SFC Security Mimicry Defense
draft-li-ietf-sfc-security-mimicry-defense-00

Abstract

   With the increase of network threats, the Service Function Chain
   (SFC) is vulnerable to various attacks, and as a key component of the
   entire SFC, the security of the service function (SF) is more
   critical. This document proposes a mimic SF security architecture
   based on the dynamic heterogeneous redundancy model, which can
   effectively protect the normal execution function of SF in SFC. The
   security architecture adopts an active defense method to defend
   against network attacks, and it can effectively defend against most
   SF attacks.

Status of This Memo

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

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

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

   This Internet-Draft will expire on September 13, 2021.

Table of Contents

1. Introduction

   SF is the most critical part of the entire SFC. The data flow in the
   SFC domain must go through specific SFs to complete a specific
   function. If the SF is controlled by the hacker, the hacker can
   tamper with and discard the data by the hijacked SF, and even
   paralyze the SFF by DOS attacks. It can not only result in the
   failure of specified functioning when operating SF, but it may also
   cause the collapse of the entire SFC domain, which completely
   violates the original intention of SFC.

   This document describes a mimic security architecture based on
   Dynamic Heterogeneous Redundancy Model (DHRM), adopting a proactive
   defense method to deal with the SFC security problems mentioned
   above.

2. Terminology

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [1].

3. Security issues of SFC

3.1. Tamper with Network Service Header

   RFC 8300 [2] points out that the NSH (Network Service Header) is the
   SFC encapsulation referenced in RFC 7665 [3].NSH contains path
   identification information to implement service paths. Metadata
   information is also included which enables service functions to share
   initial and intermediate classification results with downstream
   service functions. NSH provides functions to monitor and
   troubleshoot the SFC.

   RFC 8300 [2] mentions three methods of protecting NSH headers: SFF /
   SF NSH Verification, Transport Security, NSH Variable Header-Based
   Integrity.

   However, the above three methods cannot solve the NSH tampering
   problem effectively. If a hacker cracks the encryption mode of NSH
   and obtained the permission successfully to change the header of a
   certain SF, the hacker can change the NSH information, and the data
   is therefore discarded when returns to SFF without finding the
   mapping relationship. While in practical applications, it is
   impossible for each SFF to belong to only a specific SFC. Thus, if a
   hacker obtains the mapping relationship in the SFF, the hacker can
   change the NSH information to make the packet taking another SFC
   path, and the control plane cannot detect it. By changing the
   metadata information, the hacker can even make all devices in the SFC
   domain reach a wrong consensus.

3.2. Attack Service Function

   If a hacker attacks the SF directly and successfully controls the SF,
   obtaining all the permissions of the SF, the hacker can manipulate
   the SF arbitrarily. SFC domain may leads to the following damage.

   a. The packets passing through the SF may be tampered with,
       discarded, or even diverted to the hacker server, thereby
       intercepting the effective information in the network traffic.
       For example, a hacker can tamper with the data fields of all data
       packets flowing through the SF.

   b. This will cause all the packets in the SFC domain that pass the
       SF to be finally sent to the destination host with incorrect
       information. Even if the destination host finds an error and
       discards the packet, it will cause the waste of bandwidth and
       normal SF resources.

c. The attacker can directly change the functions completed by the
    SF, which is difficult to detect. Assuming that the firewall
    function is to filter useful information, and due to the unknown
    information that the firewall need to filter in the control
    level, it seems that the firewall still completes the filtering
    function, actually, it is the useful packet which is filtered.
    Even if it notices a lot of information that should not be
    present in the SFC, it is not easy to locate which specific part
    has been attacked.

4. SFC Security Mimicry Defense

This document designs a mimic SF defense architecture based on a
Dynamic Heterogeneous Redundancy Model (DHRM). This framework
ensures the security of SF by setting up an SF executive pool
composed of M heterogeneous SFs, and the control plane manages the SF
executive pool. Assuming that the decision algorithm is perfectly
performed without any wrong outcome, the SF will be offline and
cleaned by the control plane once the execution result of the SF is
wrong, The following sub-chapter illustrates the architecture and
mimic defense process specifically.

