Comprensione di eStreamer e risoluzione dei problemi di integrazione di eCore

Sommario

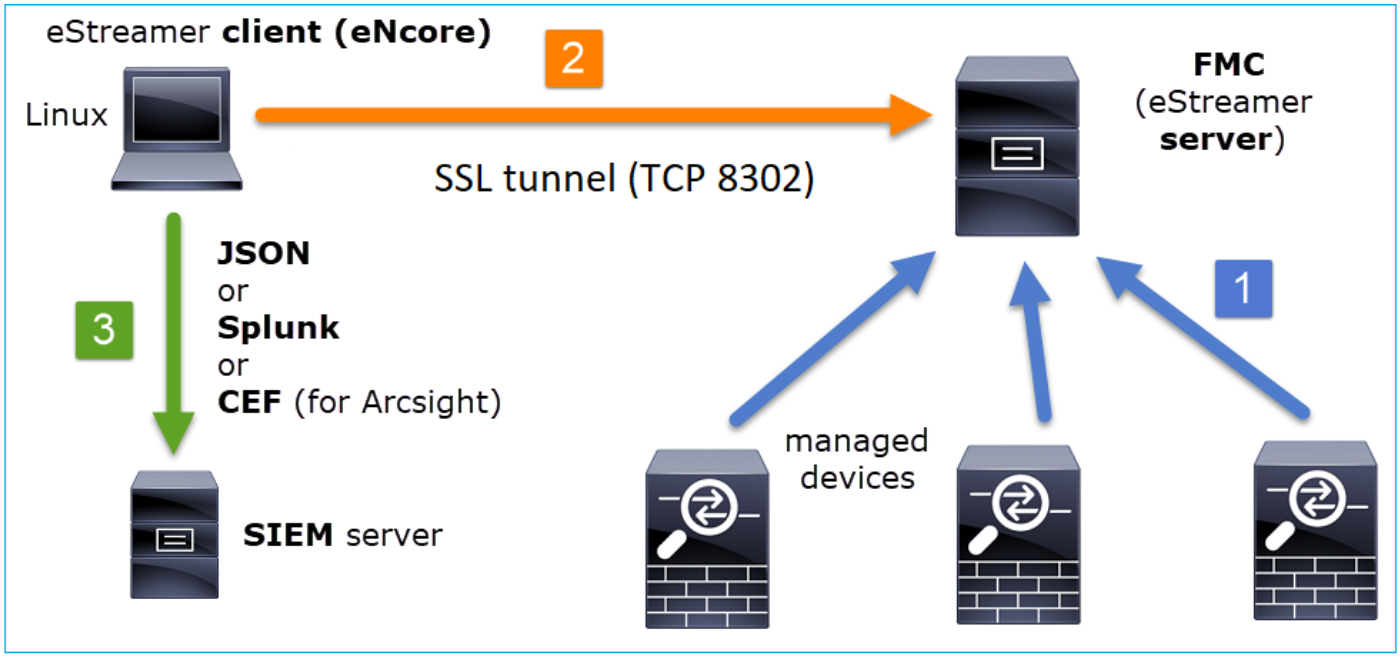
Introduzione

In questo documento viene descritto il client CLI di Ncore di Cisco Event Streamer (noto anche come eStreamer). In particolare, descrive l'operazione e fornisce informazioni sulla risoluzione dei problemi. Copre inoltre i problemi comuni rilevati dal Cisco Technical Assistance Center (TAC) e le domande frequenti (FAQ).

Contributo di David Torres Rivas, Mikis Zafeiroudis, Cisco TAC Engineers.

Panoramica

NeCore è un client multifunzione che richiede tutti i possibili eventi dal server eStreamer (FMC), analizza il contenuto binario ed emette eventi in vari formati per supportare altri strumenti SIEM (Security Information and Event Management).



Definizione connessione eStreamer

Il client (NeCore) avvia una connessione alla porta TCP FMC 8302 dove viene eseguito l'handshake SSL:

```
1: 11:34:02.901091 192.168.27.100.46538 > 10.48.26.49.8302: S 1607291631:1607291631(0) win 29200
<mss 1460,sackOK,timestamp 2350959 0,nop,wscale 10>
2: 11:34:02.902220 10.48.26.49.8302 > 192.168.27.100.46538: S 2529774236:2529774236(0) ack
1607291632 win 28960 <mss 1380,sackOK,timestamp 940036669 2350959,nop,wscale 7>
3: 11:34:02.902739 192.168.27.100.46538 > 10.48.26.49.8302: . ack 2529774237 win 29
<nop,nop,timestamp 2350959 940036669>
```

Il FMC accetta la connessione, esegue l'handshake SSL sulla stessa porta e verifica il nome comune (CN) del client:

```
Mar  2 11:45:06 FMC SF-IMS[22601]: [22601] Event Streamer:ConnectionHandler [INFO] Accepted IPv4
connection from 10.48.26.47:46538/tcp
Mar  2 11:45:06 FMC SF-IMS[22601]: [22601] Event Streamer:ConnectionHandler [INFO] Added
10.48.26.47 to host table
Mar  2 11:45:06 FMC SF-IMS[22601]: [22601] Event Streamer:ConnectionHandler [INFO] Added
10.48.26.47(23935) to host table
Mar  2 11:45:06 FMC SF-IMS[23935]: [23935] EventStreamer child(10.48.26.47):ConnectionHandler
[INFO] Resolved CN 10.48.26.47 to 10.48.26.47
Mar  2 11:45:06 FMC SF-IMS[23935]: [23935] EventStreamer child(10.48.26.47):ConnectionHandler
[INFO] Matched Certificate CN:10.48.26.47 to 10.48.26.47 (IPv4)
Mar  2 11:45:06 FMC SF-IMS[23935]: [23935] EventStreamer child(10.48.26.47):sfestreamer [INFO]
Got EVENT_STREAM_REQUEST length 8
Mar  2 11:45:06 FMC SF-IMS[23935]: [23935] EventStreamer child(10.48.26.47):sfestreamer [INFO]
Publishing service INFO total data size 48
Mar  2 11:45:06 FMC SF-IMS[23935]: [23935] EventStreamer child(10.48.26.47):sfestreamer [INFO]
Publishing service id:5001 - length size 8
Mar  2 11:45:06 FMC SF-IMS[23935]: [23935] EventStreamer child(10.48.26.47):sfestreamer [INFO]
Publishing service id:5000 - length size 8
Mar  2 11:45:06 FMC SF-IMS[23935]: [23935] EventStreamer child(10.48.26.47):sfestreamer [INFO]
Publishing service id:6667 - length size 8
```

Il client eStreamer controlla quindi la configurazione e il file dei segnalibri per determinare gli

