International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.1043

(03/2019)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Information and network security – Network security

Security framework and requirements for service function chaining based on software-defined networking

Recommendation ITU-T X.1043

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	X.1–X.199
OPEN SYSTEMS INTERCONNECTION	X.200–X.299
INTERWORKING BETWEEN NETWORKS	X.300–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600–X.699
OSI MANAGEMENT	X.700–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	X.850–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	
General security aspects	X.1000–X.1029
Network security	**X.1030–X.1049**
Security management	X.1050–X.1069
Telebiometrics	X.1080–X.1099
SECURE APPLICATIONS AND SERVICES (1)	
Multicast security	X.1100–X.1109
Home network security	X.1110–X.1119
Mobile security	X.1120–X.1139
Web security	X.1140–X.1149
Security protocols (1)	X.1150–X.1159
Peer-to-peer security	X.1160–X.1169
Networked ID security	X.1170–X.1179
IPTV security	X.1180–X.1199
CYBERSPACE SECURITY	
Cybersecurity	X.1200–X.1229
Countering spam	X.1230–X.1249
Identity management	X.1250–X.1279
SECURE APPLICATIONS AND SERVICES (2)	
Emergency communications	X.1300–X.1309
Ubiquitous sensor network security	X.1310–X.1319
Smart grid security	X.1330–X.1339
Certified mail	X.1340–X.1349
Internet of things (IoT) security	X.1360–X.1369
Intelligent transportation system (ITS) security	X.1370–X.1389
Distributed ledger technology security	X.1400–X.1429
Distributed ledger technology security	X.1430–X.1449
Security protocols (2)	X.1450–X.1459
CYBERSECURITY INFORMATION EXCHANGE	
Overview of cybersecurity	X.1500–X.1519
Vulnerability/state exchange	X.1520–X.1539
Event/incident/heuristics exchange	X.1540–X.1549
Exchange of policies	X.1550–X.1559
Heuristics and information request	X.1560–X.1569
Identification and discovery	X.1570–X.1579
Assured exchange	X.1580–X.1589
CLOUD COMPUTING SECURITY	
Overview of cloud computing security	X.1600–X.1601
Cloud computing security design	X.1602–X.1639
Cloud computing security best practices and guidelines	X.1640–X.1659
Cloud computing security implementation	X.1660–X.1679
Other cloud computing security	X.1680–X.1699

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.1043

Security framework and requirements for service function chaining based on software-defined networking

Summary

Recommendation ITU-T X.1043 analyses security threats to and specifies security requirements for service function chaining based on software-defined networking (SDN). The corresponding security countermeasures are also given. Recommendation ITU-T X.1043 also aims to help understanding of security risks encountered when using SDN-based service function chaining and implementation of secured SDN-based service function chains.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T X.1043	2019-03-16	17	11.1002/1000/13872

Keywords

SDN, service function chain, SFC, software-defined networking.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Introduction

Service function chaining enables administrators to distribute network policies more efficiently and conveniently, while software-defined networking (SDN) helps to adjust traffic dynamically according to changing requirements. These two technologies can be combined to give a network higher flexibility and stronger capability to support on-demand services.

However, it is clear that SDN-based service function chaining also introduces new security challenges to the network, not only legacy threats associated with SDN, but also new threats associated with the service function chain (SFC).

Use of SDN architecture means that common networking threats will show up in SDN-based service function chaining, e.g., more severe denial of service/distributed denial of service (DoS/DDoS) attacks caused by a centralized SDN controller and attacks on the application-control interface.

More security issues arise because of the deployment of service function chaining. New network elements are introduced, like an SFC forwarder, and new information is transferred among network elements, like SFC classification rules and service function path (SFP) forwarding rules. Threats against these entities should be mitigated. Thus there is an urgent need for the security guidelines in this Recommendation.

Recommendation ITU-T X.1043

Security framework and requirements for service function chaining based on software-defined networking

1 Scope

This Recommendation analyses security threats encountered in service function chaining based on software-defined networking (SDN) and specifies security guidelines for SDN-based service function chaining architectures. This Recommendation:

– describes a general security architecture for SDN-based service function chaining;

– analyses security threats to and requirements of network elements and corresponding interfaces in the SDN-based service function chaining architecture;

– describes and analyses policy management problems in SDN-based service function chaining;

– suggests countermeasure solutions to meet these requirements.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T X.800] Recommendation ITU-T X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications.*

[ITU-T X.1038] Recommendation ITU-T X.1038 (2016), *Security requirements and reference architecture for software-defined networking*.

[ITU-T Y.3300] Recommendation ITU-T Y.3300 (2014), *Framework of software-defined networking*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 classification [b-IETF RFC 7665]: Locally instantiated matching of traffic flows against policy for subsequent application of the required set of network service functions. The policy may be customer/network/ service specific.

3.1.2 classifier [b-IETF RFC 7665]: An element that performs classification.

3.1.3 metadata [b-IETF RFC 7665]: Provides the ability to exchange context information between classifiers and SFs, and among SFs.

3.1.4 service function (SF) [b-IETF RFC 7665]: A function that is responsible for specific treatment of received packets. A service function can act at various layers of a protocol stack (e.g., at the network layer or other OSI layers). As a logical component, a service function can be realized as a virtual element or be embedded in a physical network element. One or more service functions can be embedded in the same network element. Multiple occurrences of the service function can exist in the same administrative domain.

3.1.5 service function chain (SFC) [b-IETF RFC 7665]: A service function chain defines an ordered set of abstract service functions and ordering constraints that must be applied to packets and/or frames and/or flows selected as a result of classification.

3.1.6 service function forwarder (SFF) [b-IETF RFC 7665]: A service function forwarder is responsible for forwarding traffic to one or more connected service functions according to information carried in the SFC encapsulation, as well as handling traffic coming back from the SF. Additionally, an SFF is responsible for delivering traffic to a classifier when needed and supported, transporting traffic to another SFF (in the same or different type of overlay), and terminating the service function path (SFP).

3.1.7 service function path (SFP) [b-IETF RFC 7665]: The service function path is a constrained specification of where packets assigned to a certain service function path must go. While it may be so constrained as to identify the exact locations, it can also be less specific. The SFP provides a level of indirection between the fully abstract notion of service chain as a sequence of abstract service functions to be delivered, and the fully specified notion of exactly which SFF/SFs the packet will visit when it actually traverses the network. By allowing the control components to specify this level of indirection, the operator may control the degree of SFF/SF selection authority that is delegated to the network.

3.1.8 SFC-enabled domain [b-IETF RFC 7665]: A network or region of a network that implements SFC. An SFC-enabled domain is limited to a single network administrative domain.

3.1.9 SFC encapsulation [b-IETF RFC 7665]: The SFC encapsulation provides, at a minimum, SFP identification, and is used by the SFC-aware functions, such as the SFF and SFC-aware SFs. The SFC encapsulation is not used for network packet forwarding. In addition to SFP identification, the SFC encapsulation carries metadata including data-plane context information.

