

# 知 SecPath F1070 端口映射需要不停的更换映射公网端口才能访问服务成功问题

NAT 马雷勇 2019-11-22 发表

## 组网及说明

简化组网

公网-----F1070-----内网

## 问题描述

开始使用8080端口映射访问正常，过一段时间无法访问，后修改8089端口后测试访问又正常，过一段时间后又无法访问，如此反复，需要不停的更换映射的公网端口保证访问正常。

## 过程分析

1、原8080端口无法访问时但是telnet8080端口是正常的，有反应——说明内网端口服务开放正常，且公网没有封端口；

2、收集会话如下

```
<GZXJ-SecPath-F1070>display session table ipv4 destination-port 8088 verbose
```

Slot 1:

Initiator:

Source IP/port: x.x.164.158 /61812

Destination IP/port: 111.85.x.x/8088

DS-Lite tunnel peer: -

VPN instance/VLAN ID/Inline ID: -/-/

Protocol: TCP(6)

Inbound interface: Reth12

Source security zone: Untrust

Responder:

Source IP/port: x.x.32.99/8080

Destination IP/port: x.x.164.158/61812

DS-Lite tunnel peer: -

VPN instance/VLAN ID/Inline ID: -/-/

Protocol: TCP(6)

Inbound interface: Reth3

Source security zone: Trust

**State: TCP\_ESTABLISHED**

//tcp三次握手建立完成

Application: GENERAL\_TCP

Rule ID: 0

Rule name:

Start time: 2019-11-17 14:06:16 TTL: 3598s

Initiator->Responder: 2 packets 92 bytes

Responder->Initiator: 1 packets 52 bytes

Initiator:

Source IP/port: x.x.164.158/61813

Destination IP/port: 111.85.x.x/8088

DS-Lite tunnel peer: -

VPN instance/VLAN ID/Inline ID: -/-/

Protocol: TCP(6)

Inbound interface: Reth12

Source security zone: Untrust

Responder:

Source IP/port: x.x.32.99/8080

Destination IP/port: x.x.164.158/61813

DS-Lite tunnel peer: -

VPN instance/VLAN ID/Inline ID: -/-/

Protocol: TCP(6)

Inbound interface: Reth3

Source security zone: Trust

**State: TCP\_CLOSE**

//http close关闭

Application: HTTP

Rule ID: 0

Rule name:

Start time: 2019-11-17 14:06:16 TTL: 0s

Initiator->Responder: 4 packets 619 bytes

Responder->Initiator: 5 packets 6052 bytes

Total sessions found: 2

Slot 2:

Total sessions found: 0

3、F1070是R9333 D032版本，于是在防火墙上抓包如下，终端和服务器8080端口的包，如下

No.	Time	Source	Destination	Protocol	Length	Info
57	13.114902	164.158	76.164.158	TCP	58	62269 → 8080 [RST, ACK] Seq=2 Ack=1 Win=32768 Len=0
58	13.117619	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
59	13.117877	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1461 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
60	13.136689	164.158	76.164.158	TCP	58	62268 → 8080 [ACK] Seq=1 Ack=1 Win=65536 Len=0
61	13.143978	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
62	13.150760	76.164.158	164.158	TCP	58	8080 → 62270 [SYN, ACK] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
63	13.507817	76.164.158	164.158	TCP	58	8080 → 62270 [SYN, ACK] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
64	13.514954	164.158	76.164.158	HTTP	479	GET / HTTP/1.1
65	13.514250	164.158	76.164.158	TCP	58	62268 → 8080 [RST, ACK] Seq=2 Ack=1 Win=0 Len=0
66	13.515328	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1 Ack=4221 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
67	13.515576	76.164.158	164.158	TCP	58	8080 → 62268 [ACK] Seq=1461 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
68	13.551321	164.158	76.164.158	TCP	58	62270 → 8080 [ACK] Seq=1 Ack=1 Win=0 Len=0
69	13.560325	164.158	76.164.158	HTTP	479	GET / HTTP/1.1
70	13.560592	164.158	76.164.158	TCP	58	62270 → 8080 [RST, ACK] Seq=2 Ack=1 Win=32768 Len=0

公网用户为什么会发送RST重置报文？更换端口后就正常，比较奇怪，跟踪对应TCP数据流进行分析，

No.	Time	Source	Destination	Protocol	Length	Info
55	13.113940	164.158	76.164.158	TCP	58	62269 → 8080 [ACK] Seq=1 Ack=1 Win=65536 Len=0
56	13.114782	76.164.158	164.158	HTTP	498	GET / HTTP/1.1
57	13.114902	164.158	76.164.158	TCP	58	62268 → 8080 [RST, ACK] Seq=2 Ack=1 Win=0 Len=0
58	13.117619	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
59	13.117877	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1461 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
60	13.136689	164.158	76.164.158	TCP	58	62268 → 8080 [ACK] Seq=1 Ack=1 Win=65536 Len=0
61	13.143978	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
62	13.150760	76.164.158	164.158	TCP	58	8080 → 62270 [SYN, ACK] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
63	13.507817	76.164.158	164.158	TCP	58	8080 → 62270 [SYN, ACK] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
64	13.514954	164.158	76.164.158	HTTP	479	GET / HTTP/1.1
65	13.514250	164.158	76.164.158	TCP	58	62268 → 8080 [RST, ACK] Seq=2 Ack=1 Win=0 Len=0
66	13.515328	76.164.158	164.158	TCP	58	8080 → 62269 [ACK] Seq=1 Ack=4221 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
67	13.515576	76.164.158	164.158	TCP	58	8080 → 62268 [ACK] Seq=1461 Ack=441 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
68	13.551321	164.158	76.164.158	TCP	58	62270 → 8080 [ACK] Seq=1 Ack=1 Win=0 Len=0
69	13.560325	164.158	76.164.158	HTTP	479	GET / HTTP/1.1
70	13.560592	164.158	76.164.158	TCP	58	62270 → 8080 [RST, ACK] Seq=2 Ack=1 Win=32768 Len=0

发现从TTL跳数看，同源通目的IP

SYN (51) 、ACK (55) 、HTTP GET (56) 报文——TTL是103

和RST报文 (57) ——TTL是60

不是同一个源设备发的，尽管IP地址一样。

我司IPS设备做阻断时，伪装源IP发送RST，防火墙监测到RST报文后就会将TCP会话设置为Close状态，对应之前查看的会话信息。

此源IP来自公网，所以建议客户联系运营商排查对应攻击

### 解决方法

联系运营商排查对应攻击后问题解决