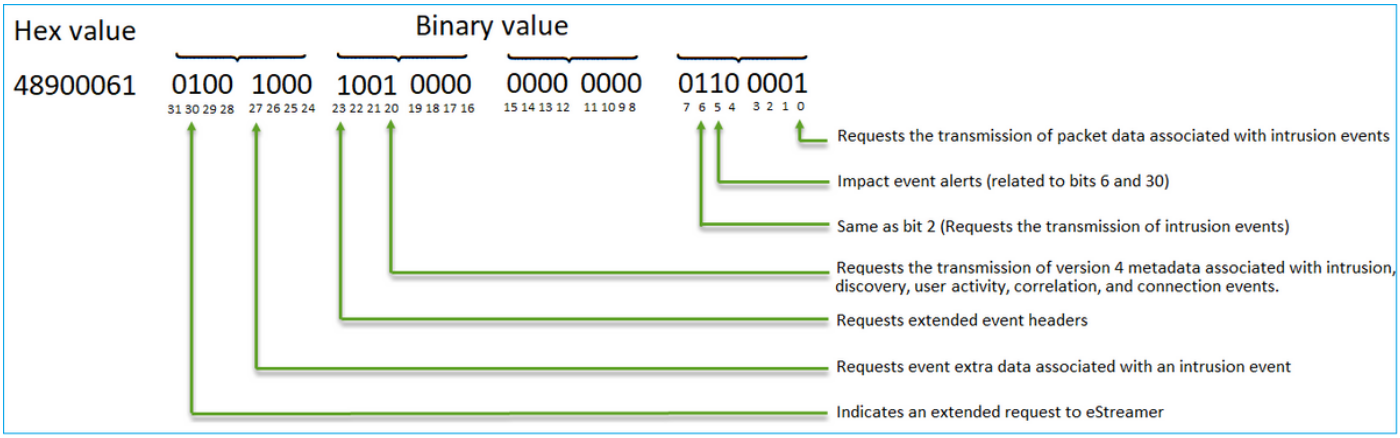
eventi da richiedere e l'ora di inizio:

```
2020-03-02 07:18:11,500 Connection   INFO     Connecting to 10.48.26.49:8302
2020-03-02 07:18:11,500 Connection   INFO     Using TLS v1.2
2020-03-02 07:18:11,500 Monitor      INFO     Starting Monitor.
2020-03-02 07:18:11,500 Monitor      INFO     Starting. 0 handled; average rate 0 ev/sec;
2020-03-02 07:18:11,501 Writer       INFO     Starting process.
2020-03-02 07:18:11,506 Transformer  INFO     Starting process.
2020-03-02 07:18:11,985 Bookmark     INFO     Bookmark file /root/eStreamer-eNcore/10.48.26.49-
8302_bookmark.dat does not exist.
2020-03-02 07:18:11,986 Settings     INFO     Timestamp: Start = 2 (Bookmark = 0)
2020-03-02 07:18:11,986 Receiver     INFO     EventStreamRequestMessage:
00010002000000080000000048900061
2020-03-02 07:18:11,986 SubscriberParser INFO     Starting process.
2020-03-02 07:18:11,996 Bookmark     INFO     Bookmark file /root/eStreamer-eNcore/10.48.26.49-
8302_bookmark.dat does not exist.
2020-03-02 07:18:11,996 Settings     INFO     Timestamp: Start = 2 (Bookmark = 0)
2020-03-02 07:18:11,997 Receiver     INFO     StreamingRequestMessage:
0001080100000038000001a0b0000003848900061000000000009000c000400150009001f000b003d000e00470004005b
0007006500060060f0002008300000000
```

EventStreamRequest può essere correlato in FMC:

```
Mar  2 12:29:16 FMC SF-IMS[6671]: [6671] EventStreamer child(10.48.26.47):sfestreamer [INFO]
EventStream Request (0x48900061): Since 0 w/ NS Events w/ NS 6.0 Events
w/ Packets w/ Extra IDS Event data w/ Metadata v4 w/ RUA 5.2 Events w/ Impact Alerts w/ 5.3
Impact Alerts w/ Impact Flags w/ RNA 5.3 Events w/ RNA 6.0 Flow w/ Policy 5.4 Events
w/ FireAMP 6.0 Events w/ Filelog 6.0 Events w/ Send archive timestamp w/ Send Detail Request
```

EventStreamRequest è la rappresentazione esadecimale dei flag di richiesta descritti in Flag di richiesta e deve essere convertita in formato binario per determinare se il client ha richiesto i dati richiesti. Questo è un esempio:



> **Nota:** Alcuni bit di flag potrebbero modificare le informazioni fornite se vengono avviate richieste estese.

In base ai bit della richiesta, il FMC invia i dati al client eStreamer.

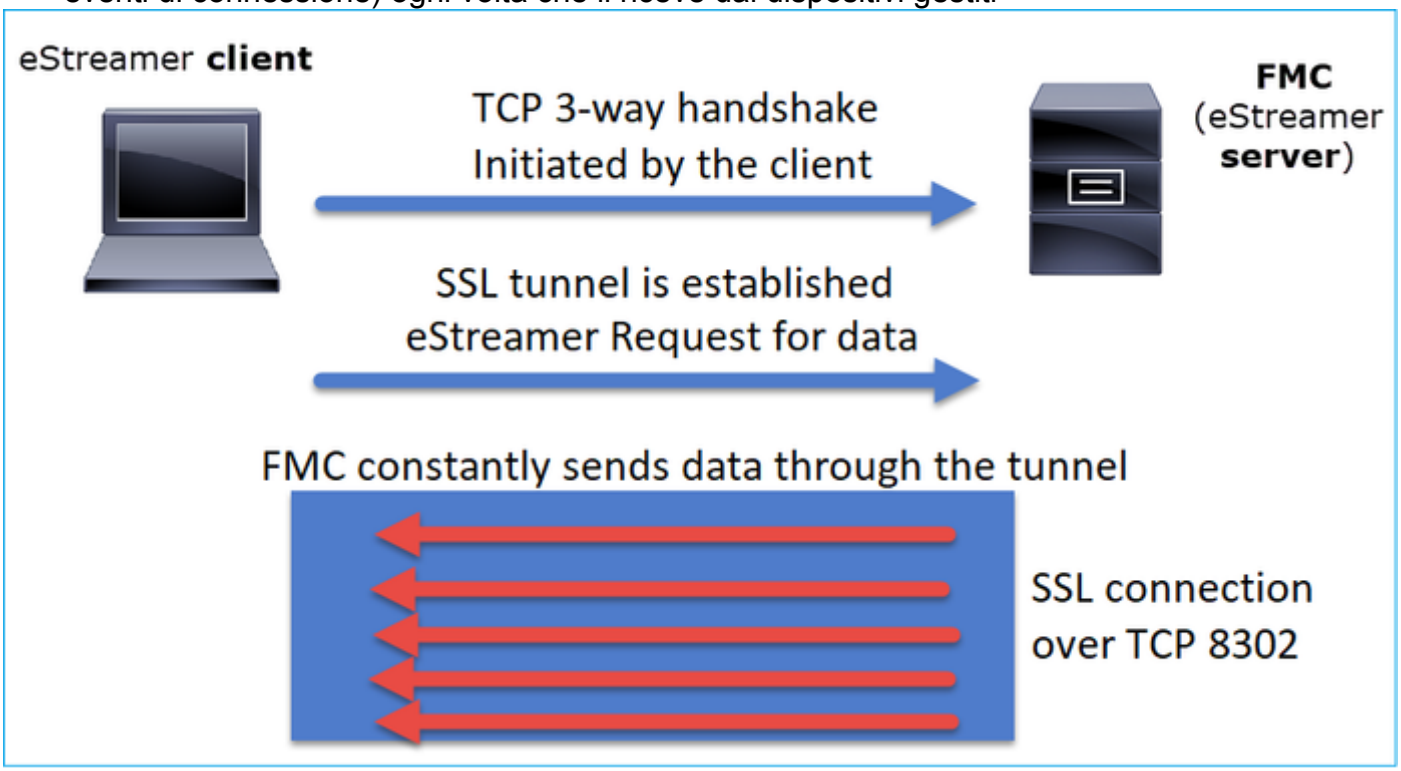
Chi avvia la connessione e il trasferimento dei dati di eStreamer?

