capabilities exist. The security solution will consist of NSFs on customer networks and NSFs on service provider networks.

3.2.2. Today's Control Requests are Vendor Specific

Customers may consume NSFs by multiple service providers. Customers need to express their security requirements, guidelines, and expectations to the service providers. In turn, the service providers must translate this customer information into customer security policies and associated configuration sets for the set of security functions in their network. Without a standard technical characterizations or a standard interface, the service provide faces many challenges.

Due the lack of standard technical characterizations and a standard interfaces, the following problems exists:

No standard technical characterization and/or APIs : Even the most common security services there is no standard technical characterization or APIs. Most security services are accessible only through disparate, proprietary interfaces (e.g., portals or APIs) in whatever format vendors choose to offer. The service provider must the customer's input to these widely varying interfaces.

No standard interface: Without standard interfaces it is complex for customers to update security policies or integrate the security functions in their enterprise with the security services provided by the security service providers. This complexity is induced by the diversity of the configuration models, policy models, and supported management interfaces. Without a standard interface, new innovative security products find a large barrier to entry into the market

Managing by scripts de-jour: The current practices rely on the use of scripts which generate other scripts which the automatically run to upload or download configuration changes, log information and other things. These scripts have to be adjusted each time an implementation from a different vendor is enabled in a provider side.

Lack of immediate Feedback: Customers may also require a mechanism to easily update/modify their security requirements with immediate effect in the underlying involved NSFs.

Lack of explicit invocation request: While security agreements are in place, security functions may be solicited without requiring an explicit invocation means. Nevertheless, some explicit invocation means may be required to interact with a service function.

To see how standard interfaces could help achieve faster implementation time cycles, let us consider a customer who would like to dynamically allow an encrypted flow with specific port, src/dst addresses or protocol type through the firewall/IPS to enable an encrypted video conferencing call only during the time of the call. With no commonly accepted interface in place, the customer would have to learn about the particular provider's firewall/IPS interface and send the request in the provider's required format.

```
        +------------+
        | security   |
        | MGT system |
        +----||------+
             ||   proprietary
             ||   or I2NSF standard
Picture:     ||
Port 10   +---------+
  --------| FW/IPS |-------------
Encrypted +---------+
Video Flow
```

Figure 2: Example of non-standard vs. standard interface

In contrast, if a firewall/IPS interface standard exists, the customer would be able to send the request, without having to do the extensive preliminary legwork. A standard interface also helps service providers since they could now offer the same firewall/IPS interface to represent firewall/IPS services for utilizing products from many vendors. The result is that the service provider has now abstracted the firewall/IPS services. The standard interface also helps the firewall/IPS vendors to focus on their core security functions or extended features rather than the standard building blocks of a management interface.

3.2.3. Difficulty to Monitor the Execution of Desired Policies

How a policy is translated into technology-specific actions is hidden from the customers. However, customers still need ways to monitor the delivered security service that results from the execution of their desired security requirements, guidelines and expectations.

Today, there is no standard way for customers to get security service assurance of their specified security policies properly enforced by the security functions in the provider domain. The customer also lacks the ability to perform "what-if" scenarios to assess the efficiency of the delivered security service.

3.3. Difficulty to Validate Policies across Multiple Domains

One key aspect of a hosted security service with security functions located at different premises is the ability to express, monitor and verify security policies that combine several distributed security functions. It is crucial to an effective service to be able to take these actions via a standard interface. This standard interface becomes more crucial to the hosted security service when NSFs are instantiated in Virtual Machines which are sometimes widely distributed in the network and sometimes are combined together in one device to perform a set of asks in a service.

Without standard interfaces and security policy data models, the enforcement of a customer-driven security policy remains challenging because of the inherent complexity created by the combining the invocation of several vendor-specific security functions into a multi-vendor, heterogeneous environment. Each vendor specific function may require specific configuration procedures and operational tasks.

Ensuring the consistent enforcement of the policies at various domains is also challenging. Standard data models are likely to contribute to ameliorating that issue.

3.4. Lack of Standard Interface to Inject Feedback to NSF

Today, many security functions, such as IPS, IDS, DDoS and Antivirus, depend heavily on the associated profiles. They can perform more effective protection if they have the up-to-date profiles. As more sophisticated threats arise, enterprises, vendors, and service providers have to rely on each other to achieve optimal protection. Cyper Threat Alliance (CA, http://cyberthreatalliance.org/) is is one of those initiatives that aim at combining efforts conducted by multiple organizations.

Today there is no standard interface to exchange security profiles between organizations.

