Faculté des Sciences de Bizerte (FSB)
Université de Carthage

Pratiques Cryptographiques

TP – Mise en place du TLS : Cas du Web (HTTPS)

Apache/Mod_SSL/PHP

CI 2

Semestre 2

Dr. Ing. Nizar Ben Neji
nizar.benneji@fsb.ucar.tn

2024 / 2025

Prérequis

- Créer un utilisateur ***fsb*** avec les privilèges du super utilisateur:

```
# su -
# adduser fsb
# passwd fsb
# visudo          # Editer le fichier des sudoers et vérifier le groupe wheel
                  # pour permettre aux utilisateurs du groupe wheel
                  # d'exécuter toutes les commandes
```

```
## Allows people in group wheel to run all commands
%wheel  ALL=(ALL)       ALL
```

- Ajouter l'utilisateur ***fsb*** au group wheel:

```
# usermod  -aG  wheel  fsb
```

- S'identifier avec l'utilisateur ***fsb*** pour toute la suite du TP:

```
# su  -  fsb
```

Prérequis

- Vérifier la présence des paquetages d'Apache, du Module SSL et du PHP nécessaires pour le TP:

```
# rpm  -qa | grep  httpd
# rpm  -qa | grep  mod_ssl
# rpm  -qa | grep  php
```

- Mettre à jour ***yum***:

```
# sudo  yum  update
```

- Installer ***Apache/mod_ssl/php***:

```
# sudo  yum  install  httpd
# sudo  yum  install  mod_ssl
# sudo  yum  install  php
```

Configuration du service Apache

- Démarrer le service d'Apache:

```
# sudo  systemctl  start  httpd
```

- Lancer Apache au démarrage de la machine:

```
# sudo  systemctl  enable  httpd
```

- Autoriser le trafic Web clair et chiffré au niveau du parefeu du système:

```
# sudo  firewall-cmd  --permanent   --add-port=80/tcp
# sudo  firewall-cmd  --permanent   --add-port=443/tcp
```

- Recharger la configuration du parefeu

```
# sudo  firewall-cmd  --reload
```

- Vérifier l'ouverture des ports 80 et 443 comme suite:

```
# sudo  firewall-cmd  --list-ports
# sudo  iptables -L
```

Configuration du service Apache

- Démarrer le service d'Apache:

```
# sudo  systemctl  start  httpd
```

- Configurer le VirtualHost pour le premier site Web en créant le fichier de configuration du site sous **/etc/httpd/conf.d/**:

```
# sudo  vi  /etc/httpd/conf.d/vHostWebsite1.conf

NameVirtualHost *:80
<VirtualHost *:80>
ServerName demo.com
ServerAlias www.demo.com
DocumentRoot /var/www/html/demo/
ErrorLog /var/www/html/demo/logs/error.log
CustomLog /var/www/html/demo/logs/access.log combined
</VirtualHost>
```

- Créer le répertoire du site:

```
# sudo mkdir -p /var/www/html/demo/logs
```

Configuration du service Apache

- Créer une page d'accueil pour le site **demo**:

```
#sudo  vi  /var/www/html/demo/index.php
<?php
 echo "PAGE D'ACCEUIL SITE DEMO";
?>
```

- Changer le propriétaire du répertoire du site **demo**:

```
# chown -R apache.apache  /var/www/html/demo/
```

- Redémarrer Apache:

```
# sudo systemctl restart httpd
```

- Le système de sécurité système SELinux va empêcher Apache d'écrire au niveau de /var/www/html/logs. Changer le niveau de restriction à permissive comme suite:

```
# sudo  setenforce  0
# sudo  getenforce        # pour vérifier le mode SELinux mis en place
```

Configuration du service Apache

- Ajouter les noms de domaine du site au niveau du fichier **/etc/hosts**:

#sudo vi /etc/hosts

```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.1.4   www.demo.com   demo.com
```

- Faites la même chose sur votre machine Windows pouvoir accéder au site depuis votre Windows:

C:\Windows\System32\drivers\etc\hosts

- Visualiser le site à partir de votre navigateur:

Configuration du module SSL d'Apache

- Pour la configuration du module SSL d'Apache on a besoin d'un certificat électronique TLS certifiant le nom du domaine du site:

```
#cd /home/fsb/pki
#mkdir certs/server
#openssl genrsa -out certs/server/server.key -des3 2048
#vi config/openssl_ssl.cnf #Personaliser les lignes qui concernent la CA
                           #et le serveur
#openssl req -new -key certs/server/server.key -out
certs/server/server.req -config config/openssl_ssl.cnf
#openssl ca -config config/openssl_ssl.cnf -in certs/server/server.req
-out certs/server/server.crt
```