3.1.10 SFC proxy [b-IETF RFC 7665]: Removes and inserts SFC encapsulation on behalf of an SFC-unaware service function. SFC proxies are logical elements.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 service function chain classification rule: A rule generated and maintained by a service function chain (SFC) controller and classifier, respectively. It reflects the policies for binding an incoming flow to a given SFC and service function path (SFP). An SFC classification rule can be translated into an SFC flow rule by the software-defined networking (SDN) controller and formed into an entry in an SFC classification table, like an SDN flow entry in an SDN flow table.

3.2.2 service function chain controller: A function in a software-defined networking (SDN) controller that instructs the functional elements on the service function chain (SFC) resource layer to process packets within an SFC-enabled domain. After receiving the SFC requirements from the applications (apps), the SFC controller translates the requirements into the SFC classification rules and service function path (SFP) forwarding rules and sends them to the classifiers and the service function forwarders (SFFs), respectively, via the SDN controller.

3.2.3 service function chain flow rule: A flow rule on the classifiers and the service function forwarders (SFFs) that are translated by the software-defined networking (SDN) controller from the service function chain (SFC) classification rule and the service function path (SFP) forwarding rule.

3.2.4 service function path forwarding rule: A rule generated and maintained by a service function chain (SFC) controller and a service function forwarder (SFF), respectively. It reflects the policies for forwarding an incoming flow to a given service function (SF). A service function path (SFP) forwarding rule can be translated into an SFC flow rule by the software-defined networking

(SDN) controller and formed into an entry in an SFP) forwarding rule table like an SDN flow entry in an SDN flow table.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ACL	Access Control List
AES	Advanced Encryption Standard
app	application
DDoS	Distributed Denial of Service
DoS	Denial of Service
HMAC	Hash-based Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
IPS	Intrusion Prevention System
IPSec	Internet Protocol Security
LSP	Label-Switched Path
MPLS	Multi-Protocol Label Switching
NBI	Northbound Interface
NSH	Network Service Header
OSI	Open Systems Interconnection
PSK	Pre-Shared Key
RBAC	Role-Based Access Control
SDN	Software-Defined Networking
SF	Service Function
SFC	Service Function Chain
SFF	Service Function Forwarder
SFP	Service Function Path
SI	Service Index
SPI	Service Path Identifier
TLS	Transport Layer Security
TTL	Time To Live
VNF	Virtualized Network Function

5 Conventions

In this Recommendation:

The keywords "**is required to**" indicate a requirement that must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

The keywords "**is recommended**" indicate a requirement that is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

The keywords "**is prohibited from**" indicate a requirement that must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

The keywords "**can optionally**" indicate an optional requirement that is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option, and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the specification.

6 Overview

The high-level architecture of SDN is specified in [ITU-T Y.3300]. It includes three layers: application layer, control layer and resource layer. The SDN control layer provides a means to dynamically and deterministically control the behaviour of network resources (e.g., data transport and processing), as instructed by the application layer. The features of the SDN (i.e., decoupled control function and transportation function, centralized control layer) are suitable for the implementation of service function chaining: the SDN control layer can program the service function chaining policy from the application layer and control the resource layer to forward packages/flows according to such policy.

Based on the high-level architecture of SDN, Figure 6-1 shows a general reference architecture of an SDN-based service function chaining as a basis for security analysis.

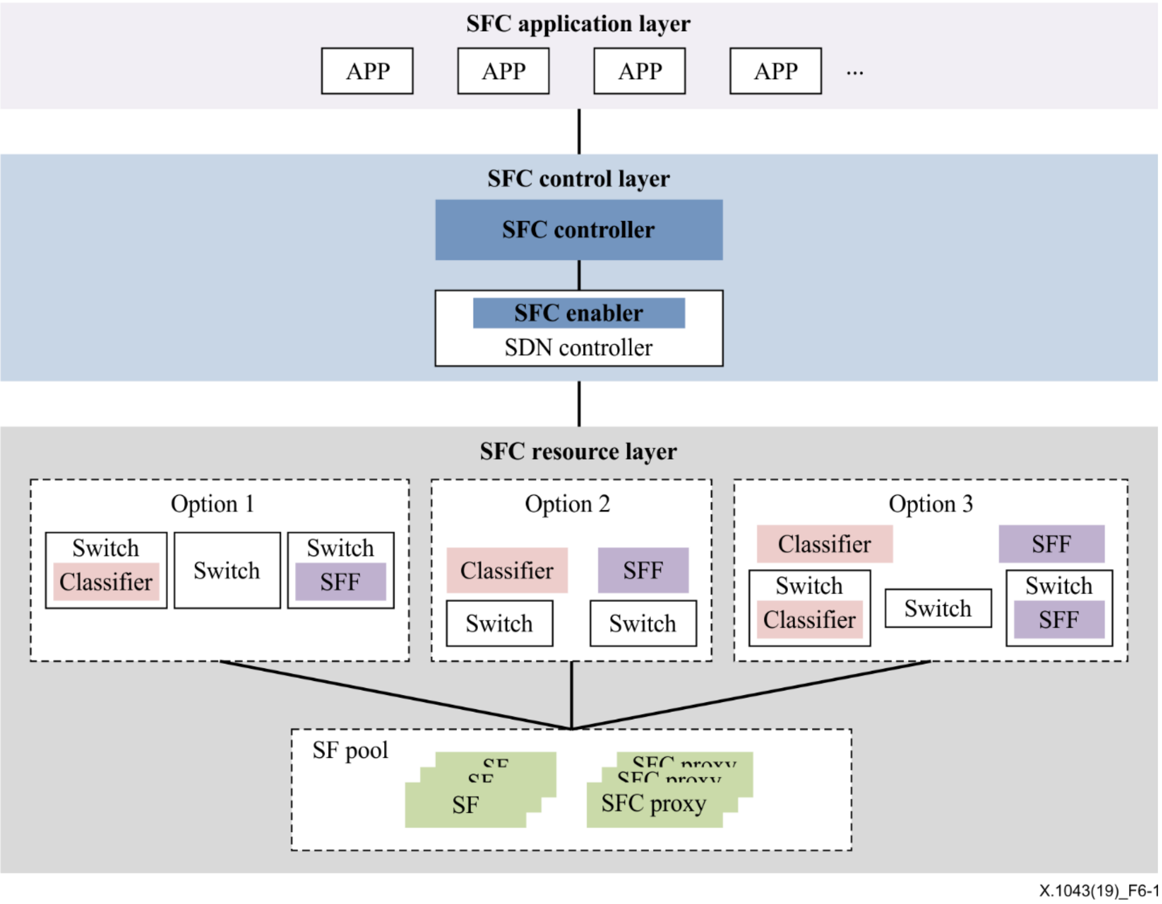

Figure 6-1 – General reference architecture of software-defined networking-based service function chaining

In Figure 6-1, the definitions of classifier, service function forwarder (SFF), service function (SF) and SFC proxy are those of [b-IETF RFC 7665]. There are three options to deploy the classifier and SFF on the SFC resource layer: 1) classifier and SFF are embedded into switches; 2) classifier and SFF are deployed in independent devices; 3) the implementation of classifier and SFF includes the combination of options 1) and 2). As for these three options, a classification table and an SFP table based on the OpenFlow flow table specified in [b-ONF TS-025] are defined in this Recommendation in order to reflect SFC rules, i.e., the classification rule and SFP forwarding rule. The detailed formats of the classification table and the SFP table are specified in Annex A.