4.1. Mimicry Defense

The mimic defense technology is an active defense strategy proposed
against the serious asymmetry of the defense cost and attack cost of
cyberspace. The redundancy and heterogeneity of the executive body
are introduced to change the unity and similarity of the system, and
the dynamic and randomness are adopted to change the stativity and
certainty of the system, and the non-similar redundancy space is used
to block or disrupt the network attack to meet the requirements of
controllable system security risks.

Specifically, the mimic security defense is composed of executive
bodies with equivalent functions but different composition structures
to form the core function. Based on the DHRM, randomness and
dynamicity are realized, and the heterogeneous executive bodies in
the heterogeneous executive pool are continuously changing, which
greatly improves the dynamics and uncertainty of functional
representation. It makes it difficult for a hacker to detect the
behavior and characteristics of the target system, increasing the
complexity of the system, and also makes it difficult for hackers to
exploit system vulnerabilities to make effective attacks.

The DHRM is the basic model of mimic defense. Heterogeneity is the
foundation of the dynamic heterogeneous redundancy architecture. It
is necessary to ensure that each heterogeneous executive has the
greatest difference in various characteristics and attributes. The

   more heterogeneous attributes a heterogeneous executor has, the more
   loopholes it can defend, and the higher the cost and price an
   attacker will have to pay for an attack. Dynamicity means that
   different sets of heterogeneous actuators can be generated in
   different periods to replace the currently compromised set of
   heterogeneous actuators. Dynamicity makes the system present
   different characteristics to the outside world in different periods.
   This brings uncertainty to the attacker's attack, which will further
   increase the difficulty of the attacker's attack. Redundancy means
   that multiple heterogeneous executors which have the same function
   process the same business request. The collaborative work between
   the heterogeneity and redundancy realizes the change of the single
   environment on which the attack depends, increasing the attacker's
   attack cost and difficulty, and improving the security and
   reliability of the system. This concept was first proposed in 2013
   and has been effectively applied in the field of cyberspace security,
   such as mimic routers, mimic web servers, mimic DNS, mimic workflow
   execution system, mimic blockchain, etc. In addition, some mimic
   network devices have been produced and tested successfully.