- Copier les fichiers créés dans le répertoire d'Apache:

```
# cp -r certs/server /etc/httpd/conf.d
# sudo cp ca/fsb.crt /etc/httpd/conf.d/server/
```

- Changer les paramètres d'accès de ce repertoire comme suite:

```
#sudo chmod -R 700 /etc/httpd/conf.d/server/
```

Configuration du module SSL d'Apache

- Editer le fichier de configuration du module SSL d'Apache:

```
#sudo vi  /etc/httpd/conf.d/ssl.conf
```

- Activer et changer les lignes suivantes:

```
<VirtualHost _default_:443>

# General setup for the virtual host, inherited from global configuration
DocumentRoot "/var/www/html/demo"
ServerName www.demo.com:443
#   Server Certificate:
# Point SSLCertificateFile at a PEM encoded certificate.  If
# the certificate is encrypted, then you will be prompted for a
# pass phrase.  Note that a kill -HUP will prompt again.  A new
# certificate can be generated using the genkey(1) command.
SSLCertificateFile /etc/httpd/conf.d/server/server.crt

#   Server Private Key:
#   If the key is not combined with the certificate, use this
#   directive to point at the key file.  Keep in mind that if
#   you've both a RSA and a DSA private key you can configure
#   both in parallel (to also allow the use of DSA ciphers, etc.)
SSLCertificateKeyFile /etc/httpd/conf.d/server/server.key

#   Server Certificate Chain:
#   Point SSLCertificateChainFile at a file containing the
#   concatenation of PEM encoded CA certificates which form the
#   certificate chain for the server certificate. Alternatively
#   the referenced file can be the same as SSLCertificateFile
#   when the CA certificates are directly appended to the server
#   certificate for convinience.
SSLCertificateChainFile /etc/httpd/conf.d/supcom.crt
```

Configuration du module SSL d'Apache

- Redémarrer le service Apache pour qu'il prend en considération les nouvelles modifications. Pour ce nouveau démarrage Apache va demander le mot de passe protégeant la clé privée du serveur:

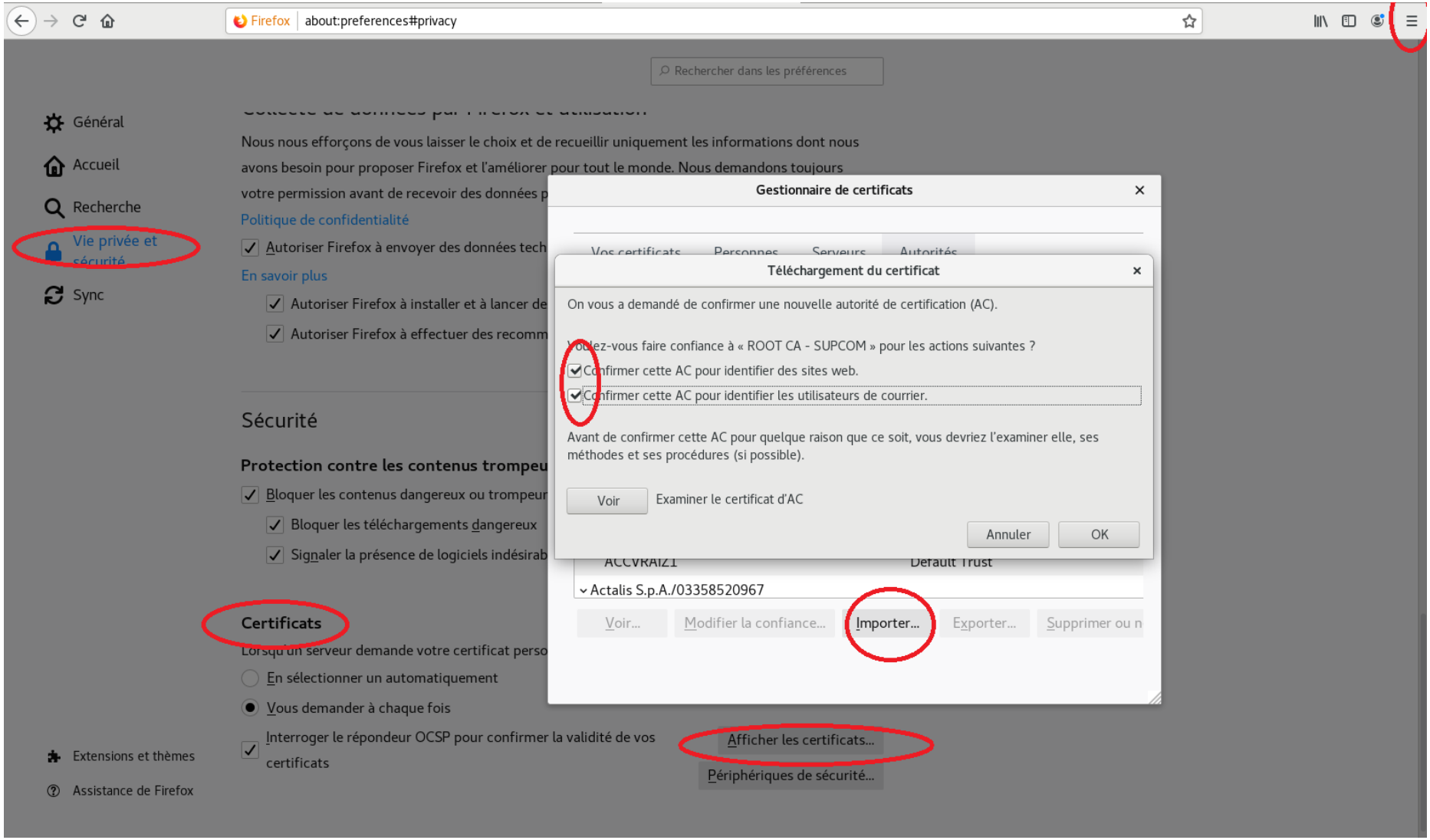
```
[supcom@localhost pki]$ sudo systemctl restart httpd
Enter SSL pass phrase for www.demo.com:443 (RSA) : ******
```