The SFC control function (referred to as an SFC controller in this Recommendation) can be implemented as an application of an SDN controller or as a logical function embedded in an SDN controller. This means that the interface between the SFC controller and the SDN controller can be either an application-control interface or a private interface, depending on the implementation.

The SFC enabler in the SDN controller is responsible for: 1) receiving and responding to the request for network information from the SFC controller in order to create valid SFCs; and 2) receiving classification and SFP forwarding rules (i.e., the flow entry of the classification table and SFP table) from the SFC controller and combining them with OpenFlow rules for distribution to the SFC resource layer.

The components and functions of each layer are as follows.

- SFC application layer: This layer refers to the application layer in the SDN architecture of [ITU-T Y.3300] where apps can send user SFC requirements to the SFC controller to customize the behaviour of user flows. The apps can also request SFC information from the SFC controller.
- SFC control layer: This layer is like the control layer in the SDN architecture [ITU-T Y.3300], except that the SFC controller is included.
    - The SFC controller is responsible for programming SFC classification rules and SFP forwarding rules according to SFC policies received from apps. Before transporting the SFC classification rules and the SFP forwarding rules to the SDN controller, the SFC controller checks whether these new SFC classification rules and SFP forwarding rules conflict with the stored active SFC classification rules and SFP forwarding rules in the SFC repository of the SFC controller.
    - When the SFC controller is implemented as an application of the SDN controller, the SFC enabler in the SDN controller receives SFC rules from the SFC controller, combines them with the OpenFlow flow rules and sends the combined rules to related entities in the SFC resource layer. The detailed processes are as follows.
        - The SFC enabler in the SDN controller combines the SFC classification rules and SFP forwarding rules with OpenFlow flow rules after receiving them from the SFC controller. The SDN controller programs the combined rules of the classifiers, SFFs and switches, respectively, and then forwards packages/flows according to SFC classification rules, SFP forwarding rules and OpenFlow flow rules. In order to support SFC classification rules and SFP forwarding rules, the classification table and the SFP table specified in Annex A are required.
        - The SDN controller processes and coordinates the policy conflict between the translated flow rules from the SFC classification rules and SFP forwarding rules and the stored SDN flow rules.
        - The SDN controller sends these translated flow rules to the related classifiers, SFFs and switches, respectively.
- SFC resource layer: This layer includes switches, classifiers, SFFs, SFC proxies and SFs.
    - The classifiers and the SFFs process the flows according to the flow rules. The classifier classifies the flows and adds the SFC encapsulation (e.g., the network service header (NSH) [b-IETF RFC 8300]) into the packet that is transmitted to the SFFs, SFs, etc. The classifier sends the flows to the SFF after adding the SFC encapsulation. The SFF transports the flows to SFs/SFC proxies or the next SFFs after receiving the flows from classifiers/other SFFs or SFs/SFC proxies, respectively, according to SFP forwarding

rules. The classifier and SFF can be implemented on the switches. In this case, the classifier and SFF need to be registered in the SFC controller, and the switches that support classifier/SFF functions need to be indicated to the SDN controller that programs the flow forwarding path.

- The SFC-aware SFs are responsible for processing the received flows and also need to register in the SFC controller and provide their status to the SFC controller by the interfaces between the SFC controller and the SFs or through the management element. The SF can be a virtualized network function (VNF) or a physical device.
- The SFC proxies are responsible for removing and inserting SFC encapsulation on behalf of any SFC-unaware service functions. They can also inform the SFC controller of SF status by the interface between the SFC controller and the SFs or through the management element. SFC proxies are logical elements.

According to Figure 6-1, the following interfaces are included.

- SFC application-control interface: The interface between the application layer and the control layer. The interface between the SFC controller and the SDN controller can also be an application-control interface or a private interface. This interface is mainly used to transmit policies (e.g., SFC policy, classification rule).
- SFC resource-control interface: The interface between the control layer and the resource layer, e.g., the interface between the SDN controller and the SFF. This interface is mainly used to transmit flow rules.
- SFC intra-interfaces in the resource layer: This includes the interface between the classifier and the SFF, the interface between the SFF and the SF, the interface between the SFF and the SFC proxy, and the SFC proxy and the SF. This interface is mainly used to transmit data flows.
- Management interface between the SFC controller and the SF/SFC proxy: The SFs/SFC proxies use this interface to register the SFs in the SFC controller and to send SF status. The SFC controller requests SF status and configures the SFs with this interface. This interface can be implemented by a direct interface between the SFC controller and SF/SFC proxy or an indirect interface through a management element between the SFC controller and SF/SFC proxy. Because there can be many implementations for this interface, it is not described in Figure 6-1.

7 General security framework of software-defined networking-based service function chaining

The security reference architecture for SDN is specified in [ITU-T X.1038] and can be applied to SDN-based service function chaining, with the addition of some specific security features. A general security architecture of SDN-based service function chaining is shown in the next paragraph, but only emphasizes security specific to the SDN-based service function chaining.

Three security feature groups are defined in Figure 7-1.

(I) Critical network elements security: A set of security features that provides security functions on network entities to support secure creation, running, maintenance and deletion of an SFC.

(II) Interface security: A set of security features that provides security functions to ensure secure transportation of communication data.

(III) Policy management: A set of security features that provides policy lifecycle security, e.g., the policy is created by a legal SFC application (app), sent with security protection and implemented correctly. It also resolves the SFC policy conflict, e.g., the conflict between the new SFC classification rules and the stored active SFC classification rules in the SFC repository of the SFC controller, the conflict between the translated flow rules from the SFC classification rules and the traditional SDN flow rules on the SDN controller.

