# Exam final – Infrastructure clé publique

Objectif : Créer une chaîne de certification complète.

Etapes de réalisation : Créer une CA racine, Créer une CA intermédiaire, Signature du certificat pour serveur.

Prérequis : OpenSSL installé, environnement de travail propre

## Initialisation de l’environnement du TP :

1. Structure des répertoires :

Tout d’abord, je vais construire mon arborescence de dossiers/fichiers, en utilisant les commandes suivantes :

mkdir -p /etc/ssl/ca\_racine/{certs,csr,newcerts,private}

mkdir -p /etc/ssl/ca\_intermediaire/{certs,csr,newcerts,private}

mkdir -p /etc/ssl/ca\_serveur/{certs,csr,newcerts,private}

touch /etc/ssl/ca\_racine/index.txt

touch /etc/ssl/ca\_intermediaire/index.txt

echo 1000 > /etc/ssl/ca\_racine/serial

echo 1000 > /etc/ssl/ca\_intermediaire/serial

1. Installation des dépendances :

Ensuite, je vais installer les dépendances nécessaires au travail pratique, en utilisant les commandes suivantes :

apt install openssl

apt install apache2

## Création de l’autorité de certification racine

1. Placer le fichier openssl.cnf dans le repertoire

Je déplace le fichier openssl.cnf dans le repertoire du CA racine à l’intérieur duquel je place les configurations de base pour le CA racine.

cp ./openssl\_racine.cnf /etc/ssl/ca\_racine/

mv /etc/ssl/ca\_racine/openssl\_racine.cnf /etc/ssl/ca\_racine/openssl.cnf

1. Génération des clés

Je génère une paire de clés avec openssl avec comme arguments :

« -aes256 » : algorithme aes 256

«-out » : fichier de sortie de la commande openssl

« 4096 » : taille de la clé en bits

openssl genrsa -aes256 -out /etc/ssl/ca\_racine/private/ca.key.pem \   
 4096

chmod 400 /etc/ssl/ca\_racine/private/ca.key.pem

J’affiche le résultat :

Une image contenant capture d’écran, texte

Le contenu généré par l’IA peut être incorrect.

1. Créer le certificat auto-signé

Je créée un certificat auto-signé avec la commande openssl avec les auguments suivants :

« req » : Lance la commande de création de requête de certificat ou de certificat auto-signé.

« -key » : chemin vers la clé privée

« -new » : Génère une **nouvelle requête** (même si ici on utilise -x509 pour l’auto-signature).

« - x509 » : Crée un **certificat auto-signé** au lieu d'une CSR. Utilisé pour les certificats racine.

« -days » : Durée de validité du certificat

« -sha256 » : Utilise l'algorithme de hachage **SHA-256** pour la signature.

« -extensions » : Utilise la section [ v3\_ca ] définie dans le fichier openssl.cnf (obligatoire pour un certificat de type CA)

«-out » : fichier de sortie de la commande openssl

openssl req -subj "/C=FR/ST=HATE-GARONNE/L=TOULOUSE/O=ESGI/OU=ESGI-TOULOUSE/CN=RootCA"\

-key /etc/ssl/ca\_racine/private/ca.key.pem \

-new -x509 -days 7300 -sha256 -extensions v3\_ca \

-out /etc/ssl/ca\_racine/certs/ca.cert.pem

chmod 444 /etc/ssl/ca\_racine/certs/ca.cert.pem

J’affiche le résultat :

Une image contenant texte, capture d’écran, Police, document

Le contenu généré par l’IA peut être incorrect.

## Création de l’autorité de certification intermédiaire

1. Créer le fichier openssl.cnf

Je déplace le fichier openssl.cnf dans le repertoire du CA intermédiaire à l’intérieur duquel je place les configurations de base pour le CA intermédiaire.

cp ./openssl\_intermediaire.cnf /etc/ssl/ca\_intermediaire/

mv /etc/ssl/ca\_intermediaire/openssl\_intermediaire.cnf /etc/ssl/ca\_intermediaire/openssl.cnf

