

# ASR5x00中SSL流应用的P2P插件分类和检测故障

## 目录

[简介](#)

[问题](#)

[故障排除](#)

[解决方案](#)

[配置示例](#)

[相关的思科支持社区讨论](#)

## 简介

本文档介绍一种特定场景，在这种场景中，用户在阻止其他用户流量的同时，使用Whatsapp、Snapchat等具有安全套接字层(SSL)流的免费速率应用。此特定应用在思科聚合服务路由器(ASR)5x00系列上运行。SSL是一种计算机网络协议，用于管理服务器身份验证、客户端身份验证以及服务器与客户端之间的加密通信。

## 问题

要检测任何应用，您需要一些初始数据包进行分析。这两个矛盾的要求尽可能的得到满足。

a)检测必须发生在第一个数据包本身

b)检测准确率必须为100%

如果您尝试满足要求(a)并标记第一个数据包中的所有应用（这实际上不可能），则对检测准确性的要求(b)会受到影响。为了使检测准确性良好，您需要更多数据包来分析许多应用（在第一个数据包中检测到应用的应用和流）。对于同一应用，您可能能够在第一个数据包中标记某些流量，而同一应用的其他流量需要更多数据包进行分析。

因此，如果任何应用在阻止任何其他流量时被免费评级，则可能会发生应用的初始数据包未被检测到，因为它没有传输足够的信息。在基于SSL流的应用的特定情况下，协议使用客户端 — hello数据包中存在的server-name-indication字段或SSL证书中存在的公用名进行标记。由于server-name是可选字段，因此并非始终存在。如此图所示，在Whatsapp SSL流中，在三次握手(TWH)后，客户端hello数据包由应用发送。显示无服务器名称指示(SNI)字段的PCAP跟踪。此外，还可以看到最终被丢弃的客户端hello数据包的多次重新传输。

