Falla en la clasificación y detección del complemento P2P para aplicaciones con flujos SSL en ASR5x00

Contenido

Introducción

Este documento describe un escenario específico en el que el suscriptor utiliza aplicaciones de velocidad libre como Whatsapp, Snapchat, etc. con flujos de Secure Sockets Layer (SSL) mientras bloquea el tráfico de otros usuarios. Esta aplicación en particular se ejecuta en los routers de servicios agregados (ASR) de Cisco serie 5x00. SSL es un protocolo de redes informáticas que administra la autenticación de servidores, la autenticación de clientes y la comunicación cifrada entre servidores y clientes.

Problema

Para detectar cualquier aplicación, necesita algunos paquetes iniciales para el análisis. Estos dos requisitos contradictorios se cumplen en la mayor medida posible.

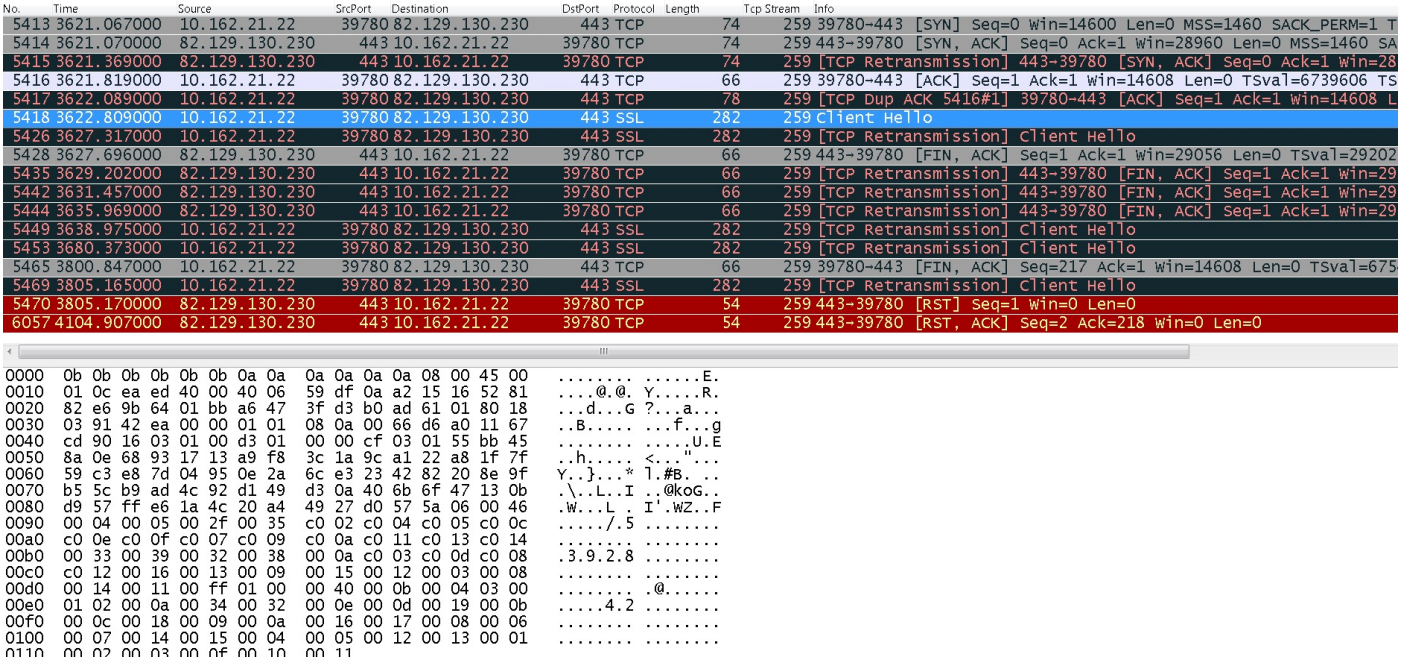
a) La detección debe ocurrir en el primer paquete mismo

b) La exactitud de la detección debe ser del 100%

Si intenta cumplir los requisitos (a) y marcar todas las aplicaciones en el primer paquete (que no es prácticamente posible), el requisito (b) de precisión de detección sufre. Para hacer que la precisión de detección sea buena, necesita más paquetes para analizar muchas aplicaciones (hay aplicaciones y flujos donde la aplicación se detecta en el primer paquete). En el caso de la misma aplicación, puede ocurrir que pueda marcar algunos flujos en el primer paquete mismo mientras que otros flujos de la misma aplicación necesitan más paquetes para el análisis.