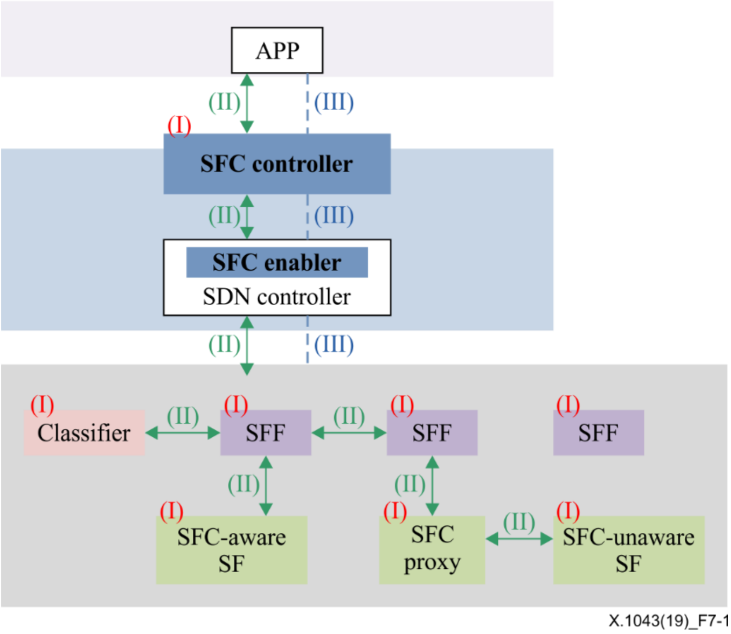

Figure 7-1 – General security framework of software-defined networking-based service function chaining

8 Threat analysis and requirements of critical network elements

8.1 Critical network elements

For SDN-based service function chaining, the critical network elements include:

– SFC application layer: app;

– SFC control layer: SDN controller, SFC controller;

– SFC resource layer: classifier, SFF, SF, SFC proxy, switch.

The security threats and requirements of the app, SDN controller and switch are described in [ITU-T X.1038].

8.2 Security threats and requirements

In this Recommendation, the SFC is SDN based, thus the security threats and security requirements of [ITU-T X.1038] apply. Clauses 8.2.1 to 8.2.5 focus only on the specific security threats and security requirements of SDN-based SFCs.

8.2.1 Security threats to and requirements of service function chain controller

8.2.1.1 Security threats

The SFC controller is an important component that programs SFC classification and SFP forwarding rules. It has the following security threats.

– **Service function chain classification rule/service function path forwarding rule conflict**: An SFC classification rule and an SFP forwarding rule can conflict with other SFC classification rules and SFP forwarding rules, respectively, stored on the SFC controller. An attacker may use the SFC classification rule conflict and the SFP forwarding rule conflict to bypass security controls.

- **Fake service function chain classification rule/service function path forwarding rule insertion**: An attacker can use a malicious app to send an SFC requirement to an SFC controller. This may cause the SFC controller to program an SFC classification rule or SFP forwarding rule to steer a flow that bypasses some SFs; thus the flow cannot be processed correctly.
- **Untrusted software-defined networking controller**: A fake SDN controller can receive SFC classification rules/SFP forwarding rules from an SFC controller, when the SFC controller is implemented as an app on the SDN controller, and can result in sensitive data leakage. When the SFC controller is implemented as a function in the SDN controller, it can also be attacked when the controller is attacked.
- **Spoofing**: An attacker can pose as a legal app or an administrator to communicate with the SFC controller to launch attacks. For example, the spoofing app can send a user SFC requirement and cause a new SFC classification rule and a new SFP forwarding rule to be installed. These rules can change the path of the flows and cause further attacks.
- **Repudiation**: The administrators of the SFC controller can repudiate behaviours, such as SFC classification rule tampering.
- **Information disclosure**: An attacker can access sensitive data on the SFC controller (e.g., SFC classification rules, SFP forwarding rules, administrator password), which can result in other attacks. For example, an attacker utilizes a leaked administrator password on the SFC controller to tamper with the SFC classification rule to cause further attacks.
- **Fake service function status**: An attacker can send a fake SF status to the SFC controller, which can lead to an incorrect SFC classification rule/SFP forwarding rule being programmed by the SFC controller.
- **Security vulnerability on the service function chain controller**: An attacker can access the SFC controller by exploiting security vulnerabilities on the SFC controller (e.g., code flaw) to compromise the SFC controller and the SDN controller.
- **Denial of service attack**: An attacker can utilize a large number of apps to send SFC requirements to the SFC controller at the same time. Then, the SFC controller translates these SFC requirements into SFC classification rules and SFP forwarding rules, which may cause an overload on the SFC controller. The SFs/SFC proxies can also launch a DoS attack on the SFC controller. For example, a large number of SFs can send a fake SF status to cause the SFC controller to change the SFC classification rule and SFP forwarding rule continually until the SFC controller is overloaded.

8.2.1.2 Security requirements

R-01: The SFC controller is recommended to support policy conflict to prevent bypassing important policies (e.g., security policy, management policy).

R-02: The SFC controller is required to authenticate the app.

R-03: The SFC controller is required to authenticate the administrator.

R-04: The SFC controller is required to authenticate the SDN controller.

R-05: The SFC controller is required to authenticate the SF/SFC proxy.

R-06: The SFC controller is required to authenticate the management.

R-07: The SDN controller is required to satisfy the requirements of clause 7.2.2 of [ITU-T X.1038].

R-08: The SFC controller is required to support log and audit function.

R-09: The SFC controller is required to authorize apps to provide SFC requirements or request SFC information.

R-10: The SFC controller is required to authorize the administrator to manage the SFC controller.

R-11: The SFC controller is required to verify the integrity of the status of the SF and authenticate the identity of the sender if SF status is transmitted by the management element.

R-12: The SFC controller is recommended to protect the confidentiality and integrity [ITU-T X.800] of stored sensitive data, e.g., SFC classification rules, SFP forwarding rules, SFC category and available SFs.

R-13: The SFC controller is recommended to provide the key/certificate management function to support authentication/authorization and data protection.

R-14: The SFC controller is recommended to support vulnerability detection and patch functions to mitigate security threats from the SFC controller software.

R-15: The SFC is recommended to provide mechanisms for anti-DoS, e.g., limiting requests from the classifiers/SFFs or SFC apps.

8.2.2 Security threats to and requirements of a classifier

8.2.2.1 Security threats

Major specific security threats to the classifier are as follows.

– **Spoofing**: An attacker can pose as a legal SDN controller to communicate with the classifier to try to launch spoofing attacks. For example, the spoofed SDN controller can make a fake FSC flow rule, thus causing an incorrect classification.

– **Repudiation**: The administrators on the classifier can repudiate their behaviours, such as SFC flow rule tampering.

– **Information disclosure**: An attacker can access sensitive data on the classifier. For example, an attacker can access SFC flow rules to acquire all flow classifications, and then the attacker can derive the SFC topology.

– **Untrusted service function chain controller**: A fake SFC controller can send malicious SFC classification rules to the SDN controller to cause classification errors on the classifier. Further attacks may also be launched.

– **Security vulnerability on the classifier**: An attacker can try to exploit a security vulnerability on the physical host or virtual host that runs the classifier application to escalate its privileges and launch further attacks, e.g., SFC flow rule tampering.

– **Denial of service attack**: An attacker can send a large number of flows to the classifier, for flow classifications at the same time, and cause an overload on the classifier.