3.5. Lack of Standard Interface for Capability Negotiation

There could be situations when the NSFs selected cannot perform the policies requested by the Security Controller, due to resource constraints. To support the automatic control in the SDN-era, it is necessary to have a set of messages for proper negotiation between the Security Controller and the NSFs.

4. Use Cases

Standard interfaces for monitoring and controlling the behavior of NSFs are essential building blocks for Security Service Providers and enterprises to automate the use of different NSFs from multiple vendors by their Security management entities. I2NSF may be invoked by any (authorized) client. Examples of authorized clients are upstream applications (controllers), orchestration systems, and security portals.

4.1. General Use Cases

User request security services through specific clients (e.g. a customer application, the NSP BSS/OSS or management platform) and the appropriate NSP network entity will invoke the (v)NSFs according to the user service request. We will call this network entity the security controller. The interaction between the entities discussed above (client, security controller, NSF) is shown in the following diagram:

```
                        +----------+
 +-------+              |          |                 +-------+
 |       |  Interface 1 |Security  |   Interface 2   | NSF(s)|
 |Client <-------------->          <------------------->       |
 |       |              |Controller|                 |       |
 +-------+              |          |                 +-------+
                        +----------+
```

Figure 2: Interaction between Entities

Interface 1 is used for receiving security requirements from client and translating them into commands that NSFs can understand and execute. The security controller also passes back NSF security reports (e.g. statistics) to the client which the control has gathered from NSFs. Interface 2 is used for interacting with NSFs according to commands, and collect status information about NSFs.

Client devices or applications can require the security controller to add, delete or update rules in the security service function for their specific traffic.

When users want to get the executing status of security service, they can request the information of NSFs from the client. The security controller will collect NSF information through Interface 2, consolidate them, and give feedback to client through Interface 1.

This interface can be used to collect not only individual service information, but also aggregated data suitable for tasks like infrastructure security assessment.

Customers may require validating NSF availability, provenance, and correct execution. This validation process, especially relevant for vNSFs, includes at least:

Integrity of the NSF: by ensuring that the NSF is not compromised;

Isolation: by ensuring the execution of the NSF is self-contained for privacy requirements in multi-tenancy scenarios.

In order to achieve this, the security controller may collect security measurements and share them with an independent and trusted third party (via the interface 1) in order to allow for attestation of NSF functions using the third party added information.

4.2. Access Networks

This scenario describes use cases for users (e.g. enterprise user, network administrator, and residential user) that request and manage security services hosted in the network service provider (NSP) infrastructure. Given that NSP customers are essentially users of their access networks, the scenario is essentially associated with their characteristics, as well as with the use of vNSFs.

The Virtual CPE described in [NFVUC] use cases #5 and #7 requires a model of access virtualization that includes mobile and residential access where the operator may offload security services from the customer local environment (E.g. device or terminal) to the operator infrastructure supporting the access network.

These use cases defines the operator interaction with vNSFs through automated interfaces, typically by B2B communications.

```
            Customer   +     Access     +     PoP/Datacenter
                       |                |     +--------+
                       |          ,-----+--.  |Network |
                       |        ,'      |   '-|Operator|
       +-------------+ |       /+----+  |     |Mgmt Sys|
       | Residential |-+------/-+vCPE+-----+  +--------+
       +-------------+ |     /  +----+  |  \     |    :
                       |    /           |   \    |     |
           +-------+   |   ;    +----+  |    +----+    |
           | Cloud |---+---+----+ vPE+--+----+ NSF|    |
           +-------+   |   :    +----+  |    +----+    |
                       |    :           |   /          |
            +--------+ |    :   +----+  |  /           ;
            | Mobile |-+-----\--+vEPC+----+           /
            +--------+ |      \ +----+  |          ,-'
                       |       '--.     |      _.-'
                       |           '----+----''
                       +                +
```

Figure 3: NSF and actors

The following are actions required for this access use case:

vNSF Deployment: The deployment process consists of instantiating a NSF on a Virtualization Infrastructure (NFVI), within the NSP administrative domain(s) or with other external domain(s). This is a required step before a customer can subscribe to a security service supported in the vNSF.

vNSF Customer Provisioning: Once a vNSF is deployed, any customer can subscribe to it. The provisioning lifecycle includes the following:

* Customer enrollment and cancellation of the subscription to a vNSF;

* Configuration of the vNSF, based on specific configurations, or derived from common security policies defined by the NSP.

* Retrieve and list of the vNSF functionalities, extracted from a manifest or a descriptor. The NSP management systems can demand this information to offer detailed information through the commercial channels to the customer.

