

Chiffrement des communications HTTP et FTP avec SSL / TLS

Projet m2l.org

Table des matières

Le chiffrement SSL / TLS.....	3
HTTPS	4
FTPS.....	9



Le chiffrement SSL / TLS

Le chiffrement SSL/TLS (Secure Sockets Layer/Transport Layer Security) est un protocole de sécurité utilisé pour sécuriser les communications sur Internet. Il assure la confidentialité et l'intégrité des données échangées entre un navigateur web (client) et un serveur web.

Regardons une trame http venant d'un formulaire sur le site www.m2l.org. En utilisant le logiciel WireShark, nous pouvons voir les données du formulaire transmises avec la méthode POST :

No.	Time	Source	Destination	Protocol	Length	Info
25	2.296824	10.107.20.66	10.31.96.80	HTTP	710	POST /index.html HTTP/1.1 (application/x-www-form-urlencoded)
> Frame 25: 710 bytes on wire (5680 bits), 710 bytes captured (5680 bits) on interface \Device\NPF_{9041CFB7-A6...}						
> Ethernet II, Src: HP_b8:dc:2e (c8:18:03:b8:dc:2e), Dst: HewlettP_19:1d:be (08:00:c7:19:1d:be)						
> Internet Protocol Version 4, Src: 10.107.20.66, Dst: 10.31.96.80						
> Transmission Control Protocol, Src Port: 54600, Dst Port: 80, Seq: 1, Ack: 1, Len: 656						
> Hypertext Transfer Protocol						
▼ HTML Form URL Encoded: application/x-www-form-urlencoded						
> Form item: "login" = "Id"						
> Form item: "mdp" = "mdp"						

Nous pouvons voir en clair le login entré ainsi que le mot de passe. Regardons maintenant le même échange de formulaire utilisant le chiffrement SSL / TLS. Le protocole utilisé est alors le protocole HTTPS :

78	1.211990	10.107.20.66	10.31.96.80	TCP	54	55133 → 443 [ACK] Seq=518 Ack=2375 Win=1051136 Len=0
79	1.212137	10.107.20.66	10.31.96.80	TLSv1.3	84	Change Cipher Spec, Application Data
80	1.212288	10.107.20.66	10.31.96.80	TCP	54	55133 → 443 [FIN, ACK] Seq=548 Ack=2375 Win=1051136 Len=0
81	1.212745	10.107.20.66	10.31.96.80	TCP	66	55136 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
82	1.212816	10.31.96.80	10.107.20.66	TCP	60	443 → 55133 [FIN, ACK] Seq=2375 Ack=548 Win=64240 Len=0
83	1.212830	10.107.20.66	10.31.96.80	TCP	54	55133 → 443 [ACK] Seq=549 Ack=2376 Win=1051136 Len=0

[TCP Segment Len: 0]	0000	00 18 03 bb dc 2e 00 00 c7 19 1d be 08 00 45 00E
Sequence Number: 2375 (relative sequence number)	0010	00 28 cc 23 40 00 3e 00 e7 40 0a 1f 60 50 0a 00P
Sequence Number (raw): 2672562934	0020	14 42 01 bb d7 5d 9f 4c 12 f6 2b 43 68 41 50 11L
[Next Sequence Number: 2376 (relative sequence number)]	0030	01 f5 05 93 00 00 00 00 00 00 00 00
Acknowledgment Number: 549 (relative ack number)			
Acknowledgment number (raw): 725837880			
0101 = Header Length: 20 bytes (5)			
Flags: 0x011 (FIN, ACK)			
Window: 501			
[Calculated window size: 64128]			
[Window size scaling factor: 128]			
Checksum: 0x0593 [unverified]			
[Checksum Status: Unverified]			
Urgent Pointer: 0			
▼ [Timestamps]			
[Time since first frame in this TCP stream: 0.012083000 seconds]			
[Time since previous frame in this TCP stream: 0.000600000 seconds]			
▼ [Seq/ACK analysis]			
[This is an ACK to the segment in frame: 80]			
[The RTT to ACK the segment was: 0.000600000 seconds]			
[RTT: 0.000652000 seconds]			



HTTPS

Pour pouvoir activer le chiffrement SSL, nous avons besoin de télécharger le paquet OpenSSL :

```
apt update && apt upgrade
apt install openssl
```

Ensuite, nous créons le répertoire qui accueillera la clé de chiffrement ainsi que le certificat :

```
mkdir /etc/ssl/localcerts
```

Nous créons ensuite la variable qui sera utilisée dans la commande de génération de clés et de certificats, puis nous générons la clé :

```
DIR=/etc/ssl/localcerts
openssl req -x509 -newkey rsa:4096 -nodes -keyout $DIR/m2lkey.key -out $DIR/m2lcert.pem -days 365
```

