HowTo ASIX Examinar Tràfic SSL

Documentació a consultar:

- man ncat
- man openssl
- man s_client
- man s_server
- man curl

seg015	Monitoritzar tràfic TLS/SSL curl, openssl s_client, openssl_s_server, ncat		ISX-M11
seg016	Python TLS/SSL Client/Servidor amb tràfic TLS/SSL: echo server Client/Servidor amb tràfic TLS/SSL: daytime server		ISX-M11

ncat

Establir un servidor ncat que permet connexions client utilitzant tràfic SSL. Usar un client ncat que permeti connectar-se a clients usant tràfic xifrat SSL. Examinar el tràfic xifrat usant netstat, ss, iptraf, tcpdump i wireshark.

SSL OPTIONS

--ssl (Use SSL) .
In connect mode, this option transparently negotiates an SSL session with an SSL server to securely encrypt the connection. This is particularly handy for talking to SSL enabled HTTP servers, etc.
In server mode, this option listens for incoming SSL connections, rather than plain untunneled traffic.

--ssl-verify (Verify server certificates) .
In client mode, --ssl-verify is like --ssl except that it also requires verification of the server certificate. Ncat comes with a default set of trusted certificates in the file ca-bundle.crt. --ssl-trustfile to give a custom list. Use -v one or more times to get details about verification failures. Ncat does not check for revoked certificates.
This option has no effect in server mode.

--ssl-cert certfile.pem (Specify SSL certificate) .
This option gives the location of a PEM-encoded certificate files used to authenticate the server (in listen mode) or the client (in connect mode). Use it in combination with --ssl-key.

--ssl-key keyfile.pem (Specify SSL private key) .
This option gives the location of the PEM-encoded private key file that goes with the certificate named with --ssl-cert.

--ssl-trustfile cert.pem (List trusted certificates) .
This option sets a list of certificates that are trusted for purposes of certificate verification. It has no effect unless combined with --ssl-verify. The argument to this option is the name of a PEM. file containing trusted certificates.
Typically, the file will contain certificates of certification authorities, though it may also contain server certificates directly. When this option is used, Ncat does not use its default certificates.

server$ ncat -l 50000 --ssl

client$ ncat --ssl localhost 50000

 # ss -t | grep 50000

ESTAB	0	0	::1:50000	::1:49219
ESTAB	0	0	::1:49219	::1:50000

```
# netstat  | grep 50000
tcp6    0      0 localhost:50000      localhost:49219      ESTABLISHED
tcp6    0      0 localhost:49219      localhost:50000      ESTABLISHED
```

openssl

Usar openssl com a client per connectar a un servidor usant tràfic xifrat. Usar openssl per exercir el rol de servidor usant tràfic SSL.

s_client
This implements a generic SSL/TLS client which can establish a transparent connection to a remote server speaking SSL/TLS. It's intended for testing purposes only and provides only rudimentary interface functionality but internally uses mostly all functionality of the OpenSSL ssl library.