4.2. **SFC Mimicry Defense Architecture**

```
   +----------------------------------------------------------------------+
   |                       Control    Plane                               |
   +----------------------------------------------------------------------+
             |           |          |              |            |
             |           | +===================+  |+====================+
             |           | |   Mimetic Control  |  ||   Mimetic Control   |
             |           | |       Plugin       |  ||       Plugin        |
             |           | |+--------+ +--------+|  || +--------+ +--------+|
             |           | ||Schedule| |Decision||  || |Schedule| |Decision||
             |           | || Module | | Module ||  || | Module | | Module ||
             |           | |+--------+ +--------+|  || +--------+ +--------+|
             |           | +===================+  |+====================+
             |           |    |      |              |   |            |
             |           |    |      |              |   |            |
             |           |    | +-------------+     |   |   +--------------+
             |           |    | |Heterogeneous|     |   |   |Heterogeneous |
             |           |    | | SF Executive|     |   |   | SF Executive |
             |           |    | |    Pool     |     |   |   |    Pool      |
             |           |    | +-------------+     |   |   +--------------+
             |           |    |      |              |   |        |
             |           |    |      |              |   |        |
   +-----+ +----------+  +-----------+        +----------------+  +------+
   |Users|-|Classifier|--|    SFF1   | ......|      SFFn       |--|Target|
   +-----+ +----------+  +-----------+        +----------------+  +------+
                 Figure 1: SFC Mimicry Defense Architecture
```
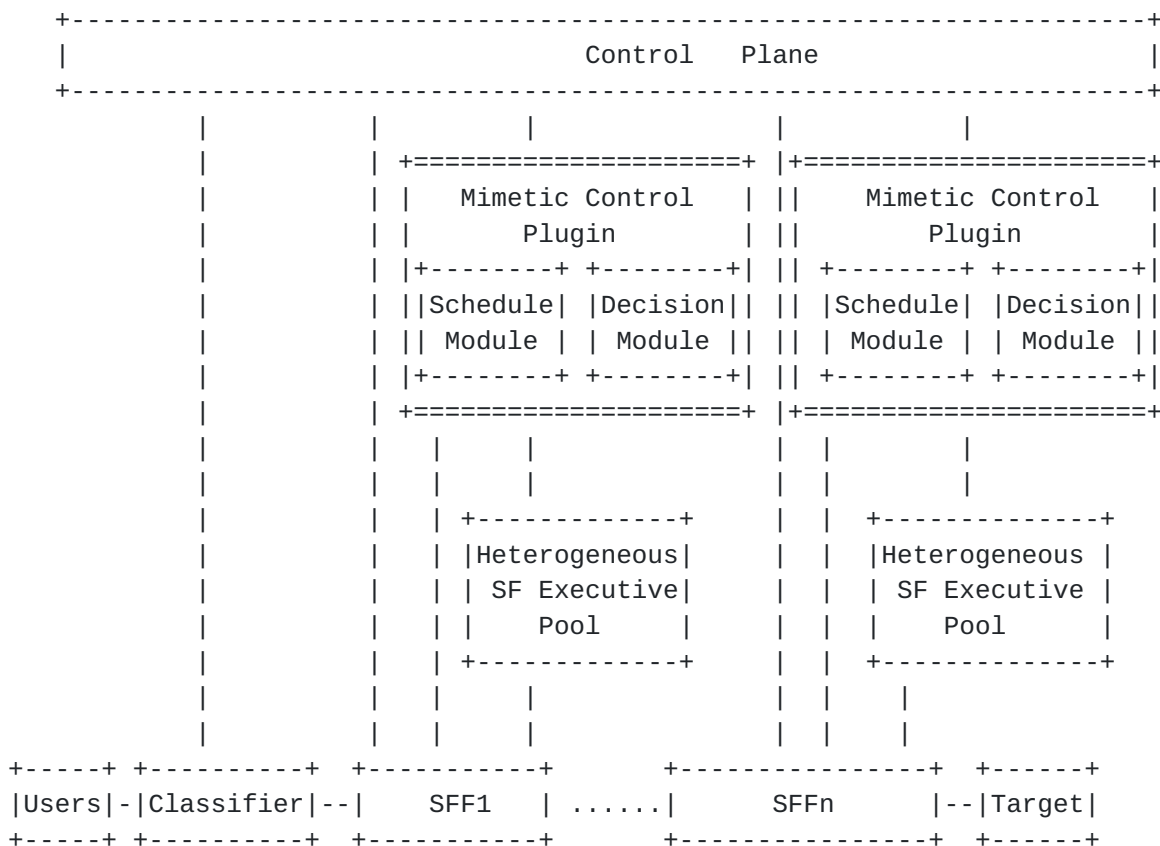
The detailed system architecture is shown in Figure 1. For a better
understanding, it is assumed that the system has a single SFC
(including the host, SFF, SF, and independent servers with other
modules, each SF connected to an SFF) and a single control plane.
The system architecture contains 05 components as follows:

Control plane defines the SFC in the network. The control plane
installs the flow rules in the SFF, and delivers the classification
strategy to the classifier based on the information uploaded by the
classifier. According to the judgment result, the abnormal SF
executor is cleaned and be offline, and the next SFF in the SFC is
notified that data is coming when the data passes through the SFF.