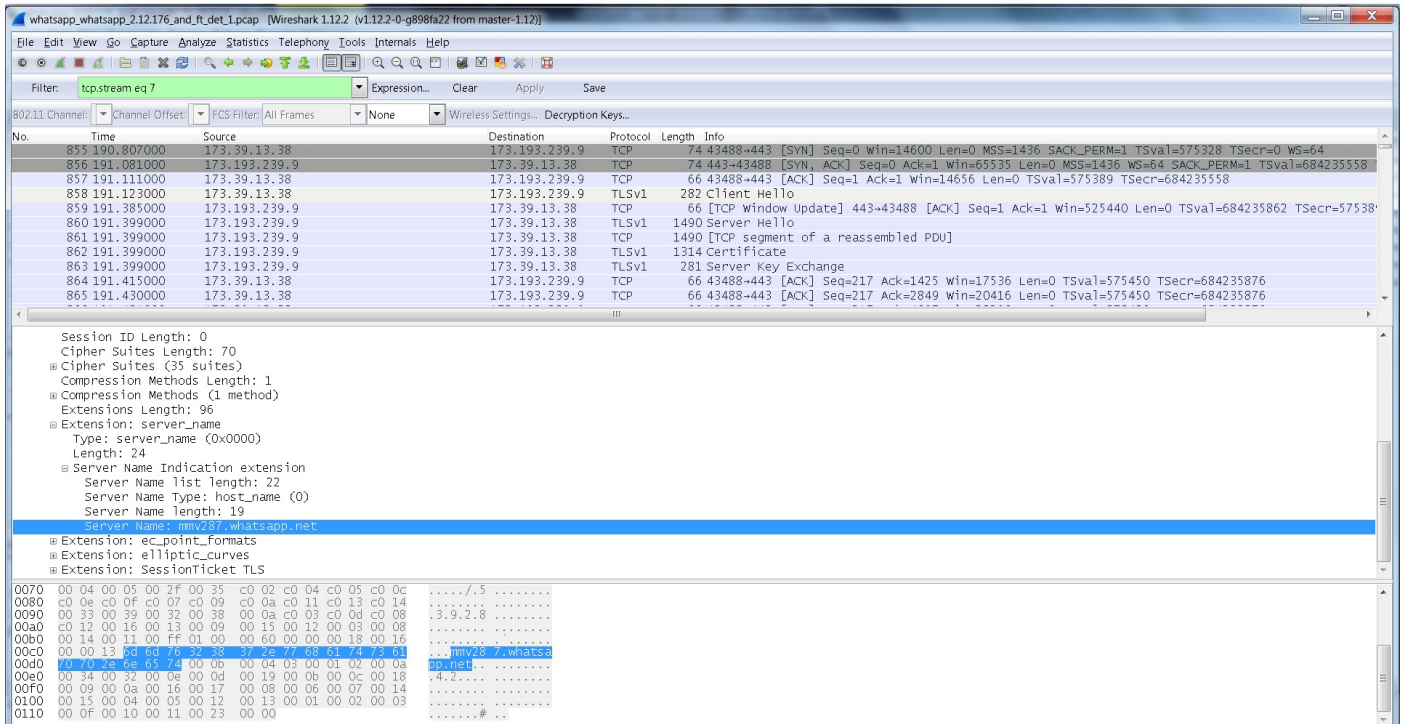
No.	Time	Source	SrcPort	Destination	DestPort	Protocol	Length	Tcp Stream	Info
5413	3621.067000	10.162.21.22	39780	82.129.130.230	443	TCP	74	259 39780-443	[SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 T
5414	3621.070000	82.129.130.230	443	10.162.21.22	39780	TCP	74	259 443-39780	[SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SA
5415	3621.369000	82.129.130.230	443	10.162.21.22	39780	TCP	74	259	[TCP Retransmission] 443-39780 [SYN, ACK] Seq=0 Ack=1 win=28
5416	3621.819000	10.162.21.22	39780	82.129.130.230	443	TCP	66	259 39780-443	[ACK] Seq=1 Ack=1 win=14608 Len=0 Tsval=6739606 TS
5417	3622.089000	10.162.21.22	39780	82.129.130.230	443	TCP	78	259	[TCP Dup ACK 5416#1] 39780-443 [ACK] Seq=1 Ack=1 win=14608 L
5418	3622.809000	10.162.21.22	39780	82.129.130.230	443	SSL	282	259	Client Hello
5426	3627.317000	10.162.21.22	39780	82.129.130.230	443	SSL	282	259	[TCP Retransmission] Client Hello
5428	3627.696000	82.129.130.230	443	10.162.21.22	39780	TCP	66	259 443-39780	[FIN, ACK] Seq=1 Ack=1 Win=29056 Len=0 Tsval=29202
5435	3629.202000	82.129.130.230	443	10.162.21.22	39780	TCP	66	259	[TCP Retransmission] 443-39780 [FIN, ACK] Seq=1 Ack=1 win=29
5442	3631.457000	82.129.130.230	443	10.162.21.22	39780	TCP	66	259	[TCP Retransmission] 443-39780 [FIN, ACK] Seq=1 Ack=1 win=29
5444	3635.969000	82.129.130.230	443	10.162.21.22	39780	TCP	66	259	[TCP Retransmission] 443-39780 [FIN, ACK] Seq=1 Ack=1 win=29
5449	3638.975000	10.162.21.22	39780	82.129.130.230	443	SSL	282	259	[TCP Retransmission] Client Hello
5453	3680.373000	10.162.21.22	39780	82.129.130.230	443	SSL	282	259	[TCP Retransmission] Client Hello
5465	3800.847000	10.162.21.22	39780	82.129.130.230	443	TCP	66	259 39780-443	[FIN, ACK] Seq=217 Ack=1 Win=14608 Len=0 Tsval=675
5469	3805.165000	10.162.21.22	39780	82.129.130.230	443	SSL	282	259	[TCP Retransmission] Client Hello
5470	3805.170000	82.129.130.230	443	10.162.21.22	39780	TCP	54	259 443-39780	[RST] Seq=1 win=0 Len=0
6057	4104.907000	82.129.130.230	443	10.162.21.22	39780	TCP	54	259 443-39780	[RST, ACK] Seq=2 Ack=218 win=0 Len=0

```

0000 0b 0b 0b 0b 0b 0a 0a 0a 0a 08 00 45 00 .....E.
0010 01 0c ea ed 00 40 06 59 df 0a a2 15 16 52 81 ...@.@.Y....R.
0020 82 e6 9b 64 01 bb a6 47 3f d3 b0 ad 61 01 80 18 ...d...G?.a..
0030 03 91 42 ea 00 00 01 01 08 0a 00 66 d6 a0 11 67 ..B.....f..g
0040 cd 90 16 03 01 00 d3 01 00 00 cf 03 01 55 bb 45 .....U.E
0050 8a 0e 68 93 17 13 a9 f8 3c 1a 9c a1 22 a8 1f 7f ..h.....<".
0060 59 c3 e8 7d 04 95 0e 2a 6c e3 23 42 82 20 8e 9f Y...}.*l.#B...
0070 b5 5c b9 ad 4c 92 d1 49 d3 0a 40 6b 6f 47 13 0b .\..L.I..@koG..
0080 d9 57 ff e6 1a 4c 20 a4 49 27 d0 57 5a 06 00 46 .w..L.I'.wz..F
0090 00 04 00 05 00 2f 00 35 c0 02 c0 04 c0 05 c0 0c ...../5.....
00a0 c0 0e c0 0f c0 07 c0 09 c0 0a c0 11 c0 13 c0 14 .....3.9.2.8.....
00b0 00 33 00 39 00 32 00 38 00 0a c0 03 c0 0d c0 08 .....@.....
00c0 c0 12 00 16 00 13 00 09 00 15 00 12 00 03 00 08 .....4.2.....
00d0 00 14 00 11 00 ff 01 00 00 40 00 0b 00 04 03 00 .....@.....
00e0 01 02 00 0a 00 34 00 32 00 0e 00 0d 00 19 00 0b .....4.2.....
00f0 00 0c 00 18 00 09 00 0a 00 16 00 17 00 08 00 06 .....#.....
0100 00 07 00 14 00 15 00 04 00 05 00 12 00 13 00 01 .....#.....
0110 00 02 00 03 00 0f 00 10 00 11