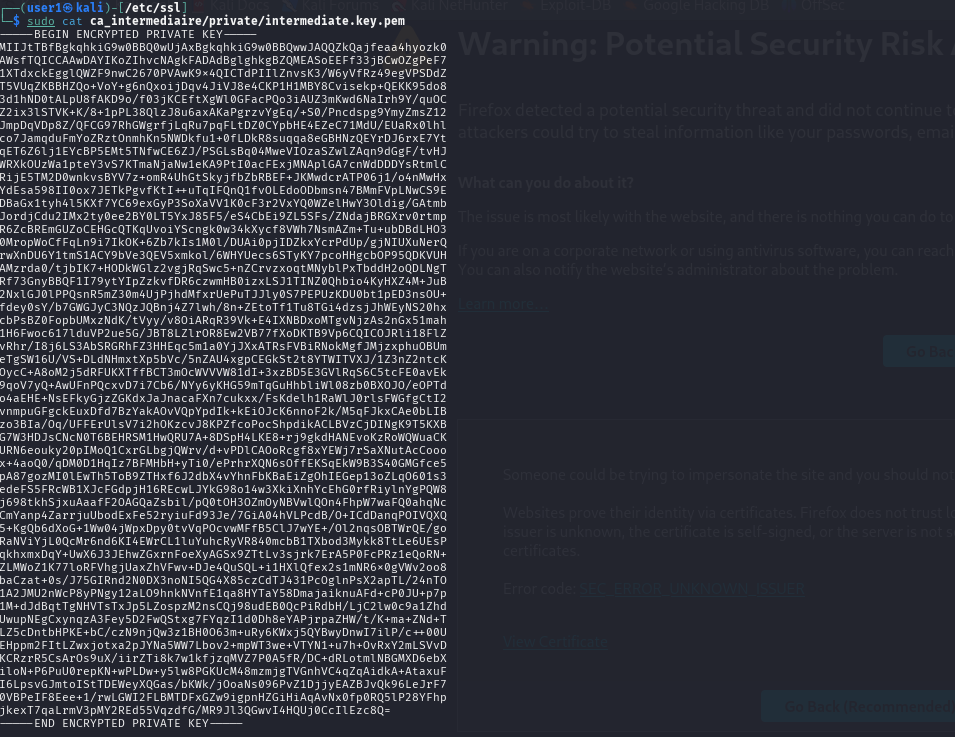
1. Génaration des clés

Je génère une paire de clés avec les mêmes arguments que la fois précédente.

openssl genrsa -aes256 -out \  
 /etc/ssl/ca\_intermediaire/private/intermediate.key.pem \  
 4096

chmod 400 certs/intermediate/private/intermediate.key.pem

J’affiche le résultat :



1. Générer une CSR pour la CA intermédiaire

Je créée un certificat auto-signé avec la commande openssl avec les mêmes arguments que la fois précédente.

openssl req req -subj "/C=FR/ST=HATE-GARONNE/L=TOULOUSE/O=ESGI/OU=ESGI-TOULOUSE/CN=RootCA" -new \ -sha256 \

-key /etc/ssl/ca\_intermediaire/private/intermediate.key.pem \

-out /etc/ssl/ca\_intermediaire/csr/intermediate.csr.pem

J’affiche le résulat :

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

1. Signer la CSR par la CA racine

Je signe la CSR par l’autorité de certification racine avec la commande openssl :

« ca » : Utilise la fonction de signature de certificats d'une autorité de certification (CA)

« -config » : Fichier de configuration contenant tous les paramètres pour la CA racine (chemins, politiques, extensions, etc.).

« -extension » : Indique d’utiliser la section « [ v3\_intermediate\_ca ] » du fichier openssl.cnf, qui définit les extensions spécifiques aux CAs intermédiaires.

« -days » : La validité du certificat

« -notext » : Ne pas inclure la représentation en clair du certificat dans la sortie. Cela allège un peu le fichier

« -md » : Algorithme de signature

« -in » : CSR (Certificate Signing Request) générée par la CA intermédiaire, que la CA racine va signer.

« -out » : Chemin où le certificat signé de la CA intermédiaire sera sauvegardé

openssl ca -config /etc/ssl/ca\_racine/openssl.cnf \

-extensions v3\_intermediate\_ca \

-days 3650 -notext -md sha256 \

-in /etc/ssl/ca\_intermediaire/csr/intermediate.csr.pem \

-out /etc/ssl/ca\_intermediaire/certs/intermediate.cert.pem

chmod 444 /etc/ssl/ca\_intermediaire/certs/intermediate.cert.pem

J’affiche le résultat :

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

## Création un certificat coté serveur

1. Génération de la clé privée pour un serveur

Je génère une paire de clés avec les mêmes arguments que la fois précédente.

openssl genrsa -out /etc/ssl/ca\_serveur/private/server.key.pem 2048

chmod 400 /etc/ssl/ca\_serveur/private/server.key.pem

J’affiche le résultat :

Une image contenant texte, capture d’écran, Police, conception

Le contenu généré par l’IA peut être incorrect.