Il client eStreamer. In particolare, il client stabilisce una connessione TCP (handshake a 3 vie), quindi viene eseguita una negoziazione SSL con l'autenticazione client (reciproca). Infine, attraverso il tunnel stabilito, il CCP invia i dati ogni volta che vi sono dati da inviare:

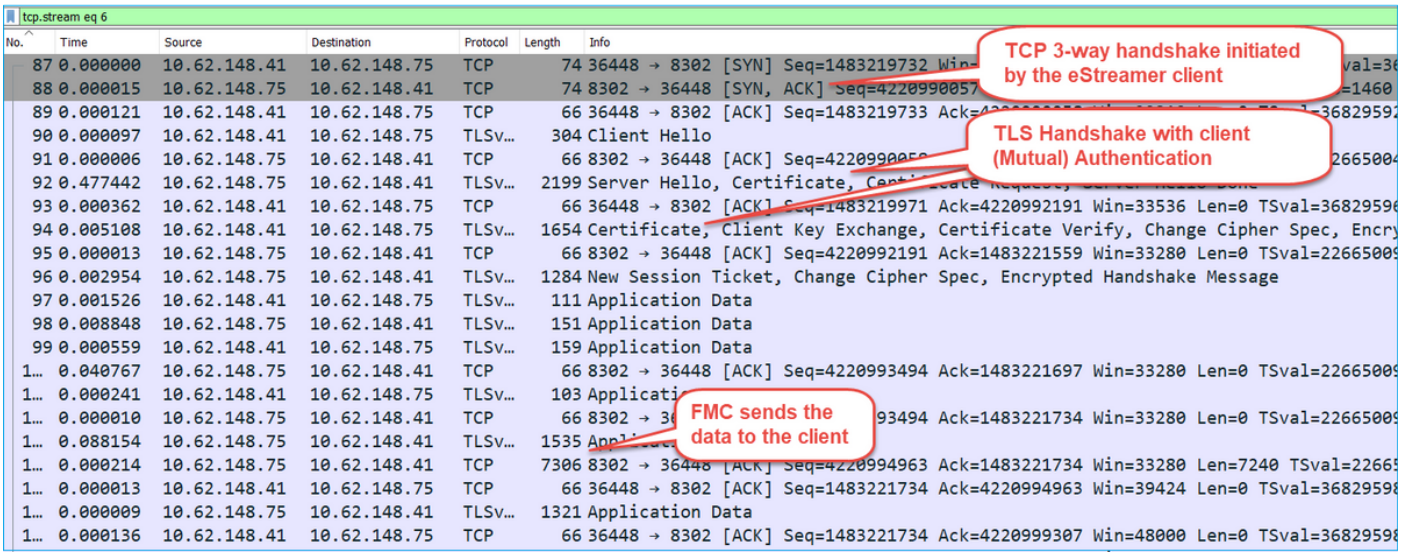
```
root@kali:~/eStreamer-eNcore# ./encore.sh foreground
2020-06-03 20:50:53,365 Monitor      INFO     Running. 100 handled; average rate 0.42 ev/sec;
2020-06-03 20:52:53,488 Monitor      INFO     Running. 100 handled; average rate 0.28 ev/sec;
2020-06-03 20:54:53,601 Monitor      INFO     Running. 100 handled; average rate 0.21 ev/sec;
2020-06-03 20:56:53,725 Monitor      INFO     Running. 100 handled; average rate 0.17 ev/sec;
```

In sintesi:

- Il client avvia il tunnel SSL per richiedere i dati (pull)
- Una volta stabilito il tunnel, il tunnel rimane attivo e la FMC comunica i dati (ad esempio, gli eventi di connessione) ogni volta che li riceve dai dispositivi gestiti



Nell'esempio, IP 10.62.148.41 è il client eStreamer (Ncore), mentre IP 10.62.148.75 è il FMC:



No.	Time	Source	Destination	Protocol	Length	Info
87	0.000000	10.62.148.41	10.62.148.75	TCP	74	36448 → 8302 [SYN] Seq=1483219732 Win=
88	0.000015	10.62.148.75	10.62.148.41	TCP	74	8302 → 36448 [SYN, ACK] Seq=4220990057
89	0.000121	10.62.148.41	10.62.148.75	TCP	66	36448 → 8302 [ACK] Seq=1483219733 Ack=
90	0.000097	10.62.148.41	10.62.148.75	TLSv…	304	Client Hello
91	0.000006	10.62.148.75	10.62.148.41	TCP	66	8302 → 36448 [ACK] Seq=4220990
92	0.477442	10.62.148.75	10.62.148.41	TLSv…	2199	Server Hello, Certificate,
93	0.000362	10.62.148.41	10.62.148.75	TCP	66	36448 → 8302 [ACK] Seq=1483219971 Ack=4220992191 Win=33536 Len=0 TSval=3682959
94	0.005108	10.62.148.41	10.62.148.75	TLSv…	1654	Certificate, Client Key Exchange, Certificate Verify, Change Cipher Spec, Encry
95	0.000013	10.62.148.75	10.62.148.41	TCP	66	8302 → 36448 [ACK] Seq=4220992191 Ack=1483221559 Win=33280 Len=0 TSval=2266500
96	0.002954	10.62.148.75	10.62.148.41	TLSv…	1284	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
97	0.001526	10.62.148.41	10.62.148.75	TLSv…	111	Application Data
98	0.008848	10.62.148.75	10.62.148.41	TLSv…	151	Application Data
99	0.000559	10.62.148.41	10.62.148.75	TLSv…	159	Application Data
1…	0.040767	10.62.148.75	10.62.148.41	TCP	66	8302 → 36448 [ACK] Seq=4220993494 Ack=1483221697 Win=33280 Len=0 TSval=2266500
1…	0.000241	10.62.148.41	10.62.148.75	TLSv…	103	Applicati
1…	0.000010	10.62.148.75	10.62.148.41	TCP	66	8302 → 3 93494 Ack=1483221734 Win=33280 Len=0 TSval=2266500
1…	0.088154	10.62.148.75	10.62.148.41	TLSv…	1535	App
1…	0.000214	10.62.148.75	10.62.148.41	TCP	7306	8302 → 36448 [ACK] Seq=4220994963 Ack=1483221734 Win=33280 Len=7240 TSval=2266
1…	0.000013	10.62.148.41	10.62.148.75	TCP	66	36448 → 8302 [ACK] Seq=1483221734 Ack=4220994963 Win=39424 Len=0 TSval=3682959
1…	0.000009	10.62.148.75	10.62.148.41	TLSv…	1321	Application Data
1…	0.000136	10.62.148.41	10.62.148.75	TCP	66	36448 → 8302 [ACK] Seq=1483221734 Ack=4220999307 Win=48000 Len=0 TSval=3682959

Configurazione

Per ulteriori informazioni sul client CLI di eStreamer, consultare la guida operativa CLI di eStreamer Ncore versione 3.5.

Per informazioni dettagliate sull'applicazione eStreamer e sui passaggi di configurazione di FMC, vedere la Guida all'integrazione di Event Streamer.