```

此外，如此图所示，它们是客户端呼叫数据包的十六进制字节，其中不存在用于标记Whatsapp的SNI字段。因此，客户端问候数据包不能标记为Whatsapp且不被检测。由于此数据包属于不同的额定组，因此会被丢弃，因此会看到客户端呼叫数据包的多次重新传输（请参阅帧号5449、5453、5469）。最后，连接终止。在pcap中可以看到多种此类流。这是无法执行任何有用活动（例如Whatsapp的映像上传）的原因。



## 故障排除

- capture monitor subscriber imsi XXXX with following options
    - 19 - User L3
    - X - PDU Hexdump
    - Verbosity level 5
- 这些命令提供分析器的应用统计信息。

```
# show act analyzer statistics name p2p application snapchat
# show act analyzer statistics name p2p application whatsapp
```

要检查插件版本，请执行以下操作：

```
#show plugin p2p
Wednesday July 29 22:12:07 SAST 2015
plugin p2p
  patch-directory /var/opt/lib
  base-directory /lib
  base-version 1.50.52055
  module priority 1 version 1.139.505
```

## 解决方案

为避免这种情况，您需要确保应用（例如whatsapp）之前的数据包被标记并必须通过。

使用此规则def:

```
ruledef ssl_clienthello
  tcp either-port = 443
  tcp payload-length >= 44
  tcp payload starts-with hex-signature 16-03
#exit
```