– **Service function chain flow rule overflow**: An SDN controller, under malicious conditions, can be induced to send a large number of SFC flow rules to the classifier (e.g., the SDN controller is compromised by an attacker or the algorithm on the SDN controller is designed unreasonably.). These flow rules are translated by the classifier and applied as entries in an SDN flow table. Because an SDN flow table has finite resources to store these entries, receiving a great number of SFC flow rules can cause an overflow of entries in the SFC flow table on the classifier.

8.2.2.2 Security requirements

R-16: The classifier is required to authenticate the SDN controller.

R-17: The classifier is required to authenticate the administrator.

R-18: The classifier is required to authenticate the SFF.

R-19: The classifier is required to support log and audit functions.

R-20: The classifier is required to authorize access to sensitive data.

R-21: The classifier is required to protect the security of stored sensitive data, e.g., SFC flow rules, SFC encapsulation.

R-22: The classifier is recommended to provide key/certificate management functions to support authentication/authorization and data protection.

R-23: The physical host or virtual machine, which runs the classifier, is recommended to support the mechanism to detect and mitigate security vulnerabilities. Provision of hardening for the classifier software is required.

R-24: The classifier is recommended to support the mechanism to detect abnormal flows and mitigate overloads.

R-25: The classifier is recommended to set an appropriate expiration time for the SFC flow rules and to support a mechanism to detect maliciously sent SFC flow rules.

R-26: The SDN controller is required to authenticate and authorize the SFC controller. This requirement is the same as R-12 and R-13 of [ITU-T X.1038].

8.2.3 Security threats to and requirements of the service function forwarder

8.2.3.1 Security threats

Major specific security threats to the SFF are described as follows.

- **Spoofing**: An attacker can pose as a legitimate SDN controller, classifier, other SFF or SFs, to communicate with the SFF to try to launch spoofing attacks. For example, a spoofed SDN controller can create a fake flow rule to cause a false flow forward.
- **Untrusted service function chain controller**: A fake SFC controller can send malicious SFP forwarding rules to the SDN controller to cause a flow forwarding path error. Further attacks may also be launched.
- **Repudiation**: The administrators on the SFF can repudiate behaviours, such as SFC flow rule tampering.
- **Information disclosure**: An attacker can access sensitive data on the SFF (e.g., the SFC rule, administrator password) and cause other attacks. For example, an attacker can utilize the leaked administrator password to log on to the SFF and tamper with SFC flow rules.
- **Security vulnerability on the service function forwarder**: An attacker can try to exploit security vulnerabilities on the SFF and then escalate their privileges to further compromise the SFF.
- **Denial of service attack**: An attacker can send a large number of flows to the SFF at the same time and cause an overload on the SFF.
- **Service function chain rule overflow**: An SDN controller, under malicious conditions, can send a large number of SFC flow rules to the SFF (e.g., the SDN controller is compromised by an attacker or the algorithm on the SDN controller is designed unreasonably). These flow rules are translated from the SFP forwarding rules and applied as entries in an SDN flow table. Because an SDN flow table has finite resources to store these entries, receiving a great number of SFC flow rules can cause an overflow of entries in the SFC flow table on the SFF.

8.2.3.2 Security requirements

R-27: The SFF is required to authenticate the SDN controller.

R-28: The SFF is required to authenticate the administrator.

R-29: The SFF is required to authenticate the classifier.

R-30: The SFF is required to authenticate the SFs/SFC proxies.

R-31: The SFF is required to support the log function and audit function.

R-32: The SFF is required to authorize access to sensitive data, e.g., the SFC flow rules.

R-33: The SFF is recommended to protect the security of sensitive data stored on the SFF.

R-34: The SFF is recommended to provide key/certificate management functions to support authentication/authorization and data protection.

R-35: The SFF is recommended to support mechanisms to detect and mitigate security vulnerabilities.

R-36: The SFF is recommended to support mechanisms to detect abnormal flows and mitigate overloads.

R-37: The SFF is recommended to set an appropriate expiration time for SFC flow rules and support a mechanism to detect maliciously sent SFC flow rules.

R-38: The SDN controller is required to authenticate and authorize the SFC controller. This requirement is in the same as R-12 and R-13 of [ITU-T X.1038].

8.2.4 Security threats to and requirements of a service function

8.2.4.1 Security threats

An SF has the following major security threats.

– **Spoofing**: An attacker can pose as a legitimate SFC controller/SFF to communicate with the SF to try to launch attacks. For example, a spoofed SFC controller can request SF location/status to launch further attacks.

– **Repudiation**: The administrators on the SF can repudiate their behaviours, e.g., SF configuration modification.

– **Information disclosure**: An unauthorized attacker can access sensitive data on the SF, e.g., administrator passwords and configuration policies.

– **Security vulnerability on the service function**: An attacker can try to exploit security vulnerabilities on the physical host or virtual host that runs the SF application to escalate its privileges and launch further attacks, e.g., tampering with SF configuration and management policies.

– **Denial of service attack**: An attacker can send a large number of the flows to the SF at the same time and cause an overload on the SF.

8.2.4.2 Security requirements

R-39: The SF is required to authenticate the SFC controller.

R-40: The SF is required to authenticate the administrator.

R-41: The SF is required to authenticate the SFF.

R-42: The SF is required to authenticate the SFC proxy.

R-43: The SF is required to authenticate the management element.

R-44: The SF is required to support the log and audit function.

R-45: The SF is required to authorize access to sensitive data, e.g., SF configuration and management policies and parameters.

R-46: The SF is required to protect the security of stored sensitive data.

R-47: The SF is required to provide key/certificate management functions to support authentication/authorization and data protection.

R-48: The SF is recommended to support mechanisms to detect and mitigate security vulnerabilities.

R-49: The SF is recommended to support mechanisms to detect abnormal flows and mitigate overloads.

8.2.5 Security threats to and requirements of a service function chain proxy

8.2.5.1 Security threats

Since the SFC proxy acts as an SFF when it communicates with an SFC-unware SF and acts as an SF when it communicates with an SFF, the security threats to the SFF and the SF apply to the SFC proxy; therefore, for the security threats to the SFC proxy, see clause 8.2.3.1 and clause 8.2.4.1.

8.2.5.2 Security requirements

For the security requirements of the SFC proxy, refer to clause 8.2.3.2 and clause 8.2.4.2.

9 Threat analysis and requirements of interfaces

9.1 Interfaces

According to the general reference architecture of SDN-based service function chaining in Figure 6-1, there are the following interfaces:

– application-control interface;

– resource-control interface;

– intra-interfaces in the resource layer;

– management interface between the SFC controller and the SF/SFC proxy.

9.2 Security threats and requirements

9.2.1 Security threats

These interfaces are threatened by common security threats to an interface, i.e., an attacker can eavesdrop, tamper and replay flows that are sent or received by the interfaces.

9.2.2 Security requirements

These interfaces are required to support the following common security requirements for an interface.

R-50: Mutual authentication between entities on the interfaces.