Ottimizzazione file estreamer.conf

In questa sezione vengono descritte le modifiche che è possibile o necessario apportare al file estreamer.conf per il corretto funzionamento della soluzione. Il file estreamer.conf si trova nella directory *path*/eStreamer-Ncore. Di seguito è riportato un esempio del contenuto del file:

```
root@kali:~/eStreamer-eNcore# cat estreamer.conf
{
    "connectTimeout": 10,
    "enabled": true,
    "handler": {
        "output@comment": "If you disable all outputters it behaves as a sink",
        "outputters": [
            {
                "adapter": "json",
                "enabled": true,
                "stream": {
                    "options": {
                        "maxLogs": 10000,
                        "rotate": true
                    },
                    "uri": "relfile:///data/json/encore.{0}.json"
                }
            }
        ],
        "records": {
            "connections": true,
            "core": true,
            "excl@comment": [
                "These records will be excluded regardless of above (overrides 'include')",
                "e.g. to exclude flow and IPS events use [ 71, 400 ]"
            ],
            "exclude": [],
            "inc@comment": "These records will be included regardless of above",
            "include": [],
            "intrusion": true,
            "metadata": true,
            "packets": true,
            "rna": true,
            "rua": true
        }
    },
    "logging": {
        "filepath": "estreamer.log",
        "format": "%(asctime)s %(name)-12s %(levelname)-8s %(message)s",
        "lev@comment": "Levels include FATAL, ERROR, WARNING, INFO, DEBUG, VERBOSE and TRACE",
        "level": "INFO",
        "stdOut": true
    },
    "monitor": {
        "bookmark": false,
        "handled": true,
        "period": 120,
        "subscribed": true,
        "velocity": false
    },
    "responseTimeout": 2,
```

```
    "star@comment": "0 for genesis, 1 for now, 2 for bookmark",
    "start": 2,
    "subscription": {
        "records": {
            "@comment": [
                "Just because we subscribe doesn't mean the server is sending. Nor does it
mean",
                "we are writing the records either. See handler.records[]"
            ],
            "archiveTimestamps": true,
            "eventExtraData": true,
            "extended": true,
            "impactEventAlerts": true,
            "intrusion": true,
            "metadata": true,
            "packetData": true
        },
        "servers": [
            {
                "host": "10.62.148.75",
                "pkcs12Filepath": "client.pkcs12",
                "port": 8302,
                "tls@comment": "Valid values are 1.0 and 1.2",
                "tlsVersion": 1.2
            }
        ]
    },
    "workerProcesses": 4
```

Sezione Sottoscrizione

Per modificare la richiesta Event Streamer verso il server (FMC), modificare la sezione delle sottoscrizioni eStreamer.conf. Ad esempio, quando si impostano le richieste estese su false, la richiesta EventStream viene modificata in FMC:

```
"subscription": {
        "records": {
            "@comment": [
                "Just because we subscribe doesn't mean the server is sending. Nor does it
mean",
                "we are writing the records either. See handler.records[]"
            ],
            "archiveTimestamps": true,
            "connection": true,
            "eventExtraData": true,
            "extended": false,
            "impactEventAlerts": true,
            "intrusion": true,
            "metadata": true,
            "packetData": true
        },
```

Con richieste estese = false:

```
Jun 3 13:48:24 firepower SF-IMS[16084]: [16084] EventStreamer child(10.48.26.47):sfestreamer
[INFO]
EventStream Request (0x08900061): Since 4294967295 w/ NS Events w/ Packets w/ Extra IDS Event
data w/
Metadata v4 w/ Impact Alerts w/ Impact Flags w/ Send archive timestamp
```

Con richieste estese = true:

```
Jun 3 13:50:52 firepower SF-IMS[17167]: [17167] EventStreamer child(10.48.26.47):sfestreamer
[INFO]
EventStream Request (0x48900061): Since 1590497346 w/ NS Events w/ NS 6.0 Events w/ Packets w/
Extra IDS Event data w/ Metadata
v4 w/ RUA 5.2 Events w/ Impact Alerts w/ 5.3 Impact Alerts w/ Impact Flags w/ RNA 5.3 Events w/
RNA 6.0 Flow w/ Policy 5.4 Events
v w/ FireAMP 6.0 Events w/ Filelog 6.0 Events w/ Send archive timestamp w/ Send Detail Request
```

Sezione Registrazione

Per abilitare i debug nella CLI di NeCore, modificare il file estreamer.conf e modificare il livello di log:

```
 "logging": {
        "filepath": "estreamer.log",
        "format": "%(asctime)s %(name)-12s %(levelname)-8s %(message)s",
        "lev@comment": "Levels include FATAL, ERROR, WARNING, INFO, DEBUG, VERBOSE and TRACE",
        "level": "DEBUG",
        "stdOut": true
    },
```

Sezione Monitor

Per visualizzare il numero di eventi al secondo elaborati e il segnalibro corrente, modificare la sezione Monitor in estreamer.conf:

```
  "monitor": {
    "bookmark": true,        #If true, adds date/timestamp (see above)
    "handled": true,          #Number of records processed
    "period": 120,            #How often (in seconds) monitor writes to the log
    "subscribed": true,       #Number of records received
    "velocity": false         #A measure of whether eNcore is keeping up (>=1 is good)
  },
```

Altre chiavi di primo livello pertinenti:

```
"connectTimeout": 10,     <- The number of seconds to wait for a response when establishing a
connection to the FMC.

"workerProcesses": 4,     <- The number of processes that eNcore spawns.
```

Questo valore può essere impostato da 2 a 12. Un numero maggiore di processi è destinato a migliorare le prestazioni, ma ogni processo comporta un costo comune. Il risultato è che le prestazioni ottimali si ottengono con la giusta combinazione di "numero di processi" con la capacità di elaborazione del computer host. Le migliori linee guida disponibili sono:

- Per 2 core: "workerProcesses": 4
- Per 4 o più core: "workerProcesses": 12

Risoluzione dei problemi

Per le procedure generiche di risoluzione dei problemi di eStreamer, consultare questo documento sulla risoluzione dei problemi tra FireSIGHT System e eStreamer Client (SIEM)

A scopo di test, è possibile abilitare Ncore come processo in primo piano e verificare la comunicazione con FMC

```
root@kali:~/eStreamer-eNcore# ./encore.sh foreground
2020-06-04 11:48:00,048 Controller   INFO     eNcore version: 3.5.4
2020-06-04 11:48:00,049 Controller   INFO     Python version: 2.7.13 (default, Jan 19 2017,
14:48:08) \n[GCC 6.3.0 20170118]
2020-06-04 11:48:00,051 Controller   INFO     Platform version: Linux-4.13.0-kali1-amd64-x86_64-
with-Kali-kali-rolling-kali-rolling
2020-06-04 11:48:00,052 Controller   INFO     Starting client (pid=12374).
2020-06-04 11:48:00,052 Controller   INFO     Sha256:
77ac7e72d0b96e0a4b9c1c4f9a16c2de0b2b5ccf2929dd2857cf94ed96b295e3
2020-06-04 11:48:00,052 Controller   INFO     Processes: 4
2020-06-04 11:48:00,053 Controller   INFO     Settings:
 …
2020-06-04 11:48:00,053 Diagnostics  INFO     Check certificate
2020-06-04 11:48:00,054 Diagnostics  INFO     Creating connection
2020-06-04 11:48:00,054 Connection   INFO     Connecting to 10.62.148.75:8302
2020-06-04 11:48:00,054 Connection   INFO     Using TLS v1.2
2020-06-04 11:48:00,136 Diagnostics  INFO     Creating request message
2020-06-04 11:48:00,137 Diagnostics  INFO     Request message=00010002000000008ffffffff48900061
2020-06-04 11:48:00,137 Diagnostics  INFO     Sending request message
2020-06-04 11:48:00,137 Diagnostics  INFO     Receiving response message
2020-06-04 11:48:00,229 Diagnostics  INFO     Response
message=KGRwMApTJ2xlbmd0aCcKcDEKSTQ4CnNTJ3ZlcnNpb24nCnAyCkkxCnNTJ2RhdGEnCnAzClMnXHgwMFx4MDBceDEz
XHg4OVx4MDBceDAwXHgwMFx4MDhceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgxM1x4ODhceDAw
XHgwMFx4MDBceDA4XHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MWFceDBiXHgwMFx4MDBceDAw
XHgwOFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwJwpwNApzUydtZXNzYWdlVHlwZScKcDUKSTIwNTEKcy4=
2020-06-04 11:48:00,229 Diagnostics  INFO     Streaming info response
2020-06-04 11:48:00,230 Diagnostics  INFO     Connection successful
2020-06-04 11:48:00,230 Monitor      INFO     Starting Monitor.
2020-06-04 11:48:00,236 Decorator    INFO     Starting process.
2020-06-04 11:48:00,236 Transformer  INFO     Starting process.
2020-06-04 11:48:00,237 Connection   INFO     Connecting to 10.62.148.75:8302
2020-06-04 11:48:00,237 Connection   INFO     Using TLS v1.2
2020-06-04 11:48:00,238 Writer       INFO     Starting process.
2020-06-04 11:48:00,639 Bookmark     INFO     Opening bookmark file /root/eStreamer-
eNcore/10.62.148.75-8302_bookmark.dat.
2020-06-04 11:48:00,640 Settings     INFO     Timestamp: Start = 2 (Bookmark = 1591210934)
2020-06-04 11:48:00,640 Receiver     INFO     EventStreamRequestMessage:
00010002000000085ed7f3b648900061
2020-06-04 11:48:00,640 SubscriberParser INFO     Starting process.
2020-06-04 11:48:00,640 Bookmark     INFO     Opening bookmark file /root/eStreamer-
eNcore/10.62.148.75-8302_bookmark.dat.
2020-06-04 11:48:00,646 Bookmark     INFO     Opening bookmark file /root/eStreamer-
eNcore/10.62.148.75-8302_bookmark.dat.
2020-06-04 11:48:00,646 Settings     INFO     Timestamp: Start = 2 (Bookmark = 1591210934)
2020-06-04 11:48:00,647 Receiver     INFO     StreamingRequestMessage:
000108010000003800001a0b0000003848900061 5ed7f3b60009000c000400150009001f000b003d000e00470004005b
0007006500060006f0002008300000000
2020-06-04 11:48:00,653 Monitor      INFO     Running. 0 handled; average rate 1.2 ev/sec;
```