不能丢弃与上述规则定义匹配的任何数据包。此规则def的优先级必须刚好高于匹配此数据包并导致其被丢弃的默认规则def(ip-any ruledef)。

使用此配置时，只有与上述三条规则行匹配的数据包是空闲额定的。这些仅包括使用此规则def允许的SSL流中的初始握手数据包（如client-hello、server-hello），而SSL流中的所有其他数据包与此规则def不匹配。因此，如果有属于某个其他应用（您想要自由速率的什么应用除外）的SSLflow，则不能有任何有用的事务，因为仅允许SSL流的初始两到三个数据包使用此规则定义。

## 配置示例

建议的ruledef的优先级需要高于all-ip\_004\_012\_00016 ruledef(ip any-match = TRUE),

允许与whatsapp ruledef类似的流量的计费操作。

(sid\_040\_rg\_400\_rate\_99999/sid\_040\_rg\_400\_rate\_00032/sid\_040\_rg\_400\_rate\_00064，带rating-group 400和any rate)。

使用此配置时，客户端hello数据包将符合建议的规则定义，并且允许该数据包，而不是被重定向。以下是查看哪些应用规则的两个规则库：

```
rulebase mbc-internet-rs action priority 1087 dynamic-only ruledef WhatsApp_P2P_040_400_99999_All_internet charging-
action sid_040_rg_400_rate_99999 action priority 1088 dynamic-only ruledef WhatsApp_P2P_040_400_00064_All_internet
charging-action sid_040_rg_400_rate_00064 action priority 1089 dynamic-only ruledef
WhatsApp_P2P_040_400_00032_All_internet charging-action sid_040_rg_400_rate_00032 action priority [1090-9909]
dynamic-only ruledef ssl_clienthello charging-action sid_040_rg_400_rate99999/00064/00032 -->
Higher priority than all-ip ruledef and charging action with rating group 400
action priority 9910 dynamic-only ruledef all-ip_004_012_00016_MI_internet charging-action
sid_004_rg_012_rate_00016
action priority 9920 dynamic-only ruledef all-ip_004_012_00032_MI_internet charging-action
```

```
sid_004_rg_012_rate_00032
action priority 9930 dynamic-only ruledef all-ip_004_012_00064_MI_internet charging-action
sid_004_rg_012_rate_00064
```

```
rulebase mbc-iphone-rs
action priority 1206 dynamic-only ruledef WhatsApp_P2P_040_400_99999_All_iphone charging-action
sid_040_rg_400_rate_99999
action priority 1207 dynamic-only ruledef WhatsApp_P2P_040_400_00064_All_iphone charging-action
sid_040_rg_400_rate_00064
action priority 1208 dynamic-only ruledef WhatsApp_P2P_040_400_00032_All_iphone charging-action
sid_040_rg_400_rate_00032
```

```
action priority [1209-8999] dynamic-only ruledef ssl_clienthello charging-action
sid_040_rg_400_rate99999/00064/00032 --> Higher priority than all-ip ruledef and charging action
with rating group 400
```

```
action priority 9000 dynamic-only ruledef all-ip_015_150_00016_ALL_iphone charging-action
sid_015_rg_150_rate_00016
action priority 9010 dynamic-only ruledef all-ip_015_150_00032_ALL_iphone charging-action
sid_015_rg_150_rate_00032
action priority 9020 dynamic-only ruledef all-ip_015_150_00064_ALL_iphone charging-action
sid_015_rg_150_rate_00064
action priority 9030 dynamic-only ruledef all-ip_015_150_99999_ALL_iphone charging-action
sid_015_rg_150_rate_99999
```

```
charging-action sid_040_rg_400_rate_99999
content-id 400
service-identifier 40
billing-action egcdr
cca charging credit
exit
```

```
ruledef ssl_clienthello
tcp either-port = 443
tcp payload-length >= 44
tcp payload starts-with hex-signature 16-03
exit
```