R-51: The flows that are transformed or received by these interfaces are required to support confidentiality, integrity [ITU-T X.800] and anti-replay protection.

10 Security considerations of policy management

A policy determines the forwarding behaviour of flows in an SDN-based service function chaining. There are two types of policy: the SFC policy and the traditional SDN flow policy. These policies are transformed into flow rules. For example, SFC policies are transformed into SFC classification rules and SFP forwarding rules by the SFC controller; the SDN controller then transforms the received SFC classification rules and SFP forwarding rules into the flow rules.

The security threat to the policy management then is that an attacker can fake a policy, which can result in a conflict between the new flow rules and the stored rules. Thus, security requirements for the policy management involve resolving conflict between flow rules.

11 Countermeasures

This clause recommends countermeasures to meet security requirements corresponding to those in clause 8.2 and clause 9.2. Some security mechanisms refer to the security mechanisms in

[ITU-T X.1038]. These countermeasures can guide the design and implementation of security functions when developing SDN-based service function chaining.

11.1 Countermeasures for critical network elements

A list of recommended countermeasures to meet security requirements corresponding to those in clause 8 and clause 9 follows.

- Resolving policy conflict: To meet the requirement of R-01, first, the SFC controller needs to authenticate and authorize the SFC app to ensure the policy comes from a trusted app. Then, the SFC controller can use a mechanism similar to that in clause B.1 of [ITU-T X.1038] to set priorities for all policies.
- Authentication: Username/password-based authentication, authentication based on a pre-shared key (PSK) ([b-IETF RFC 4279], [b-IETF RFC 4306]) and certificate-based authentication ([b-IETF RFC 4306], [b-IETF RFC 5246] etc.) can be used to meet the authentication requirements, especially for security requirements R-02, R-03, R-04, R-05, R-06, R-16, R-17, R-18, R-26, R-27, R-28, R-29, R-30, R-38, R-39, R-40, R-41, R-42 and R-43. An appropriate countermeasure should be selected according to the deployment environment.
- Secure SDN controller: The SDN controller security mechanisms proposed in clause 8.2 of [ITU-T X.1038] can be used. These mechanisms are especially used to meet requirement R-07.
- Log and audit: The OS log function can be used and log server employed to store the security logs, operation logs etc. Logs can be used to audit. These mechanisms can meet log and audit requirements, especially for security requirements R-08, R-19, R-31 and R-44.
- Authorization: Access control list (ACL), role-based access control (RBAC) etc. can be used to meet authorization requirements, especially for security requirements R-09, R-10, R-20, R-26, R-32, R-38 and R-45.
- Data integrity: A hash-based message authentication code (HMAC) [b-IETF RFC 2104] or digital signature etc. can be used to protect data, e.g., SF status to meet the data integrity requirements, especially for security requirements R-11, R12, R-21, R-33 and R-46.
- Confidentiality: The advanced encryption standard (AES) can be used to meet confidentiality protection requirements, especially R-12, R-21, R-33 and R-46.
- Key/certificate management: The key management mechanism specified in [ITU-T X.800] and certificate management protocol defined in [b-IETF RFC 4210] can be applied. The requirements of R-13, R-22, R-34 and R-47 can be satisfied.
- Vulnerability detection and patch functions: Security tools to detect and patch these vulnerabilities can be used. This mechanism especially meets requirements R-14, R-23, R-35 and R-48.
- Anti-DoS: Limiting the number of requests, authenticating the requester, deploying anti-DoS devices etc. can be used together. This mechanism especially meets requirement R-15.
- Detection of abnormal flow and flow overload: An intrusion prevention system (IPS) can be deployed to detect abnormal flow. Protected entities can support the setting of a threshold value to limit processing flows. These mechanisms especially meet requirements R-24, R-36 and R-49.
- Protection of SFC flow rule against overflow: The protected entity (e.g., classifier, SFF) needs to authenticate the SDN controller, set an appropriate expiration time for the SFC flow rule and a threshold value for receiving the SFC flow rule. These mechanisms especially meet requirements R-25 and R-37.

11.2 Countermeasures for interface security

A list of recommended countermeasures to meet corresponding security requirements for interfaces follows.

– Mutual authentication on the interfaces: PSK-based authentication ([b-IETF RFC 4279], [b-IETF RFC 4306]), and certificate-based authentication ([b-IETF RFC 4306], [b-IETF RFC 5246] etc.) can be used by parties to authenticate each other on the interfaces. This mechanism meets requirement R-50.

– Confidentiality, integrity and anti-replay protection on the interfaces that meet requirement R-51.

– Application-control interface: Like the recommended countermeasures in clause 8.4 of [ITU-T X.1038], it is recommended that transport layer security (TLS) [b-IETF RFC 5246] or hypertext transfer protocol secure (HTTPS) protocols be implemented and deployed in the SFC application and the SFC controller, or SFC application and SDN controller to provide mutual authentication between the SFC application and the SFC controller, or between SFC application and SDN controller, as well as to provide data confidentiality, data integrity and anti-replay for data transportation over the application-control interface.

– Resource-control interface: Like the recommended countermeasures in clause 8.4 of [ITU-T X.1038], it is recommended that TLS [b-IETF RFC 5246] or Internet protocol security (IPSec) protocols ([b-IETF RFC 4301], [b-IETF RFC 4303], [b-IETF RFC 4835]) be implemented and deployed in the SDN controller and classifier/SFF/switch to provide mutual authentication between the SDN controller and classifier/SFF/switch, as well as to provide data confidentiality, data integrity and anti-replay for data transportation over the resource-control interface.

– Intra-interfaces in the resource layer: It is recommended that IPSec protocols ([b-IETF RFC 4301], [b-IETF RFC 4303], [b-IETF RFC 4835]) be implemented and deployed in the intra-interfaces, e.g., between classifier and SFF, SFF and SF/SFC proxy, to provide mutual authentication between the entities on the intra-interface, as well as providing data confidentiality, data integrity and anti-replay for data transportation over the intra-interface.

– Management interface between the SFC controller and the SF/SFC proxy: It is recommended that TLS [b-IETF RFC 5246] or IPSec protocols ([b-IETF RFC 4301], [b-IETF RFC 4303], [b-IETF RFC 4835]) be implemented and deployed in the SFC controller and SF/SFC proxy to provide mutual authentication between the SFC controller and SF/SFC proxy, as well as providing data confidentiality, data integrity and anti-replay for data transportation over the management interface.

11.3 Countermeasures for secure policy management

It is recommended that: 1) the SFC controller authenticates the SFC app, the SDN controller authenticates the SFC controller with PSK-based authentication ([b-IETF RFC 4279], [b-IETF RFC 4306]) and certificate-based authentication ([b-IETF RFC 4306], [b-IETF RFC 5246] etc.), respectively; 2) the SFC controller authorizes the SFC app, the SDN controller authorizes the SFC controller with an ACL, RBAC, etc.; 3) the SFC controller and SDN controller can also set the priority for all flow rules like the proposed mechanism in clause B.1 of [ITU-T X.1038].