Allo stesso tempo, in FMC è possibile visualizzare registri come questi quando il client Streamer di Ncore stabilisce la connessione. Il fuso orario del back-end FMC è sempre UTC:

```
root@FMC2000-2:~# tail -f /var/log/messages
Jun  4 09:48:00 FMC2000-2 SF-IMS[4135]: [4135] Event Streamer:ConnectionHandler [INFO] Accepted
```

IPv4 connection from 10.62.148.41:36528/tcp
Jun 4 09:48:00 FMC2000-2 SF-IMS[4135]: [4135] Event Streamer:ConnectionHandler [INFO] **Added 10.62.148.41(8512) to host table**
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):SFUtil [INFO] **Found IPv4 address 10.62.148.41 for ksec-sfvm-win7-3.cisco.com**
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] **Resolved CN ksec-sfvm-win7-3.cisco.com to 10.62.148.41**
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] **Matched Certificate CN:ksec-sfvm-win7-3.cisco.com to 10.62.148.41 (IPv4)**
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):sfestreamer [INFO] Got EVENT_STREAM_REQUEST length 8
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):sfestreamer [INFO] Publishing service INFO total data size 48
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):sfestreamer [INFO] Publishing service id:5001 - length size 8
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):sfestreamer [INFO] Publishing service id:5000 - length size 8
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):sfestreamer [INFO] Publishing service id:6667 - length size 8
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):sfestreamer [INFO] Got UEC_STREAM_REQUEST length 56
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] requested service [6667] timestamp [1591210934]
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 12, version 9
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 21, version 4
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 31, version 9
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 61, version 11
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 71, version 14
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 91, version 4
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 101, version 7
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 111, version 6
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] Got Event type 131, version 2
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):sfestreamer [INFO] **EventStream Request (0x48900061): Since 1591210934 w/ NS Events w/ NS 6.0 Events w/ Packets w/ Extra IDS Event data w/ Metadata v4 w/ RUA 5.2 Events w/ Impact Alerts w/ 5.3 Impact Alerts w/ Impact Flags w/ RNA 5.3 Events w/ RNA 6.0 Flow w/ Policy 5.4 Events w/ FireAMP 6.0 Events w/ Filelog 6.0 Events w/ Send archive timestamp w/ Send Detail Request**
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):ConnectionHandler [INFO] creating iterator for service [6667] prefix [unified2.] timestamp [1591210934]
Jun 4 09:48:00 FMC2000-2 SF-IMS[8512]: [8512] EventStreamer child(ksec-sfvm-win7-3.cisco.com):**Unified2Iterator [INFO] Opened /var/sf/archive/netmap_2/unified2.1591210800**
Jun 4 09:48:02 FMC2000-2 SF-IMS[4135]: [4135] Event Streamer:ConnectionHandler [INFO] Child with pid 8510 exited with status 5120
Jun 4 09:48:02 FMC2000-2 SF-IMS[4135]: [4135] Event Streamer:ConnectionHandler [INFO] Removed host entry for pid: 8510
Jun 4 09:48:04 FMC2000-2 SF-IMS[22296]: [25092] CloudAgent:url_license [INFO] Peer with active URLFiltering: 310f4c00-a415-11ea-bf5b-a2d6028849fe
Jun 4 09:48:04 FMC2000-2 SF-IMS[22296]: [25092] CloudAgent:url_license [INFO] Peer with active URLFiltering: d637b6f0-a414-11ea-ad97-cc17b6ea4c03
Jun 4 09:48:04 FMC2000-2 SF-IMS[22296]: [25092] CloudAgent:url_license [INFO] Peer with active URLFiltering: 873709b8-78b6-11ea-ae87-b82f93835447
Jun 4 09:48:04 FMC2000-2 SF-IMS[22296]: [25092] CloudAgent:url_license [INFO] Peer with active

```
URLFiltering: c7c0217c-78b6-11ea-a719-b7f0a277eb86
```

Elementi da raccogliere prima di contattare il Cisco Technical Assistance Center (TAC)

Si consiglia di raccogliere questi elementi prima di contattare Cisco TAC:

- Versione di eStreamer NeCore
- La versione di Python
- Versione del sistema operativo host
- Vedi gli eventi sul FMC? Condividi uno screenshot da eventi + configurazione FMC eStreamer
- Abilitare il debug sulla CLI di Netcore (come descritto nella sezione 'registrazione')
- Genera un file di risoluzione dei problemi da FMC
- Fornire questi file da NeCore:
  estreamer.conf
  estreamer.log

Problemi comuni

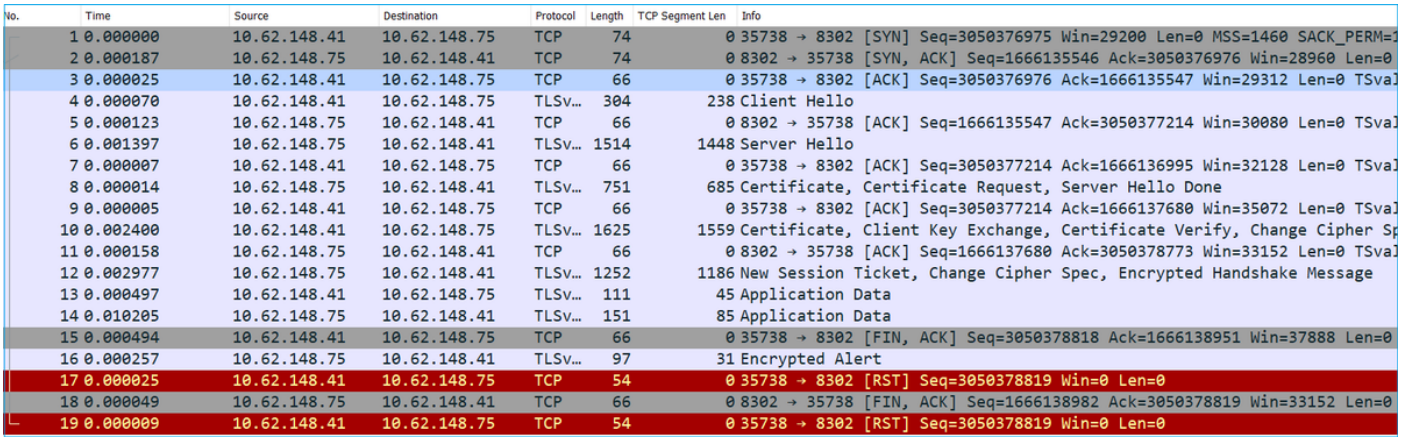
Nessuna connettività sulla porta TCP 8302

Telnet dal client eStreamer alla porta FMC 8302 e verificare che la connettività sia stabilita.