s_server
This implements a generic SSL/TLS server which accepts connections from remote clients speaking SSL/TLS. It's intended for testing purposes only and provides only rudimentary interface functionality but internally uses mostly all functionality of the OpenSSL ssl library. It provides both an own command line oriented protocol for testing SSL functions and a simple HTTP response facility to emulate an SSL/TLS-aware webserver.

```
server$ ncat -l 50000 --ssl

client$ openssl  s_client -connect localhost:50000
CONNECTED(00000003)
depth=0 CN = localhost
verify error:num=18:self signed certificate
verify return:1
depth=0 CN = localhost
verify return:1
---
Certificate chain
 0 s:/CN=localhost
   i:/CN=localhost
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICBTCCAW6gAwIBAgIEJknR6zANBgkqhkiG9w0BAQUFADAUMRIwEAYDVQQDDAls
b2NhbGhvc3QwHhcNMTYwMjEyMTkzMTEyWhcNMTcwMjExMTkzMTEyWjAUMRIwEAYD
VQQDDAlsb2NhbGhvc3QwgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAML1sAAh
GcjR15pmcMCNMOxY6OZ22htJQWVuHDkpSDyVpUV9+NCD8234uNGKS1A4ZpX4ZhQy
i90cM9qUZLCdiizzm1XQBsuDz36A5SqoYhMlFAwYogMZVKMs1+tAWCpP/dNt/jKA
jOlV89N2oop59yKf4O6wYzXtIdkoSSMit5jHAgMBAAGjZDBiMBQGA1UdEQQNMAuC
CWxvY2FsaG9zdDBKBglghkgBhvhCAQ0EPRY7QXV0b21hdGljYWxseSBnZW5lcmF0
```

```
ZWQgYnkgTmNhdC4gU2VlIGh0dHA6Ly9ubWFwLm9yZy9uY2F0Ly4wDQYJKoZIhvcN
AQEFBQADgYEAbG4vXo9xPnPn+wjGaZVNzd6kNJMJZH8GUnM5uOKYQK3/htMxIoop
YRLe64PEcHM7TWbk7M4VBUKXbs9ymjsa4xZMjBaEw4fpMrpi2pOxj3jWxKGXWUhA
KOnzDM5C/Eq874P6NgZBurp8gD7wGmc7C4ERopdn7o7fftrOyn28LY8=
-----END CERTIFICATE-----
subject=/CN=localhost
issuer=/CN=localhost
---
No client certificate CA names sent
---
SSL handshake has read 834 bytes and written 397 bytes
---
New, TLSv1/SSLv3, Cipher is AES128-GCM-SHA256
Server public key is 1024 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
        Protocol  : TLSv1.2
        Cipher    : AES128-GCM-SHA256
        Session-ID:
9F6654404396DD00AF5EDF47420A190792501CD37C2DEB11DF83A1A0789BC3FC
        Session-ID-ctx:
        Master-Key:
0DF51D01ED1542ECFDCB60A8A7435B666BCCF3E6E43377A362AFE5CA12DD1D2A39AAC5367
64BB438422672A9DBA2D95E
        Key-Arg   : None
        Krb5 Principal: None
        PSK identity: None
        PSK identity hint: None
        TLS session ticket lifetime hint: 300 (seconds)
        TLS session ticket:
        0000 - b0 13 f7 b2 2f 9e e9 44-fd 1d 4a 0e 59 0f 7c 40   ..../..D..J.Y.|@
        0010 - 2c 36 04 21 7f ad 04 85-e7 6d bd a0 c8 4f b6 8f   ,6.!.....m...O..
        0020 - 34 2f ca d0 7a 94 28 95-0c 0d c8 05 df 49 a5 fc   4/..z.(......I..
        0030 - 9e 99 22 35 8f 00 f8 d7-3a e6 87 8f dc a6 95 c1   .."5....:.......
        0040 - c5 bd a2 22 c4 24 e6 07-88 da fd a9 79 63 83 d8   ...".$......yc..
        0050 - ad 60 b4 d1 d2 14 b1 4e-0a f0 78 d5 15 97 aa 6d   .`.....N..x....m
        0060 - c6 ba 82 24 b9 8c 95 e7-04 f6 37 82 09 d9 90 9b   ...$......7.....
        0070 - 57 71 7b a9 6c b6 0f 2f-22 0a 10 66 94 d9 4b 43   Wq{.l../"..f..KC
        0080 - 09 bc b9 11 e2 44 9c cd-68 84 17 dc 51 5d 8d 94   .....D..h...Q]..
        0090 - 1c 50 67 1b 89 fb 7d 87-ec 97 f0 5e 1f 83 d2 c4   .Pg...}....^....

        Start Time: 1455305513
        Timeout   : 300 (sec)
        Verify return code: 18 (self signed certificate)
---
hola que tal
tot va bé?
```

```
client$ openssl  s_client -connect www.gmail.com:443
….
GET  /  HTTP/1.0
```

 # openssl s_server -accept 50000 -nocert
Using default temp DH parameters
Using default temp ECDH parameters
ACCEPT
ERROR
3070981920:error:1408A0C1:SSL routines:SSL3_GET_CLIENT_HELLO:no shared cipher:s3_srvr.c:1396:
shutting down SSL
CONNECTION CLOSED
ACCEPT

$ openssl s_client -connect localhost:50000
CONNECTED(00000003)
3071403808:error:14077410:SSL routines:SSL23_GET_SERVER_HELLO:sslv3 alert handshake failure:s23_clnt.c:770:
---
no peer certificate available

No client certificate CA names sent

SSL handshake has read 7 bytes and written 207 bytes

New, (NONE), Cipher is (NONE)
Secure Renegotiation IS NOT supported
Compression: NONE
Expansion: NONE
---

pendent exemple openssl s_server

curl

--ssl (FTP, POP3, IMAP, SMTP)
Try to use SSL/TLS for the connection. Reverts to a non-secure connection if the server doesn't support SSL/TLS. See also --ftp-ssl-control and --ssl-reqd for different levels of encryption required. (Added in 7.20.0) This option was formerly known as --ftp-ssl (Added in 7.11.0). That option name can still be used but will be removed in a future version.

--ssl-reqd
(FTP, POP3, IMAP, SMTP) Require SSL/TLS for the connection. Terminates the connection if the server doesn't support SSL/TLS. (Added in 7.20.0)
This option was formerly known as --ftp-ssl-reqd (added in 7.15.5). That option name can still be used but will be removed in a future version.

--ssl-allow-beast
(SSL) This option tells curl to not work around a security flaw in the SSL3 and TLS1.0

protocols known as BEAST. If this option isn't used, the SSL layer may use workarounds known to cause interoperability problems with some older SSL implementations. WARNING: this option loosens the SSL security, and by using this flag you ask for exactly that. (Added in 7.25.0)

$ curl --ssl http://www.gmail.com

```
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="https://mail.google.com/mail/">here</A>.
</BODY></HTML>
```

Python: echo-client-ssl.py

Documentació:

- https://docs.python.org/2.7/library/ssl.html
- 17.3. ssl — TLS/SSL wrapper for socket objects