Annex A

Classification table and service function path forwarding table

(This annex forms an integral part of this Recommendation.)

This annex specifies the classification table and the SFP forwarding table based on the OpenFlow flow table in [b-ONF TS-025] in order to reflect classification rules and SFP forwarding rules. In this way, service function chaining can be implemented based on SDN according to [ITU-T Y.3300].

Table A.1 shows the main components of a flow entry in a flow table specified in [b-ONF TS-025].

Table A.1 – Main components of a flow entry in an OpenFlow flow table

Match fields	Priority	Counters	Instructions	Timeouts	Cookie	Flags

Each flow table entry (see Table A.1) contains the following.

- Match fields: To match against packets. These consist of the ingress port and packet headers, and optionally other pipeline fields, e.g., metadata specified by a previous table.
- Priority: Matching precedence of the flow entry.
- Counters: Updated when packets are matched.
- Instructions: To modify the action set or pipeline processing.
- Timeouts: Maximum amount of time or idle time before flow is expired by the switch.
- Cookie: Opaque data value chosen by the controller. May be used by the controller to filter flow entries affected by flow statistics, flow modification and flow deletion requests. Not used when processing packets.
- Flags: flags alter the way flow entries are managed, e.g., the flag OFPFF_SEND_FLOW_REM triggers flow removed messages for that flow entry.

In order to specify the classification table and SFP forwarding table to reflect classification rules and SFP forwarding rules, two components of the flow table in Table A.1 (i.e., "match fields" and "instructions") will be extended. Moreover, some new components of the flow table will also be introduced. The other five components (i.e., "priority", "counters", "timeouts", "cookie" and "flags") will be applied to the classification table and SFP forwarding table without change.

A.1 Classification table

Compared to Table A.1, Table A.2 shows how to extend two components (i.e., "match fields" and "instructions") and how to introduce new components (i.e., "NSH" and "Next hop") in order to construct a classification table that reflects classification rules.

Table A.2 – extended components and new components of a flow entry in a classification table

Classification table of classifier													
Match fields						NSH				Next hop	Instruction		
											Apply action		Goto_table
Source IP. addr.	Destination IP addr.	Protocol type	In port	Application type	User Information	SPI	SI	Next protocol	Context headers	Next Hop	Update packet header	Update header match field	

A description of the extended components and new components in Table A.2 follows.

– Match fields: Referring to the "match fields" (i.e., to match against packets) defined in [b-ONF TS-025], with some extensions in this Recommendation as follows: match fields optionally includes other information about packet payload, e.g., application type and user information.

– NSH: A new component of a flow entry in a classification table to support the NSH is specified in [b-IETF RFC 8300].

– Next hop: A new component of a flow entry in the classification table to support the next hop in an SFP which is specified according to service path identifier (SPI) and service index (SI), defined in [b-IETF RFC 8300]. The next hop field is used for the underlying network protocol to establish a tunnel to transport the packet encapsulated by the NSH. For example, if the multi-protocol label switching (MPLS) protocol is used to transport the packet encapsulated by the NSH, more than one label-switched path (LSP) may be established for an SFP if the service functions are deployed in different data centres. An LSP is established between the current node and the next hop node.

– Instruction: Referring to the "instructions" (i.e., to modify the action set or pipeline processing) specified in [b-ONF TS-025], with some extensions in this Recommendation as follows: pushing the NSH, replacing the destination Internet protocol (IP) address with that of the next hop in order to support establishing an LSP if the protocol MPLS is used to transmit the packet encapsulated with the NSH.

  – If the MPLS protocol is used to transmit the packet encapsulated with the NSH, the label stack entry of MPLS specified in [b-IETF RFC 3032] may be extended as follows in order to indicate that the transmitted packet is encapsulated with the NSH. In this way, SFC implementation based on SDN can be done.

  – The label stack entry defined in [b-IETF RFC3032] is described in Table A.3.

Table A.3 – Label stack entry defined in [b-IETF RFC3032]

0	19	22 23	31
Label: 20 bits	Exp: 3 bits	S: 1bit	TTL: 8 bits

– Label: label value, 20 bits

– Exp: experimental use, 3 bits, reserved for experimental use

– S: bottom of stack, 1 bit

– TTL: time to live, 8 bits

– In order to support NSH encapsulation, 1 bit of Exp as in Table A.4 is used to indicate that its MPLS is used for transport encapsulation

Table A.4 – Indication that multi-protocol label switching is used for transport encapsulation

0	19 20	22 23	31
Label: 20 bits	N: 1bit Exp: 3 bits	S: 1bit	TTL: 8 bits

N: To indicate that MPLS is used for transport encapsulation for the NSH, 1 bit

A.2 Service function path forwarding table

Compared with Table A.1, Table A.5 shows how to extend two components (i.e., "match fields" and "instructions") and how to introduce new components (i.e., "Next hop") in order to construct an SFP forwarding table that reflects SFP forwarding rules.

Table A.5 – Extended and new components of a flow entry in a service function path forwarding table

<table>
<tr><th colspan="8">SFP table of SFF</th></tr>
<tr><th colspan="2">Match fields</th><th rowspan="4">Next hop</th><th colspan="5">Instruction</th></tr>
<tr><td rowspan="3">SPI</td><td rowspan="3">SI</td><td colspan="4">Apply action</td><td rowspan="3">Goto_table</td></tr>
<tr><td rowspan="2">Update Packet header</td><td>Update header match field</td><td rowspan="2" colspan="2">Out port</td></tr>
<tr><td>Destination IP addr.</td></tr>
</table>

Legend:

IP: Internet protocol; SFF: service function forwarder; SFP: service function path;
SI: service index; SPI: service path identifier

The extended components and new components in Table A.5 are described as follows.

– Match fields: To extend Table A.1 with some new components of a flow entry to support matching the SPI and SI of the NSH defined in [b-IETF RFC 8300].

– Next hop: A new component of a flow entry in the SFP forwarding table to support the next hop in an SFP, which is specified according to SPI and SI, defined in [b-IETF RFC 8300]. The next hop field is used for the underlying network protocol to establish a tunnel to transport the packet encapsulated by the NSH.

Instruction: referring to the "instructions" (i.e., to modify the action set or pipeline processing) defined in [b-ONF TS-025], with some extensions in this Recommendation as follows: popping NSH, replacing the destination IP address with the IP address of the next hop in order to support the establishment of an LSP, if the MPLS protocol is used to transmit the packet encapsulated with the NSH.

Appendix I

Service function chain architecture specified by other standards development organizations

(This appendix does not form an integral part of this Recommendation.)

Figure I.1 is taken from Figure 3 of [b-IETF RFC 7665].