Èinoltre possibile utilizzare l'opzione di test di eCore per verificare la connettività:

```
root@kali:~/eStreamer-eNcore# ./encore.sh test
2020-05-28T16:02:56.931919 Diagnostics  INFO    Checking that configFilepath (estreamer.conf)
exists
2020-05-28 16:02:56,935 Diagnostics  INFO     Check certificate
2020-05-28 16:02:56,936 Diagnostics  INFO     Creating connection
2020-05-28 16:02:56,936 Connection   INFO     Connecting to 10.62.148.75:8302
2020-05-28 16:02:56,936 Connection   INFO     Using TLS v1.2
2020-05-28 16:02:56,946 Diagnostics  INFO     Creating request message
2020-05-28 16:02:56,946 Diagnostics  INFO     Request message=00010002000000008ffffffff48900061
2020-05-28 16:02:56,946 Diagnostics  INFO     Sending request message
2020-05-28 16:02:56,946 Diagnostics  INFO     Receiving response message
2020-05-28 16:02:56,957 Diagnostics  INFO     Response
message=KGRwMApTJ2xlbmd0aCcKcDEKSTQ4CnNTJ3ZlcnNpb24nCnAyCkkxCnNTJ2RhdGEnCnAzClMnXHgwMFx4MDBceDEz
XHg4OVx4MDBceDAwXHgwMFx4MDhceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgxM1x4ODhceDAw
XHgwMFx4MDBceDA4XHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MWFceDBiXHgwMFx4MDBceDAw
XHgwOFx4MDBceDAwXHgwMFx4MDBceDAwXHgwMFx4MDBceDAwJwpwNApzUydtZXNzYWdlVHlwZScKcDUKSTIwNTEKcy4=
2020-05-28 16:02:56,957 Diagnostics  INFO     Streaming info response
2020-05-28 16:02:56,957 Diagnostics  INFO     Connection successful
```

Questo è un tentativo di connessione riuscito, come si vede in Wireshark (10.62.148.41 è l'IP di Ncore mentre 10.62.148.75 è l'FMC):



No.	Time	Source	Destination	Protocol	Length	TCP Segment Len	Info
1	0.000000	10.62.148.41	10.62.148.75	TCP	74	0	35738 → 8302 [SYN] Seq=3050376975 Win=29200 Len=0 MSS=1460 SACK_PERM=1
2	0.000187	10.62.148.75	10.62.148.41	TCP	74	0	8302 → 35738 [SYN, ACK] Seq=1666135546 Ack=3050376976 Win=28960 Len=0
3	0.000025	10.62.148.41	10.62.148.75	TCP	66	0	35738 → 8302 [ACK] Seq=3050376976 Ack=1666135547 Win=29312 Len=0 TSval
4	0.000070	10.62.148.41	10.62.148.75	TLSv…	304	238	Client Hello
5	0.000123	10.62.148.75	10.62.148.41	TCP	66	0	8302 → 35738 [ACK] Seq=1666135547 Ack=3050377214 Win=30080 Len=0 TSval
6	0.001397	10.62.148.75	10.62.148.41	TLSv…	1514	1448	Server Hello
7	0.000007	10.62.148.41	10.62.148.75	TCP	66	0	35738 → 8302 [ACK] Seq=3050377214 Ack=1666136995 Win=32128 Len=0 TSval
8	0.000014	10.62.148.75	10.62.148.41	TLSv…	751	685	Certificate, Certificate Request, Server Hello Done
9	0.000005	10.62.148.41	10.62.148.75	TCP	66	0	35738 → 8302 [ACK] Seq=3050377214 Ack=1666137680 Win=35072 Len=0 TSval
10	0.002400	10.62.148.41	10.62.148.75	TLSv…	1625	1559	Certificate, Client Key Exchange, Certificate Verify, Change Cipher Sp
11	0.000158	10.62.148.75	10.62.148.41	TCP	66	0	8302 → 35738 [ACK] Seq=1666137680 Ack=3050378773 Win=33152 Len=0 TSval
12	0.002977	10.62.148.75	10.62.148.41	TLSv…	1252	1186	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
13	0.000497	10.62.148.41	10.62.148.75	TLSv…	111	45	Application Data
14	0.010205	10.62.148.75	10.62.148.41	TLSv…	151	85	Application Data
15	0.000494	10.62.148.41	10.62.148.75	TCP	66	0	35738 → 8302 [FIN, ACK] Seq=3050378818 Ack=1666138951 Win=37888 Len=0
16	0.000257	10.62.148.75	10.62.148.41	TLSv…	97	31	Encrypted Alert
17	0.000025	10.62.148.41	10.62.148.75	TCP	54	0	35738 → 8302 [RST] Seq=3050378819 Win=0 Len=0
18	0.000049	10.62.148.75	10.62.148.41	TCP	66	0	8302 → 35738 [FIN, ACK] Seq=1666138982 Ack=3050378819 Win=33152 Len=0
19	0.000009	10.62.148.41	10.62.148.75	TCP	54	0	35738 → 8302 [RST] Seq=3050378819 Win=0 Len=0

Il CN del certificato non corrisponde all'host remoto

Se il client eStreamer è dietro NAT, il certificato deve essere generato con l'indirizzo IP upstream o con errori simili a quelli riportati di seguito:

```
Mar  2 11:30:01 FMC SF-IMS[16921]: [16921] Event Streamer:ConnectionHandler [INFO] Accepted IPv4
connection from 10.48.26.47:46529/tcp
Mar  2 11:30:01 FMC SF-IMS[16921]: [16921] Event Streamer:ConnectionHandler [INFO] Added
10.48.26.47 to host table
Mar  2 11:30:01 FMC SF-IMS[16921]: [16921] Event Streamer:ConnectionHandler [INFO] Added
10.48.26.47(17659) to host table
Mar  2 11:30:01 FMC SF-IMS[17659]: [17659] EventStreamer child(192.168.27.100):ConnectionHandler
[INFO] Resolved CN 192.168.27.100 to 192.168.27.100
Mar  2 11:30:01 FMC SF-IMS[17659]: [17659] EventStreamer child(192.168.27.100):ConnectionHandler
[ERROR] Certificate Common Name 192.168.27.100 does not match remote host: 10.48.26.47. It was
issued to a different client.
Mar  2 11:30:02 FMC SF-IMS[16921]: [16921] Event Streamer:ConnectionHandler [INFO] Child with
pid 17659 exited with status 0
Mar  2 11:30:02 FMC SF-IMS[16921]: [16921] Event Streamer:ConnectionHandler [INFO] Removed host
entry for pid: 17659
```

