

Classificazione e rilevamento plug-in P2P non riusciti per l'applicazione con flussi SSL in ASR5x00

Sommario

[Introduzione](#)

[Problema](#)

[Risoluzione dei problemi](#)

[Soluzione](#)

[Esempio di configurazione](#)

[Discussioni correlate nella Cisco Support Community](#)

Introduzione

Questo documento descrive uno scenario specifico in cui il sottoscrittore usa applicazioni a velocità libera come Whatsapp, Snapchat ecc. con flussi SSL (Secure Sockets Layer) bloccando il traffico di altri utenti. Questa particolare applicazione viene eseguita su Cisco Aggregated Service Router (ASR) serie 5x00. SSL è un protocollo di rete per computer che gestisce l'autenticazione server, l'autenticazione client e la comunicazione crittografata tra server e client.

Problema

Per rilevare qualsiasi app, sono necessari alcuni pacchetti iniziali per l'analisi. Queste due esigenze contraddittorie sono soddisfatte nella misura massima possibile.

- a) Il rilevamento deve avvenire nel primo pacchetto
- b) La precisione del rilevamento deve essere del 100%

Se si tenta di soddisfare il requisito (a) e contrassegnare tutte le applicazioni nel primo pacchetto (che non è praticamente possibile), il requisito (b) sulla precisione del rilevamento subisce. Per rendere ottimale la precisione del rilevamento, sono necessari più pacchetti per analizzare molte applicazioni (ci sono app e flussi in cui l'applicazione viene rilevata nel primo pacchetto stesso). Nel caso della stessa app, può accadere che si sia in grado di contrassegnare alcuni flussi nel primo pacchetto stesso, mentre altri flussi della stessa app richiedono più pacchetti per l'analisi.

Quindi se una qualsiasi delle app è gratuita mentre blocca qualsiasi altro traffico, può succedere che il pacchetto iniziale dell'app non venga rilevato in quanto non contiene informazioni sufficienti. In particolare, nel caso di applicazioni basate su flussi SSL, il protocollo è contrassegnato utilizzando il campo di indicazione del nome del server presente nel pacchetto client-hello o il nome comune presente nel certificato SSL. Poiché il campo server-name è facoltativo, non è sempre presente. Come mostrato in questa immagine, in un flusso SSL Whatsapp, dopo Three-Way-Handshake (TWH) il pacchetto hello del client viene inviato dall'app. **Traccia PCAP priva di campo SNI (Server Name Indication). Sono inoltre presenti più ritrasmissioni di pacchetti hello del client che alla fine vengono scartati.**