4.3. Cloud Datacenter Scenario

In a datacenter, network security mechanisms such as firewalls may need to be added or removed dynamically for a number of reasons. These changes may be explicitly requested by the user, or triggered by a pre-agreed upon service level agreement (SLA) between the user and the provider of the service. For example, the service provider may be required to add more firewall capacity within a set timeframe whenever the bandwidth utilization hits a certain threshold for a specified period. This capacity expansion could result in adding new instances of firewalls on existing machines or provisioning a completely new firewall instance in a different machine.

The on-demand, dynamic nature of deployment essentially requires that the network security "devices" be in software or virtual form factors, rather than in a physical appliance form. This requirement is a provider-side concern. Users of the firewall service are agnostic (as they should) as to whether or not the firewall service is run on a VM or any other form factor. Indeed, they may not even be aware that their traffic traverses firewalls.

Furthermore, new firewall instances need to be placed in the "right zone" (domain). The issue applies not only to multi-tenant environments where getting the tenant in the right domain is of paramount importance, but also in environments owned and operated by a single organization with its own service segregation policies. For example, an enterprise may mandate that firewalls serving Internet traffic and business-to-business (B2B) traffic be separate. Another example is that IPS/IDS services for investment banking and non-banking traffic may be need to separated for regulatory reasons.

4.3.1. On-Demand Virtual Firewall Deployment

A service provider operated cloud data center could serve tens of thousands of clients. Clients' compute servers are typically hosted on virtual machines (VMs), which could be deployed across different server racks located in different parts of the data center. Often it is not technically and/or financially feasible to deploy dedicated physical firewalls to suit each client's myriad security policy requirements. What is needed is the ability to dynamically deploy virtual firewalls for each client's set of servers based on established security policies and underlying network topologies.

```
           ---+-----------------------------+-----
              |                             |
            +---+                         +-+-+
            |vFW|                         |vFW|
            +---+                         +-+-+
              |    Client #1                |  Client #2
           ---+-------+---             ---+-------+---
            +-+-+   +-+-+                 +-+-+   +-+-+
            |vM |   |vM |                 |vM |   |vM |
            +---+   +---+                 +---+   +---+
```

Figure 4: NSF in Data Center

4.3.2. Firewall Policy Deployment Automation

Firewall rules setting is often a time consuming, complex and error-prone process even within a single organization/enterprise framework. It becomes far more complex in provider-owned cloud networks that serve myriad customers.

Firewall rules today are highly tied with ports and addresses of the traffic. This makes it very difficult for clients of cloud data center to construct rules for their own traffic as the clients only see the virtual networks and the virtual addresses. The customer-visible virtual networks and addresses may be different from the actual packets traversing the FWs.

Even though most vendors support similar firewall features, the actual rule configuration key words are different from vendors to vendors, making it difficult for automation. Automation works best when it can leverage a common set of standards that will work across NSFs by multiple vendors. Without automation, it is virtually impossible for clients to dynamically specify their desired rules for their traffic.

4.3.3. Client-Specific Security Policy in Cloud VPNs

Clients of service provider operated cloud data centers need not only secure virtual private networks (VPNs) but also virtual security functions that enforce the clients’ security policies. The security policies may govern communication within the clients’ own virtual networks as well as communication with external networks. For example, VPN service providers may need to provide firewall and other security services to their VPN clients. Today, it is generally not possible for clients to dynamically view (much less change) what, where and how security policies are implemented on their provider-operated clouds. Indeed, no standards-based framework that allows clients to retrieve/manage security policies in a consistent manner across different providers exists.

4.3.4. Internal network monitoring

There are many types of internal traffic monitors that may be managed by a security controller. This includes a new class of services referred to as DLP, Data Loss Prevention, or Reputation Protection Services. Depending on the class of event, alerts may go to internal administrators, or external services.

5. Management Considerations

Management of NSFs usually include configuration of devices, signaling and policy provisioning. I2NSF will only focus on the policy provisioning part.

6. IANA Considerations

No IANA considerations exist for this document.

7. Security Considerations

Having a secure access to control and monitor NSFs is crucial for hosted security service. The new NSF security controller introduces a new attack surface. It needs to be resilient to attack and recovery from attack needs to be quick and trivial (thus making attacking it ’uninteresting’). Therefore, proper secure communication channels have to be carefully specified for carrying the controlling and monitoring information between the NSFs and their management entity (or entities).

8. Contributors

I2NSF is a group effort. The following people contributed actively to the initial use case text: Diego R. Lopez (Telefonica I+D), Xiaojun Zhuang (China Mobile), Minpeng Qi (China Mobile), Sumandra Majee (F5), Nic Leymann (Deutsche Telekom), Ed Lopez (Fortinet), and Robert Moskowitz (Huawei).

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

9.2. Informative References

[Gartner-2013]
Messmer, E., "Gartner: Cloud-based security as a service set to take off", October 2013.