Por lo tanto, si alguna de las aplicaciones está calificada de forma gratuita mientras se bloquea cualquier otro tráfico, puede ocurrir que el paquete inicial de la aplicación no se detecte, ya que no lleva suficiente información. En el caso particular de las aplicaciones basadas en flujos SSL, el protocolo se marca usando el campo server-name-signal presente en el paquete client-hello o el nombre común presente en el certificado SSL. Como el nombre de servidor es un campo opcional, no siempre está presente. Como se muestra en esta imagen, en un flujo SSL de Whatsapp, después de un intercambio de señales a tres (TWH) la aplicación envía el paquete hello del cliente. **Un seguimiento PCAP que no muestra ningún campo de indicación de nombre**

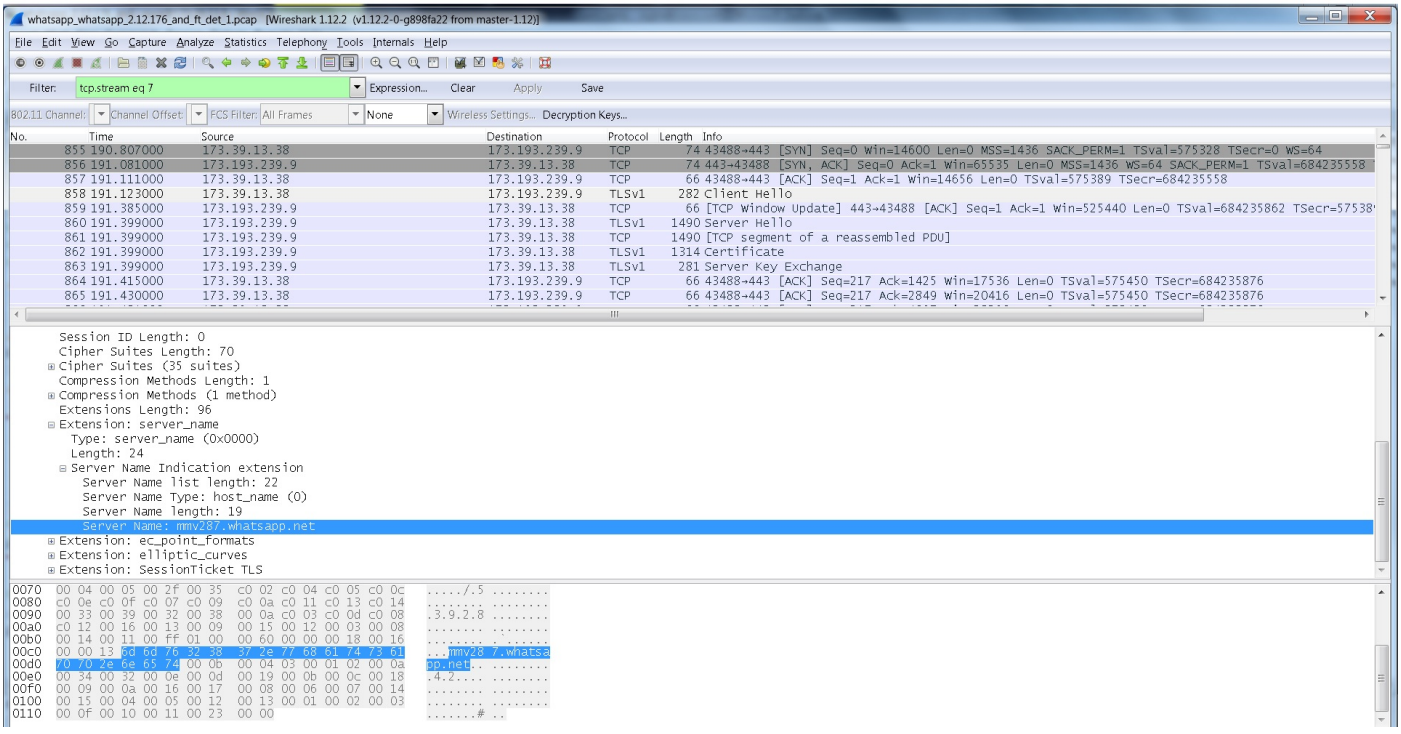
de servidor (SNI). También se ven varias retransmisiones de paquetes hello de cliente que finalmente se pierden.