1. Générer une CSR pour le serveur

Je créée un certificat auto-signé avec la commande openssl avec les mêmes arguments que la fois précédente.

openssl req -subj "/C=FR/ST=HATE-GARONNE/L=TOULOUSE/O=ESGI/OU=ESGI-TOULOUSE/CN=Server" \

-key /etc/ssl/ca\_serveur/private/server.key.pem \

-new -sha256 -out /etc/ssl/ca\_serveur/csr/server.csr.pem

J’affiche le résultat :

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

1. Signature du certificat serveur par le CA intermédiaire

Je signe la CSR par l’autorité de certification intermédiaire avec les mêmes arguments que la fois précédente.

openssl ca -config /etc/ssl/ca\_intermediaire/openssl.cnf \

-extensions server\_cert -days 750 -notext -md sha256 \

-in /etc/ssl/ca\_serveur/csr/server.csr.pem \

-out /etc/ssl/ca\_serveur/certs/server.cert.pem

chmod 444 /etc/ssl/ca\_serveur/certs/server.cert.pem

Une image contenant texte, capture d’écran, Police, conception

Le contenu généré par l’IA peut être incorrect.

1. Créer la chaine de certification

Je concatène les certificats serveur, intermédiaires puis racine dans un seul fichier, pour créer la chaine de certification.

cat /etc/ssl/ca\_intermediaire/certs/intermediate.cert.pem\  
 /etc/ssl/ca\_racine/certs/ca.cert.pem \

    /etc/ssl/ca\_serveur/certs/server.cert.pem > \  
 /etc/ssl/ca\_intermediaire/certs/ca-chain.cert.pem

chmod 444 /etc/ssl/ca\_intermediaire/certs/ca-chain.cert.pem

J’affiche le résultat :

Une image contenant texte, capture d’écran, livre

Le contenu généré par l’IA peut être incorrect.

## Configuration ssl d’Apache

1. Activer les modules Apache nécessaires

J’installe les module neccessaire au bon fonctionnement d’un site avec ssl, le module de gestion des connexions HTTPS (mod\_ssl) et le module qui authorise la réécriture d’une url (mod\_rewrite).

sudo a2enmod ssl

sudo a2enmod rewrite

1. Créer un dossier pour stocker les certificats

APACHE\_SSL\_DIR="/etc/apache2/ssl"

sudo mkdir -p $APACHE\_SSL\_DIR

1. Copier les certificats et la clé dans le dossier Apache

sudo cp /etc/ssl/ca\_serveur/certs/server.cert.pem $APACHE\_SSL\_DIR/server.cert.pem

sudo cp /etc/ssl/ca\_intermediaire/certs/ca-chain.cert.pem $APACHE\_SSL\_DIR/ca-chain.cert.pem

sudo cp /etc/ssl/ca\_serveur/private/server.key.pem $APACHE\_SSL\_DIR/server.key.pem

sudo chmod 600 $APACHE\_SSL\_DIR/server.key.pem

sudo chmod 644 $APACHE\_SSL\_DIR/server.cert.pem $APACHE\_SSL\_DIR/ca-chain.cert.pem

1. Créer le fichier de configuration Apache SSL

On modifie la configuration de base du fichier Apache pour y ajouter les fichiers de certification nécessaires avec les options :

« SSLCertificateFile » : Spécifie le certificat du serveur

« SSLCertificateChainFile » : Spécifie la chaîne de certification menant à l’autorité racine

« SSLCertificateKeyFile » : Spécifie la clé privée du certificat du serveur

APACHE\_SSL\_CONF="/etc/apache2/sites-available/default-ssl.conf"

sudo tee $APACHE\_SSL\_CONF > /dev/null <<EOF

<VirtualHost \*:443>

ServerName localhost

ServerAdmin webmaster@localhost

DocumentRoot /var/www/html

SSLEngine on

SSLCertificateFile $APACHE\_SSL\_DIR/server.cert.pem

SSLCertificateChainFile $APACHE\_SSL\_DIR/ca-chain.cert.pem

SSLCertificateKeyFile $APACHE\_SSL\_DIR/server.key.pem

<Directory /var/www/html>

Options Indexes FollowSymLinks

AllowOverride All

Require all granted

</Directory>

ErrorLog \${APACHE\_LOG\_DIR}/error.log

CustomLog \${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

EOF

1. Activer le site SSL

J’instancie et démarre le site par default.

sudo a2ensite default-ssl

1. Redémarrer Apache

J’applique les changements apportés à Apache.

sudo systemctl restart apache2

## Vérification des certificats

1. Fonctionnement du site

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.

1. Certificat serveur

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.

1. Certificat CA intermédiaire

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.

1. Certificat CA racine

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.