- Ré-ouvrir le site demo en mode sécurisé à partir du navigateur: https://www.demo.com . Une erreur TLS va apparaitre en relation avec le certificat de l'autorité de la FSB:

- Ajouter le certificat de l'autorité **fsb.crt** au niveau du truststore du navigateur. Pour Firefox, il faut procéder comme suite:

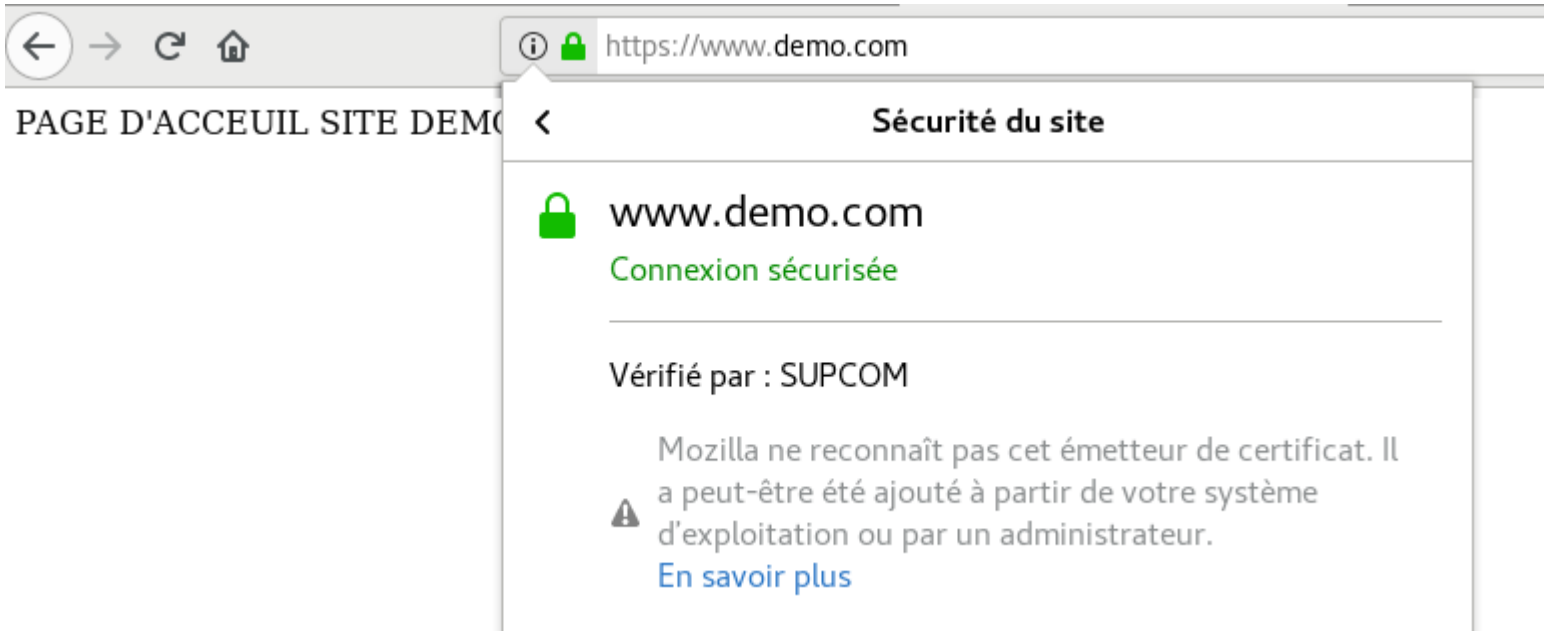
Menu > Preferences > Vie Privée et Sécurité > Certificats > Afficher les certificats > Autorités > Importer