Nous entrons ensuite les différentes informations nécessaires à la création de la clé et du certificat :

```
root@web:~# openssl req -x509 -newkey rsa:4096 -nodes -keyout $DIR/m2lkey.key -out $DIR/m2lcert.pem -days 365
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/ssl/localcerts/m2lkey.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FR
State or Province Name (full name) [Some-State]:Haute-Vienne
Locality Name (eg, city) []:Limoges
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Beaupeyrat
Organizational Unit Name (eg, section) []:SIO
Common Name (e.g. server FQDN or YOUR name) []:m2l.org
Email Address []:luciedumas24430pro@gmail.com
```



Nous activons ensuite le Virtual Host SSL par défaut ainsi que le module SSL pour apache :

```
a2ensite default-ssl
a2enmod ssl
```

Nous modifions ensuite les chemins du Virtual Host SSL par défaut (/etc/apache2/sites-available/default-ssl.conf) :

```
# SSLCertificateFile directive is needed.
SSLCertificateFile      /etc/ssl/localcerts/m21cert.pem
SSLCertificateKeyFile   /etc/ssl/localcerts/m21key.key
```

Nous redémarrons ensuite le service apache :

```
systemctl restart apache2
```

Nous vérifions ensuite que le port 443 soit en mode « listen » :

```
netstat -nat
```

```
root@web:/etc/ssl/localcerts# netstat -nat
Connexions Internet actives (serveurs et établies)
Proto Recv-Q Send-Q Adresse locale      Adresse distante     Etat
tcp        0      0 0.0.0.0:3306         0.0.0.0:*            LISTEN
tcp        0      0 0.0.0.0:22          0.0.0.0:*            LISTEN
tcp        0      0 127.0.0.1:25        0.0.0.0:*            LISTEN
tcp6       0      0 :::80               :::*                 LISTEN
tcp6       0      0 :::22               :::*                 LISTEN
tcp6       0      0 :::1:25             :::*                 LISTEN
tcp6       0      0 :::443              :::*                 LISTEN
```

Nous ajoutons enfin à nos Virtual Hosts les directives concernant le port 443 afin que le protocole HTTPS soit pris en charge :



/etc/apache2/sites-available/www.m2l.org.conf :

```
<VirtualHost *:80>
    ServerName m2l.org
    ServerAlias www.m2l.org
    DocumentRoot /home/htdocs/m2l.org/www
    ErrorLog /var/log/apache2/www-error.log
    CustomLog /var/log/apache2/www-access.log combined
    <Directory /home/htdocs/m2l.org/www>
        Require all granted
    </Directory>
</VirtualHost>

<VirtualHost *:443>
    ServerName m2l.org
    ServerAlias www.m2l.org
    DocumentRoot /home/htdocs/m2l.org/www
    ErrorLog /var/log/apache2/www-error.log
    CustomLog /var/log/apache2/www-access.log combined
    <Directory /home/htdocs/m2l.org/www>
        Require all granted
    </Directory>
    SSLCertificateFile /etc/ssl/localcerts/m2lcert.pem
    SSLCertificateKeyFile /etc/ssl/localcerts/m2lkey.key
</VirtualHost>
```

/etc/apache2/sites-available/intranet.m2l.org.conf :

```
<VirtualHost *:80>
    ServerName m2l.org
    ServerAlias intranet.m2l.org
    DocumentRoot /home/htdocs/m2l.org/intranet
    ErrorLog /var/log/apache2/intranet-error.log
    CustomLog /var/log/apache2/intranet-access.log combined
    <Directory /home/htdocs/m2l.org/intranet>
        Require all granted
        AllowOverride All
    </Directory>
</VirtualHost>
```



```
<VirtualHost *:443>
    ServerName m2l.org
    ServerAlias intranet.m2l.org
    DocumentRoot /home/htdocs/m2l.org/intranet
    ErrorLog /var/log/apache2/intranet-error.log
    CustomLog /var/log/apache2/intranet-access.log combined
    <Directory /home/htdocs/m2l.org/intranet>
        Require all granted
        AllowOverride All
    </Directory>
    SSLCertificateFile /etc/ssl/localcerts/m2lcert.pem
    SSLCertificateKeyFile /etc/ssl/localcerts/m2lkey.key
</VirtualHost>
```

/etc/apache2/sites-available/extranet.m2l.org.conf :

```
<VirtualHost *:80>
    ServerName m2l.org
    ServerAlias extranet.m2l.org
    DocumentRoot /home/htdocs/m2l.org/extranet
    ErrorLog /var/log/apache2/extranet-error.log
    CustomLog /var/log/apache2/extranet-access.log combined
    <Directory /home/htdocs/m2l.org/extranet>
        Require all granted
    </Directory>
</VirtualHost>

<VirtualHost *:443>
    ServerName m2l.org
    ServerAlias extranet.m2l.org
    DocumentRoot /home/htdocs/m2l.org/extranet
    ErrorLog /var/log/apache2/extranet-error.log
    CustomLog /var/log/apache2/extranet-access.log combined
    <Directory /home/htdocs/m2l.org/extranet>
        Require all granted
    </Directory>
    SSLCertificateFile /etc/ssl/localcerts/m2lcert.pem
    SSLCertificateKeyFile /etc/ssl/localcerts/m2lkey.key
</VirtualHost>
```