[I-D.hares-i2nsf-gap-analysis]
Hares, S., Zhang, D., Moskowitz, R., and H. Rafiee, "Analysis of Existing work for I2NSF", draft-hares-i2nsf-gap-analysis-00 (work in progress), July 2015.

[I-D.ietf-netmod-acl-model]
Bogdanovic, D., Koushik, K., Huang, L., and D. Blair, "Network Access Control List (ACL) YANG Data Model", draft-ietf-netmod-acl-model-06 (work in progress), December 2015.

[I-D.ietf-opsawg-firewalls]
Baker, F. and P. Hoffman, "On Firewalls in Internet Security", draft-ietf-opsawg-firewalls-01 (work in progress), October 2012.

[I-D.jeong-i2nsf-sdn-security-services]
Jeong, J., Kim, H., and P. Jung-Soo, "Requirements for Security Services based on Software-Defined Networking", draft-jeong-i2nsf-sdn-security-services-01 (work in progress), March 2015.

[I-D.lopez-i2nsf-packet]
Ed, E., "Packet-Based Paradigm For Interfaces To NSFs", draft-lopez-i2nsf-packet-00 (work in progress), March 2015.

[I-D.pastor-i2nsf-access-usecases]
Pastor, A. and D. Lopez, "Access Use Cases for an Open OAM Interface to Virtualized Security Services", draft-pastor-i2nsf-access-usecases-00 (work in progress), October 2014.

[I-D.pastor-i2nsf-merged-use-cases]
Pastor, A., Lopez, D., Wang, K., Zhuang, X., Qi, M., Zarny, M., Majee, S., Leymann, N., Dunbar, L., and M. Georgiades, "Use Cases and Requirements for an Interface to Network Security Functions", draft-pastor-i2nsf-merged-use-cases-00 (work in progress), June 2015.

[I-D.qi-i2nsf-access-network-usecase]
Wang, K. and X. Zhuang, "Integrated Security with Access Network Use Case", draft-qi-i2nsf-access-network-usecase-02 (work in progress), March 2015.

[I-D.zarny-i2nsf-data-center-use-cases]
Zarny, M., Leymann, N., and L. Dunbar, "I2NSF Data Center Use Cases", draft-zarny-i2nsf-data-center-use-cases-00 (work in progress), October 2014.

[I-D.zhou-i2nsf-capability-interface-monitoring]
Zhou, C., Xia, L., Boucadair, M., and J. Xiong, "The Capability Interface for Monitoring Network Security Functions (NSF) in I2NSF", draft-zhou-i2nsf-capability-interface-monitoring-00 (work in progress), October 2015.

[RFC4948] Andersson, L., Davies, E., and L. Zhang, "Report from the IAB workshop on Unwanted Traffic March 9-10, 2006", RFC 4948, DOI 10.17487/RFC4948, August 2007, <http://www.rfc-editor.org/info/rfc4948>.

[RFC7277] Bjorklund, M., "A YANG Data Model for IP Management", RFC 7277, DOI 10.17487/RFC7277, June 2014, <http://www.rfc-editor.org/info/rfc7277>.

Authors' Addresses

Susan Hares
Huawei
7453 Hickory Hill
Saline, MI 48176
USA

Phone: +1-734-604-0332
Email: shares@ndzh.com

Linda Dunbar
Huawei
5340 Legacy Drive, Suite 175
Plano, TX 75024
USA

Phone: +1-734-604-0332
Email: ldunbar@huawei.com

Antonio Pastor
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid 28006
Spain

Email: antonio.pastorperales@telefonica.com

Diego R. Lopez
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid 28006
Spain

Phone: +34 913 129 041
Email: diego.r.lopez@telefonica.com

Myo Zarny
Goldman Sachs
30 Hudson Street
Jersey City, NJ 07302
USA

Email: myo.zarny@gs.com

Nic Leymann
Deutsche Telekom

Email: N.Leymann@telekom.de

Michael Georgiades
Prime Tel

Email: michaelq@prime-tel.com

Minpeng Qi
China Mobile
32 Xuanwumenxi Ave,Xicheng District
Beijing 100053
China

Email: qiminpeng@chinamobile.com

Mohamed Boucadair
France Telecom
Rennes, 35000
France

Email: mohamed.boucadair@orange.com

Christian Jacquenet
France Telecom
Rennes, 35000
France

Email: Christian.jacquenet@orange.com

Shaibal Chakrabarty
US Ignite
1776 Massachusetts Ave NW, Suite 601
Washington, DC 20036
USA

Phone: (214) 708-6163
Email: shaibalc@us-ignite.org