No.	Time	Source	SrcPort	Destination	DstPort	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 0b 0a 0a  0a 0a 0a 0a 08 00 45 00   ........ ......E.
0010  01 0c ea ed 40 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   ........ ........
00b0  00 33 00 39 00 32 00 38  00 0a c0 03 c0 0d c0 08   .3.9.2.8 ........
00c0  c0 12 00 16 00 13 00 09  00 15 00 12 00 03 00 08   ........ ........
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
```

Además, como se muestra en esta imagen, sus son los hex-bytes para el paquete hello del cliente en el que el campo SNI, utilizado para marcar Whatsapp, no está presente. Por lo tanto, el paquete de saludo del cliente no se puede marcar como Whatsapp y pasa desapercibido. A medida que este paquete cae en un grupo de clasificación diferente, se pierde y, por lo tanto, se ven varias retransmisiones de paquetes de saludo de cliente (consulte la trama no 5449, 5453, 5469). Por último, la conexión finaliza. Varios de esos flujos se ven en el programa de control de la pobreza. Esta es la razón por la que no se puede realizar ninguna actividad útil, por ejemplo la carga de imágenes para
Whatsapp.



Troubleshoot

```
1. capture  monitor subscriber imsi XXXX with following options
19 - User L3
X - PDU Hexdump
Verbosity level 5
```

Estos comandos proporcionan las estadísticas del analizador para las aplicaciones.

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

Para verificar la versión del complemento:

```
#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
```

Solución

Para evitarlo, debe asegurarse de que los paquetes antes de que una aplicación (por ejemplo, whatsapp) se marquen y deben pasar.

Utilice esta regla:

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

No se debe descartar ningún paquete que coincida con la regla anterior. La prioridad de esta regla debe estar justo por encima de la regla predeterminada (ip-any ruledef) que coincidía con este paquete y causaba que se descartara.

Al utilizar esta configuración, sólo los paquetes que coinciden con las tres líneas de regla anteriores son de clasificación libre. Estos incluyen solamente los paquetes iniciales de intercambio de señales en el flujo SSL (como client-hello, server-hello) que se permiten usando esta regla, mientras que el resto de los paquetes en el flujo SSL no coinciden con esta regla. Por lo tanto, si hay un SSLflow que pertenece a alguna otra aplicación (que no sea whatsapp que desea liberar velocidad), no puede haber ninguna transacción útil, ya que sólo los dos o tres paquetes iniciales de un flujo SSL pueden utilizar esta regla.

Configuración de muestra:

La regla sugerida debe tener una prioridad más alta que la regla all-ip_004_012_00016 (ip any-match = TRUE) y

acción de carga que permite el tráfico similar a whatsapp ruledef.(sid_040_rg_400_rate_9999/sid_040_rg_400_rate_00032/ sid_040_rg_400_rate_00 064 con el grupo de clasificación 400 y cualquier tasa).

Con esta configuración, el paquete hello del cliente llega a la regla propuesta y se permite en lugar de ser redirigido. Estas son las dos bases de reglas donde se ven las reglas de whatsapp:

```
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
```