Mimic control plug-in: composed of 02 modules.

a.   Scheduling module: accept the running instruction of the control
     plane, select N SF executives from the SF executive pool to
     activate by using the scheduling algorithm.

b.   Judgment module: Judge the execution result of the SF executive
     body, and randomly select one from the correct results and send
     it to the SFF. If an abnormal SF executor is judged, the
     abnormal information will be transmitted to the control plane.

SF executive body pool: it consists of M heterogeneous SF executive
bodies, and N of them are activated by the scheduling module every
time when data passes.

SFF: forward the data according to the NSH information, copy the
packet into N copies and send the information to activate the
scheduling module.

Classifier: send the packet information to the control plane, and add
the corresponding NSH to the packet according to the classification
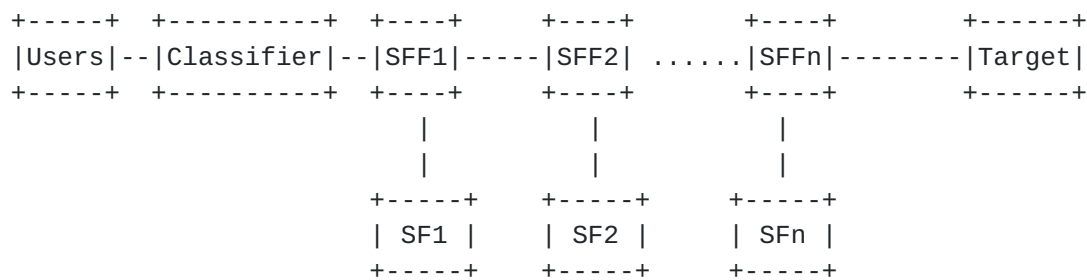strategy issued by the control plane.

4.3. Data Flow

```
+-----+  +----------+  +----+     +----+         +----+          +------+
|Users|--|Classifier|--|SFF1|-----|SFF2| ......|SFFn|--------|Target|
+-----+  +----------+  +----+     +----+         +----+          +------+
                          |          |              |
                          |          |              |
                       +-----+    +-----+       +-----+
                       | SF1 |    | SF2 |       | SFn |
                       +-----+    +-----+       +-----+
                           Figure 2: SFC Use Case
```

Take the SFC in Figure 2 as an example, we assume that the packet
path is the source host-classifier-SFF1-SF1-SFF1-SFF2-SF2 ...- target
host. The data first reaches the classifier before entering the SFC
domain, and the classifier sends the data information to the control
plane. The controller sends a classification strategy to the
classifier based on the uploaded information, then the classifier
adds NSH to the data packet according to the strategy and forwards
the data packet to SFF1 based on the header information. When the
data arrives at SFF1, we need to judge whether the data packet needs
to go through SF1 according to the NSH of the data packet.

When the packet reaches SFF1 for the first time, it should be
forwarded to SF. Therefore, SFF1 first sends a running instruction
to the scheduling module. After the scheduling module receives the
instruction, it uses a scheduling algorithm to select N activations
from the SF executive pool. Then, the information for activating the
SF actuator is forwarded to SFF1. SFF1 copies the packet into N
copies and sends the data to N SF actuators according to the
information provided by the scheduling module. Each SF performs its
designated network function and sends the execution result to the
decision module. After the judgment module accepts the execution
result, it judges whether there is an abnormal SF executive body by
using a judgment algorithm. If there is an abnormal execution body,
the abnormal information is sent to the control plane, and the
abnormal SF will be cleaned and offline by the control plane, then
the decision module selects one from the correct results and sends it
to SFF1. If there is no abnormal execution body, it will directly
select one from the correct results and send it to SFF1. SFF1 judges
whether to forward to SF1 according to the NSH of the returned
packet.

In this case, the packet should be forwarded to SFF2. After the
packet reaches SFF2, the same judgment and operation are performed
until it leaves the SFC domain.