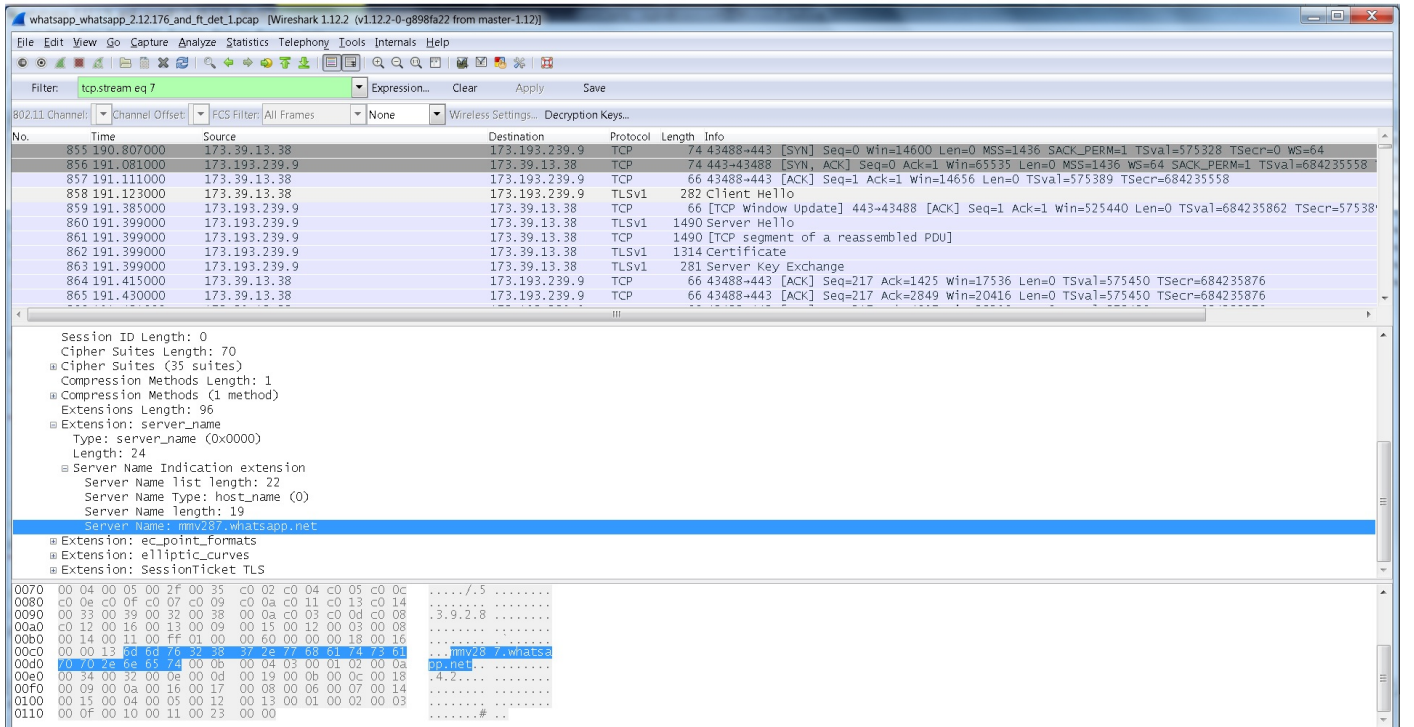
No.	Time	Source	SrcPort	Destination	DestPort	Protocol	Length	Tcp Stream	Info
5413	3621.067000	10.162.21.22	39780	82.129.130.230	443	TCP	74	259 39780-443	[SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 T
5414	3621.070000	82.129.130.230	443	10.162.21.22	39780	TCP	74	259 443-39780	[SYN, ACK] Seq=0 Ack=1 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 40 00 04 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.L.I'..@koG..
0080 d9 57 ff e6 1a 4c 20 a4 49 27 d0 57 5a 06 00 46 .w.L.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 04 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 .....#.....

```

Inoltre, come mostrato in questa immagine, sono i byte esadecimali del pacchetto client-hello in cui il campo SNI, usato per contrassegnare Whatsapp, non è presente. Pertanto, il pacchetto client-hello non può essere contrassegnato come Whatsapp e non viene rilevato. Poiché il pacchetto rientra in un gruppo di classificazione diverso, viene scartato e si verificano quindi più ritrasmissioni di un pacchetto client-hello (vedere i frame n. 5449, 5453, 5469). Infine, la connessione viene terminata. Diversi flussi di questo tipo sono visibili nella capsula. Per questo motivo non è possibile eseguire alcuna attività utile, ad esempio il caricamento di immagini per Whatsapp.



Risoluzione dei problemi

1. capture monitor subscriber imsi XXXX with following options

19 - User L3

X - PDU Hexdump
Verbosity level 5

Questi comandi forniscono lo stato dell'analizzatore per le applicazioni.

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

Per controllare la versione del plug-in:

```
#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
```

Soluzione

Per evitarlo, devi assicurarti che i pacchetti prima che un'app (ad esempio whatsapp) venga contrassegnata e debba passare.

Utilizza il seguente oggetto ruledef:

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

I pacchetti che corrispondono alla definizione di regola precedente non devono essere eliminati. La priorità di questo oggetto ruledef deve essere immediatamente superiore all'oggetto ruledef predefinito (ip-any ruledef) corrispondente al pacchetto e causarne l'eliminazione.

Utilizzando questa configurazione, solo i pacchetti che corrispondono alle tre righe sopra menzionate sono a tariffa libera. Tra questi sono inclusi solo i pacchetti di handshake iniziali nel flusso SSL (ad esempio client-hello, server-hello) consentiti tramite questo oggetto ruledef, mentre tutti gli altri pacchetti nel flusso SSL non corrispondono a questo oggetto ruledef. Pertanto, se esiste un SSLflow che appartiene a qualche altra app (diversa da whatsapp che si desidera rendere disponibile), non può esserci alcuna transazione utile, poiché solo i due o tre pacchetti iniziali di un SSL flow possono utilizzare questo ruledef.

Esempio di configurazione

Il ruledef suggerito deve avere una priorità più alta di all-ip_004_012_00016 ruledef (ip any-match = TRUE) e

operazione di caricamento che consente un traffico simile al traffico di whatsapp ruledef.(sid_040_rg_400_rate_9999/sid_040_rg_400_rate_00032/ sid_040_rg_400_rate_00064 con gruppo di classificazione 400 e qualsiasi tariffa).

Con questa configurazione, il pacchetto hello del client raggiunge il ruledef proposto ed è consentito anziché essere reindirizzato. Queste sono le due basi delle regole in cui vengono visualizzate le regole 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
```