Risoluzione DNS FMC per il client eStreamer non corretta

Nel caso in cui il CCP disponga di voci DNS errate per il client eStreamer, gli eventi non raggiungono il client. Per capire se questo è il problema, prendere un'acquisizione su FMC. In questo esempio, il FMC riceve un pacchetto TCP SYN dall'host client del gestore di streaming ksec-sfvm-win7-3.cisco.com:

```
root@FMC2000-2:/var/sf/archive/netmap_2# tcpdump -i eth0 port 8302
HS_PACKET_BUFFER_SIZE is set to 4.
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
18:32:45.453401 IP ksec-sfvm-win7-3.cisco.com.36428 > FMC2000-2.8302: Flags [S], seq 2427598184,
win 29200, options [mss 1460,sackOK,TS val 3681355935 ecr 0,nop,wscale 7], length 0
18:32:45.453425 IP FMC2000-2.8302 > ksec-sfvm-win7-3.cisco.com.36428: Flags [S.], seq
1996800475, ack 2427598185, win 28960, options [mss 1460,sackOK,TS val 2264897265 ecr
3681355935,nop,wscale 7], length 0
18:32:45.453539 IP ksec-sfvm-win7-3.cisco.com.36428 > FMC2000-2.8302: Flags [.], ack 1, win 229,
options [nop,nop,TS val 3681355935 ecr 2264897265], length 0
```

Èpossibile utilizzare il flag **-n** per visualizzare l'indirizzo IP risolto:

```
root@FMC2000-2:/var/sf/archive/netmap_2# tcpdump -i eth0 port 8302 -n
HS_PACKET_BUFFER_SIZE is set to 4.
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
18:34:58.015971 IP 10.62.148.41.36434 > 10.62.148.75.8302: Flags [S], seq 713101140, win 29200,
options [mss 1460,sackOK,TS val 3681488496 ecr 0,nop,wscale 7], length 0
```

In alternativa, è possibile utilizzare lo strumento di comando **nslookup** dalla CLI di FMC:

```
root@FMC2000-2:/var/sf/archive/netmap_2# nslookup ksec-sfvm-win7-3.cisco.com
Server:         1.2.3.4
Address:        1.2.3.4#53

Name: ksec-sfvm-win7-3.cisco.com Address: 10.62.148.41
```

Problema di comunicazione con eStreamer causato da un errore del certificato SSL

Verificare che il client eStreamer utilizzi il certificato SSL FMC corretto. Se il certificato non è corretto nei file FMC /var/log/message, verranno visualizzati i seguenti eventi:

```
Jun 11 14:15:33 FMC SF-IMS[14211]: [14211] Event Streamer:ConnectionHandler [INFO]
estreamer.c:2149:AcceptConnections(): Accepted IPv4 connection from 192.0.2.100:42143/tcp
Jun 11 14:15:33 FMC SF-IMS[14211]: [14211] Event Streamer:ConnectionHandler [INFO]
estreamer.c:389:allowConnection(): Added 192.0.2.100 to host table
Jun 11 14:15:33 FMC SF-IMS[14211]: [14211] Event Streamer:ConnectionHandler [INFO]
estreamer.c:334:rememberPid(): Added 192.0.2.100(13687) to host table
Jun 11 14:15:33 FMC SF-IMS[14211]: [14211] Event Streamer:ConnectionHandler [DEBUG]
estreamer.c:1347:AcceptConnection(): Created new estreamer child with src 192.0.2.100 : pid
13615
Jun 11 14:15:34 FMC SF-IMS[13687]: [13615] Event Streamer:ConnectionHandler [ERROR]
estreamer.c:1116:AcceptConnection(): SSL_accept failed, SSL_get_error reports SSL_ERROR_SYSCALL
```

Èpossibile eliminare il client eStreamer in FMC e riconfigurarlo. Il certificato SSL verrà rigenerato. Importare il nuovo certificato nel client eStreamer.

Indirizzo IP errato configurato su eStreamer per l'integrazione del modulo ASA SFR

Sul client eStreamer, è necessario utilizzare il modulo SFR IP. Sull'appliance ASA, eseguire il comando **show sfr module details** per verificare l'indirizzo IP del modulo.

ArcSight Common Event Format (CEF)

Lo standard Arcsight Common Event Format definisce le coppie chiave-valore che devono essere inviate dalla CLI di NeCore. In caso di incoerenza nei dati ricevuti al momento della ricezione di Arcisight, ad esempio: campi mancanti, non in ordine oppure alcuni dati non sono analizzati correttamente sul client Arcsight, è utile modificare la configurazione per scrivere in un file di log impostando. Questo aiuta a determinare dove sta il problema.

```
    "handler": {
        "output@comment": "If you disable all outputters it behaves as a sink",
        "outputters": [
```

```
            {
                "adapter": "cef",
                "enabled": true,
                "stream": {
                    "uri": "relfile:///data/data.{0}.cef"
                }
            }
        ],
```

Gli eventi RAW CEF vengono scritti in una riga con ogni campo separato dalla pipe "|":

```
<13>May 26 09:31:39 kali2 CEF:0|Cisco|Firepower|6.0|RNA:1003:1|CONNECTION STATISTICS|3|act=Allow
app=STUN bytesOut=820 cs1=test cs1Label=fwPolicy
cs2=Default Action cs2Label=fwRule cs3=INSIDE cs3Label=ingressZone cs4=OUTSIDE
cs4Label=egressZone cs5Label=secIntelCategory deviceExternalId=1
deviceInboundInterface=inside deviceOutboundInterface=outside dpt=9000 dst=216.151.129.103
dvchost=10.48.26.45 dvcpid=2 end=1590497212000 externalId=50850
proto=17 reason=N/A requestClientApplicatio
```

Il client eStreamer non visualizza tutti i registri

Questo problema è spesso dovuto alla sottoscrizione eccessiva del client eStreamer (troppi eventi inviati dal FMC). Eseguire questo comando sul lato client di eStreamer e verificare se il contatore Recv-Q è alto. Numero di byte non copiati dal programma utente connesso al socket. Nell'esempio, sono presenti 143143 byte in sospeso sul lato client:

```
root@kali:~# netstat -an | egrep "8302|Recv-Q"
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp   143143  0 10.62.148.41:36732      10.62.148.75:8302       ESTABLISHED
```

Controllare gli eventi ricevuti al secondo dal client eStreamer. Questo fornisce un'indicazione della frequenza di eventi al secondo:

```
root@kali:~/eStreamer-eNcore# cat estreamer.log | grep "ev/sec"
```

Provare a ridurre la quantità di dati richiesti dal client eStreamer o i tipi di eventi inviati dal CCP. In alternativa, è possibile provare ad aumentare la quantità di risorse allocate sul lato client di eStreamer.

Domande frequenti (FAQ)

Dove trovare il pacchetto NeCore-cli?

- Controllare la pagina di download del software FMC, **Firepower System Tools and API - Encore for CEF**
- In alternativa, è possibile ottenere l'ultimo file NeCore da https://github.com/CiscoSecurity/fp-05-firepower-cef-connector-arcsight/tree/master/assets

Quando è in corso un backup completo FMC, eStreamer non genera eventi. Si tratta di un comportamento normale?

Sì, è previsto il comportamento. Dalla guida di configurazione FMC Quando eseguire il backup:

Durante la raccolta dei dati di backup, è possibile che si verifichi una pausa temporanea nella correlazione dei dati (solo FMC) e che non sia possibile modificare le configurazioni relative al backup.

Sono necessarie licenze speciali per l'integrazione di FMC con il client eStreamer (ad esempio Qradar)?

No

Da dove vengono originati gli eventi eStreamer?

Il CCP. In particolare, il FMC ottiene gli eventi dai dispositivi gestiti (FTD) e li inoltra ai client eStreamer come NeCore, ArcSight, Splunk, QRadar, LogRhythm, ecc.

Esiste una matrice di compatibilità tra Splunk e NeCore?

Consultare i documenti Splunk per informazioni sulla compatibilità. Ad esempio, per vedere quali versioni di Splunk sono compatibili con NeCore versione 3.6.8, selezionare https://splunkbase.splunk.com/app/3662/

eStreamer NeCore può utilizzare i dati di più FMC?