4.4. Analysis of Attack and Defense

The SF executive pool is the foundation of the security of this
framework. By configuring M heterogeneous SF execution bodies to
form an SF execution body pool and using a scheduling algorithm to
select N activation methods from the M SF execution bodies to improve
the heterogeneity and dynamicity of the architecture, which increases
the difficulty of the hacker's attacks. Assuming that the packet
path is the source host-classifier-SFF1-SF1-SFF1-SFF2-SF2 ...- target
host. For a traditional SFC, if a hacker hijacks SF1, it can
manipulate SF1 to tamper with NSH so that traffic cannot reach SFF2.
Furthermore, if the attacker intercepts the mapping table of SFF2,
the hacker can tamper with the NSH header to make the traffic go

through a completely different SFC link. Any device in the SFC
domain, including the controller, will not find any abnormalities. A
hacker can also directly tamper with the service functions of SF, for
instance, the original SF performs load-balancing function. The
hacker can tamper with all the data to the same link, which may
paralyze the entire link.

If the security framework of this document is adopted, first, during
the SF attack detection phase, the hacker will find that there are M
SFs with the same function but different operating environments and
hardware devices mounted under the same SFF, which has caused certain
difficulty for the attack. If the hacker successfully controls one
SF executive in the M SF executive pool under SFF2. Before the data
is forwarded from the control module to the SF executive body, the
scheduling module uses a scheduling algorithm to activate N of the M
SF executive bodies. When M and N differ greatly, the probability of
selecting an abnormal SF execution body by random scheduling is
extremely low. It is further assumed that even if an abnormal
executor is selected, no matter whether the attacker tampers with the
packet or data function in the controlled SF executor, the decision
module can judge the exception when it judges the N execution
results, it can select the correct result and send abnormal
information to the control module for offline processing. Unless the
hacker successfully attacks more than half of the SF execution
bodies, the attack can be completely successful. While development
vendors and hardware equipment are different in different SFs
operating environment, which will cause great difficulty to the
hacker.

In another case, after the packet arrives at SFF1, it is not
forwarded to SF to complete the specified function but directly
forwarded to SFF2. When the packet arrives at SFF2, SFF2 will infer
from the NSH's SPI and SI information that the packet do not pass the
specified function in the previous SFF, and then report an error to
the control plane.

5. Acknowledgements

Thanks to Lei Rui and Chen Zebin for the first draft.

6. References

6.1. Normative References

[1]   Bradner, S., "Key words for use in RFCs to Indicate
            Requirement Levels", BCP 14, RFC 2119, March 1997,
            <http://xml.resource.org/public/rfc/html/rfc2119.html>.

   [2]   Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed.,
              "Network Service Header (NSH)", RFC 8300,
              DOI 10.17487/RFC8300, January 2018,
              <https://www.rfc-editor.org/info/rfc8300>.

6.2. Informative References

   [3]   Halpern, J., Ed. and C. Pignataro, Ed., "Service Function
              Chaining (SFC) Architecture", RFC 7665,
              DOI 10.17487/RFC7665, October 2015,
              <https://www.rfc-editor.org/info/rfc7665>.

Authors' Addresses

   Chuanhuang Li
   Zhejiang Gongshang University
   18 Xuezheng Str., Xiasha University Town
   Hangzhou 310018
   P.R.China

   Phone: +86 571 28877723
   Email: chuanhuang_li@zjsu.edu.cn

   Zhongyun Tang
   Zhejiang Gongshang University
   18 Xuezheng Str., Xiasha University Town
   Hangzhou 310018
   P.R.China

   Phone: +86 571 28877723
   Email: tangzy@zjsu.edu.cn

   Chao Chen
   Zhejiang Gongshang University
   18 Xuezheng Str., Xiasha University Town
   Hangzhou 310018
   P.R.China

   Phone: +86 571 28877723
   Email: eckio_491@zjgsu.edu.cn