/etc/apache2/sites-available/wiki.m2l.org.conf :

```
<VirtualHost *:80>
    ServerName m2l.org
    ServerAlias wiki.m2l.org
    DocumentRoot /home/htdocs/m2l.org/wiki
    ErrorLog /var/log/apache2/wiki-error.log
    CustomLog /var/log/apache2/wiki-access.log combined
    <Directory /home/htdocs/m2l.org/wiki>
        Require all granted
    </Directory>
</VirtualHost>

<VirtualHost *:443>
    ServerName m2l.org
    ServerAlias wiki.m2l.org
    DocumentRoot /home/htdocs/m2l.org/wiki
    ErrorLog /var/log/apache2/wiki-error.log
    CustomLog /var/log/apache2/wiki-access.log combined
    <Directory /home/htdocs/m2l.org/wiki>
        Require all granted
    </Directory>
    SSLCertificateFile /etc/ssl/localcerts/m2lcert.pem
    SSLCertificateKeyFile /etc/ssl/localcerts/m2lkey.key
</VirtualHost>
```

Nous pouvons redémarrer le service apache2 et vérifier que nous avons accès à chaque page en utilisant le protocole HTTPS.



FTPS

Nous allons maintenant appliquer ce principe d'ajout de couche de chiffrement au protocole FTP. Pour ce faire, nous allons créer dans le conteneur FTP le répertoire qui accueillera la clé de chiffrement ainsi que le certificat :

```
mkdir /etc/proftpd/ssl/
```

Nous générons la clé et le certificat SSL :

```
DIR=/etc/proftpd/ssl/  
openssl req -x509 -newkey rsa:4096 -nodes -keyout $DIR/ftpkey.key -out $DIR/ftpcert.pem -  
days 365
```

```
writing new private key to '/etc/proftpd/ssl//ftpkey.key'  
-----  
You are about to be asked to enter information that will be incorporated  
into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter '.', the field will be left blank.  
-----  
Country Name (2 letter code) [AU]:FR  
State or Province Name (full name) [Some-State]:Haute-Vienne  
Locality Name (eg, city) []:Limoges  
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Beaupeyrat  
Organizational Unit Name (eg, section) []:SIO  
Common Name (e.g. server FQDN or YOUR name) []:ftp  
Email Address []:luciedumas24430@gmail.com
```

Nous allons ensuite modifier le fichier `/etc/proftpd/proftpd.conf` pour pour décommenter la ligne incluant le TLS :

```
# This is used for FTPS connections  
#  
Include /etc/proftpd/tls.conf
```

Nous allons maintenant éditer le fichier `/etc/proftpd/tls.conf` pour modifier les lignes suivantes :

- TLSEngine (activer/désactiver TLS)
- TLSLog (logguer les connexions chiffrées dans un fichier à part)
- TLSRSACertificateFile (chemin vers le certificat)



- TLSRSACertificateKeyFile (chemin vers la clé)
- TLSOptions (voir http://www.proftpd.org/docs/contrib/mod_tls.html)

```
<IfModule mod_tls.c>
TLSEngine                                on
TLSLog                                   /var/log/proftpd/tls.log
#TLSProtocol                             SSLv23
#
# Server SSL certificate. You can generate a self-signed certificate using
# a command like:
#
# openssl req -x509 -newkey rsa:1024 \
#             -keyout /etc/ssl/private/proftpd.key -out /etc/ssl/certs/proftpd.crt \
#             -nodes -days 365
#
# The proftpd.key file must be readable by root only. The other file can be
# readable by anyone.
#
# chmod 0600 /etc/ssl/private/proftpd.key
# chmod 0640 /etc/ssl/private/proftpd.key
#
TLSRSACertificateFile                     /etc/proftpd/ssl/ftpcert.pem
TLSRSACertificateKeyFile                  /etc/proftpd/ssl/ftpkey.key
```

Nous devons ensuite activer les modules TLS. Pour cela, nous modifions le fichier `/etc/proftpd/modules.conf` :

```
# Install proftpd-mod-crypto to use this module for TLS/SSL support.
LoadModule mod_tls.c
```

Nous installons ensuite les paquets nécessaires :

```
apt install proftpd-mod-crypto
```

Nous redémarrons ensuite le conteneur :

```
systemctl restart proftpd
```

Nous pouvons tester maintenant le transfert de fichier via le site FileZilla.