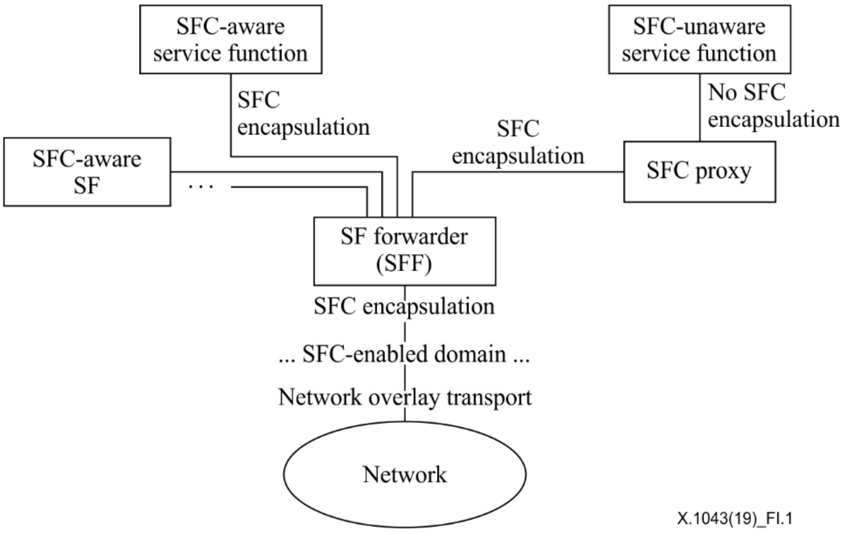

Figure I.1 – Service function chain architecture components after initial classification [b-IETF RFC 7665]

The architecture in Figure I.1 only describes the network elements in the data plane, i.e., SFF, SF and SFC proxy. The classifier is not included in Figure 6-1 because it shows architecture after initial classification.

Figure I.2 is taken from section 4 of [b-ONF TS-027].

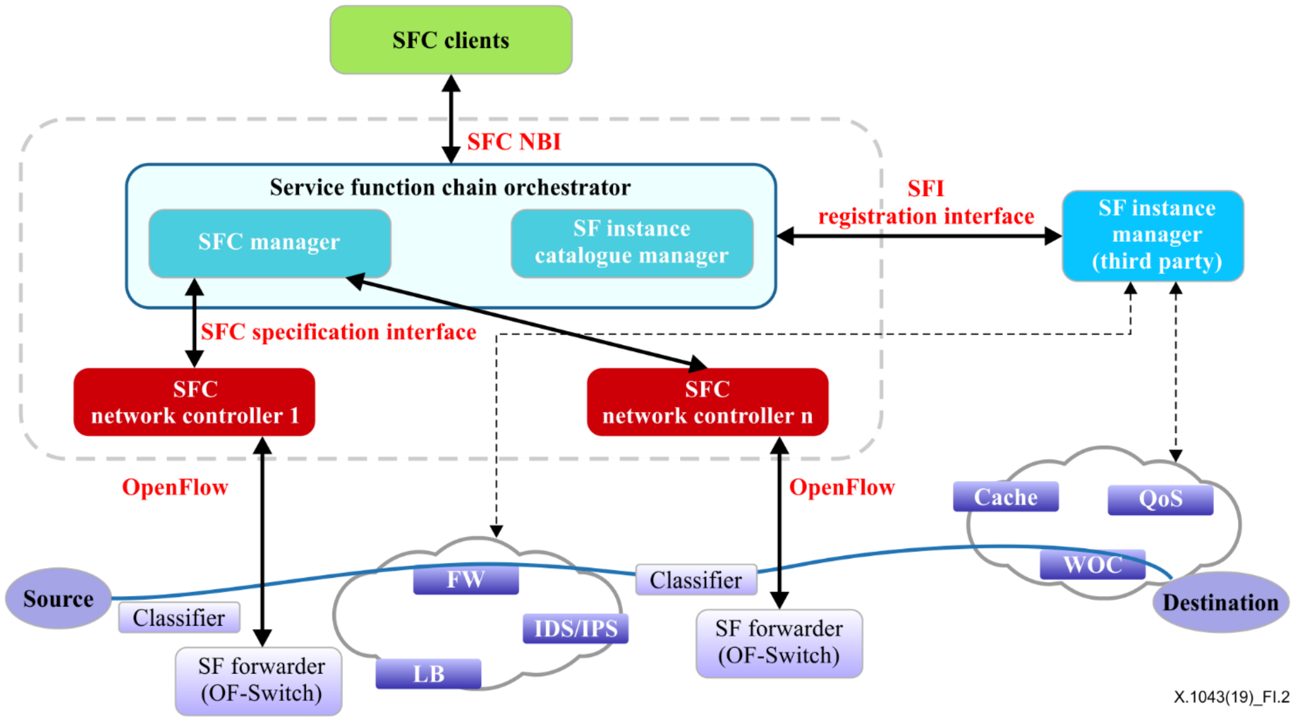

Figure I.2 – L4-L7 software-defined networking service function chain architecture [b-ONF TS-027]

The architecture in Figure I.2 is Open Networking Foundation (ONF) SDN-based and the intent is to build a common base for concrete northbound interface (NBI) specifications and OpenFlow extensions needed for SFC.

Bibliography

[b-IETF RFC 2104] IETF RFC 2104 (1997), HMAC: *Keyed-hashing for message authentication*.

[b-IETF RFC 3032] IETF RFC 3032 (2001), *MPLS label stack encoding*.

[b-IETF RFC 4210] IETF RFC 4210 (2005), *Internet X.509 public key infrastructure certificate management protocol (CMP)*.

[b-IETF RFC 4279] IETF RFC 4279 (2005), *Pre-shared key ciphersuites for transport layer security (TLS)*.

[b-IETF RFC 4301] IETF RFC 4301 (2005), *Security architecture for the Internet protocol*.

[b-IETF RFC 4303] IETF RFC 4303 (2005), *IP encapsulating security payload (ESP)*.

[b-IETF RFC 4306] IETF RFC 4306 (2005), *Internet key exchange (IKEv2) protocol*.

[b-IETF RFC 4835] IETF RFC 4835 (2007), *Cryptographic algorithm implementation requirements for encapsulating security payload (ESP) and authentication header (AH)*.

[b-IETF RFC 5246] IETF RFC 5246 (2008), *The transport layer security (TLS) protocol, version 1.2*.

[b-IETF RFC 7665] IETF RFC 7665 (2015), *Service function chaining (SFC) architecture*.

[b-IETF RFC 8300] IETF RFC 8300 (2018), *Network service header (NSH)*.

[b-ONF TS-025] ONF TS-025 (2015), *OpenFlow switch specification*, v.1.5.1. Available [viewed 2019-05-08] at: https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf

[b-ONF TS-027] ONF TS-027 (2015), *L4-L7 service function chaining solution architecture*, v.1.0. Available [viewed 2019-05-09] at: https://www.opennetworking.org/wp-content/uploads/2014/10/L4-L7_Service_Function_Chaining_Solution_Architecture.pdf

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2019