Configuration du module SSL d'Apache



Configuration du module SSL d'Apache

- Ré-tester l'accès au site en mode sécurisé à partir du navigateur: https://www.demo.com .



Configuration du module SSL d'Apache

- Pour forcer l'accès **https**, placer les lignes suivantes au niveau du Virtual Host du site et redémarrer Apache:

```
RewriteEngine on
RewriteCond %{SERVER_PORT} !^443$
RewriteRule ^/(.*) https://%{SERVER_NAME}/$1 [L,R]
```

- Créer une zone **admin** au niveau du site **demo** à laquelle on va restreindre l'accès par certificat, par adresse IP et dans le temps:

```
#sudo mkdir  /var/www/html/demo/admin/
#sudo vi /var/www/html/demo/admin/index.php
<?php
 echo "Zone Administrateur";
 echo $_SERVER['SSL_CLIENT_M_SERIAL'];
 echo $_SERVER['SSL_CLIENT_S_DN'];
 echo $_SERVER['SSL_CLIENT_I_DN'];
?>
```

Configuration du module SSL d'Apache

- Créer un certificat utilisateur (admin) à installer par la suite au niveau du navigateur pour pouvoir accéder à la zone d'administration du site:

```
#cd  /home/fsb/pki
#mkdir  certs/user/
#openssl  genrsa  -out certs/user/user.key  -des3  2048
#vi  config/openssl_mail.cnf
#openssl  req  -new  -key  certs/user/user.key  -out  certs/user/user.req
-config config/openssl_user.cnf
#openssl  ca  -config  config/openssl_user.cnf  -in  certs/user/user.req
-out certs/user/user.crt
#openssl  pkcs12  -export
                    -inkey          certs/user/user.key
                    -in             certs/user/user.crt
                    -out            certs/user/user.p12
                    -certfile       ca/fsb.crt
                    -name           " User First & Last Name"
```

Configuration du module SSL d'Apache

- Re-Editer le fichier de configuration du module SSL d'Apache:

```
#sudo vi  /etc/httpd/conf.d/ssl.conf
```

- Activer et changer les lignes suivantes pour la sécurisation de la zone **admin** puis redémarrer le service **httpd** et tester l'accès à la zone **admin**:

```
<Location /admin>
#   Certificate Authority (CA):
#   Set the CA certificate verification path where to find CA
#   certificates for client authentication or alternatively one
#   huge file containing all of them (file must be PEM encoded)
SSLCACertificateFile /etc/httpd/conf.d/server/supcom.crt

#   Client Authentication (Type):
#   Client certificate verification type and depth.  Types are
#   none, optional, require and optional_no_ca.  Depth is a
#   number which specifies how deeply to verify the certificate
#   issuer chain before deciding the certificate is not valid.
SSLVerifyClient require
SSLVerifyDepth  1

#   Access Control:
#   With SSLRequire you can do per-directory access control based
#   on arbitrary complex boolean expressions containing server
#   variable checks and other lookup directives.  The syntax is a
#   mixture between C and Perl.  See the mod_ssl documentation
#   for more details.
#<Location />
SSLRequire (    %{SSL_CIPHER} !~ m/^(EXP|NULL)/ \
            and %{SSL_CLIENT_S_DN_O} eq "Snake Oil, Ltd." \
            and %{SSL_CLIENT_S_DN_OU} in {"Staff", "CA", "Dev"} \
            and %{TIME_WDAY} >= 1 and %{TIME_WDAY} <= 5 \
            and %{TIME_HOUR} >= 8 and %{TIME_HOUR} <= 20       ) \
            or %{REMOTE_ADDR} =~ m/^192\.76\.162\.[0-9]+$/
#</Location>
</Location>
```