Al momento della scrittura, no. Controllare la richiesta di miglioramento CSCvq14351

Quali sono le opzioni consigliate per configurare eStreamer per l'installazione di FMC High Availability (HA)?

Si consiglia di configurare solo l'unità FMC attiva per eStreamer. Se si configurano entrambe le unità FMC per eStreamer, il SIEM riceve gli eventi duplicati perché il FMC in standby risponde alla richiesta eStreamer. Richiesta di miglioramento correlata: CSCvi95944

L'aggiornamento di un FMC richiede la generazione manuale di nuovi certificati eStreamer?

No

Gli eventi di Security Intelligence vengono inviati al client eStreamer? È possibile selezionare gli eventi di Security Intelligence come categoria separata e inviarli a un client eStreamer?

Gli eventi di Security Intelligence (SI) sono inclusi nella categoria degli eventi di connessione e non in una categoria separata. Per questo motivo, non esiste alcun evento SI separato da inviare al gestore di streaming. Richiesta di miglioramento correlata: CSCva39052

È possibile specificare su FMC i sensori/dispositivi gestiti che hanno i loro eventi eStreamer inviati al client eStreamer?

Con un solo dominio FMC, non è possibile. Richiesta di miglioramento correlata CSCvt31270. In alternativa, è possibile configurare in FMC due domini diversi. Nel primo dominio si aggiungono tutti i dispositivi gestiti per i quali si desidera abilitare eStreamer e si configura il client eStreamer. Per il secondo dominio, si aggiungono gli altri dispositivi e non si configura eStreamer.

Qual è la versione di eStreamer su Firepower? Queste informazioni sono necessarie per la configurazione di SIEM (ad esempio LogRhythm)

Per controllare la versione di Firepower (FMC) dall'interfaccia utente di FMC, selezionare **Guida** (angolo superiore destro) **> Informazioni su > Versione software**

Quando FMC è configurato con domini come visualizzare le informazioni sul dominio nei dati FMC eStreamer?

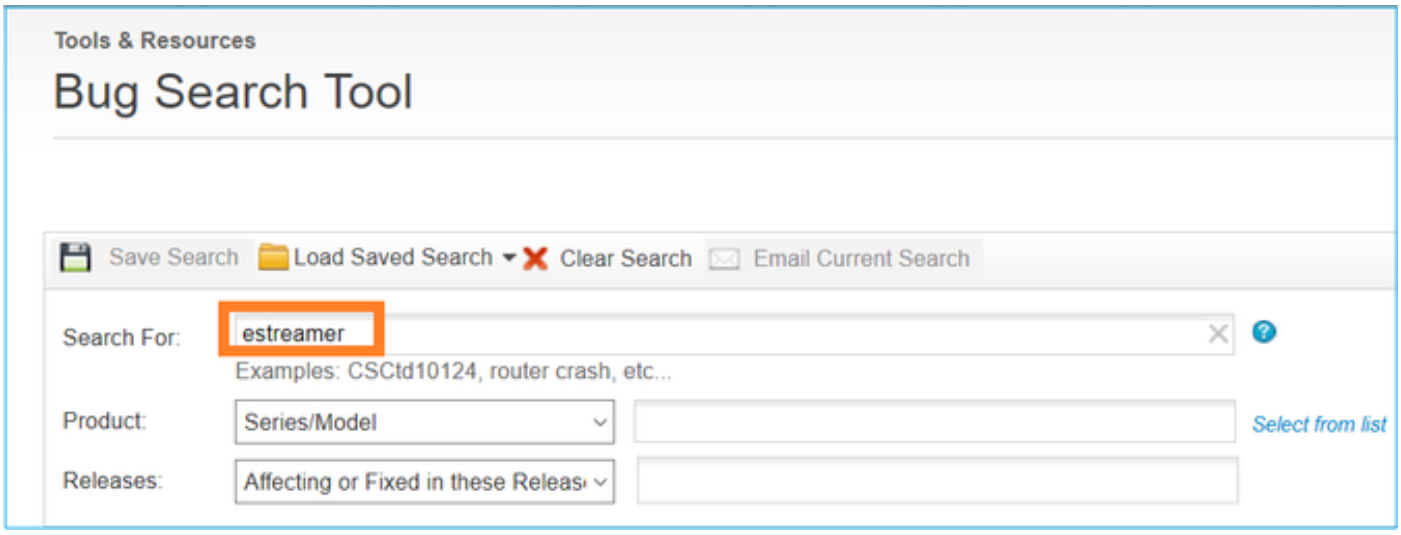
Nella guida all'integrazione di eStreamer controllare il numero **ID Netmap** accanto al Tipo di record nella sezione dell'intestazione di molti tipi di record diversi. Il numero ID Netmap può essere convertito in nome dominio o dispositivo utilizzando rispettivamente **Metadati dominio Netmap** (tipo di record 350) e **Metadati record dispositivo gestito** (tipo di record 123).

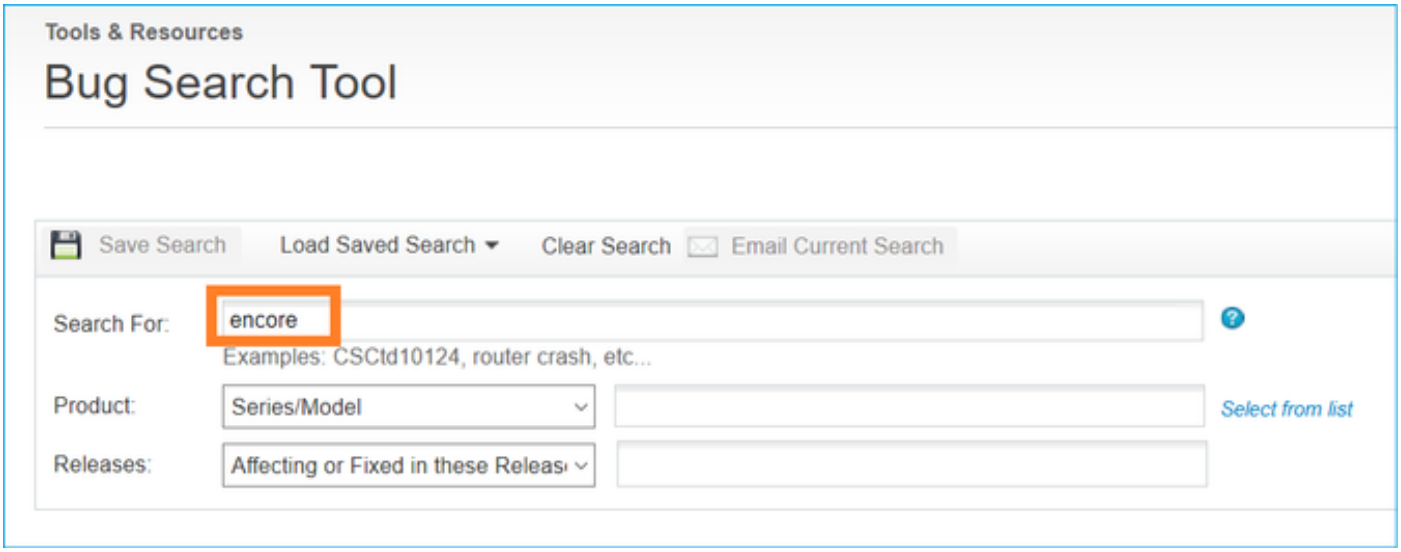
L'applicazione client deve interpretare i dati binari e i metadati in base alle informazioni fornite

nella Guida all'integrazione di eStreamer.

Problemi noti

Aprire Bug Search Tool e cercare i problemi relativi a streamer e encore, ad esempio





Informazioni correlate

- Streaming server eStreamer