

COMMENT CREER VOTRE PROPRE CERTIFICAT NUMERIQUE X509!

Je sais que ce n'est pas le bon endroit mais comme je ne sais pas où voici : je voudrais l'avis de la Communauté sur le meilleur (le plus sûr) logiciel gratuit de cryptage de mail. Maintenant que PGP est passé chez SYMANTEC il faut payer pour l'avoir.

A utiliser sur PC et Mac

Merci.

Tous les bons maillers (et mêmes des moins bons comme TB) intègrent le cryptage/signature à l'aide de certificats X509, c'est bien plus simple que d'utiliser PGP ou autre gadget et moins contraignants/emmerdant pour ton correspondant.

Il te faut juste un certificat utilisateur X509 à usage signature/cryptage mail.

T'en as des payants, il existait du gratos chez Tawte pour les particuliers ou te peux te générer toi-même un auto-signé avec la recette ci-dessous.

Si t'as un serveur Windows 200 ou + sous la main, tu peux aussi te monter une p'tite autorité de certification autonome pour générer tes certificats.

Il existait une page très sympa avec la recette pour ce faire soi-même ces p'tits certificats avec OpenSSL ... mais sans les affres de OpenSSL :

<http://www.chez.com/winterminator/x509.html>

Cette page à disparu et même dans les "caches" genre web archives elle est très dur à récupérer.

Je mets ci-dessous un copier-coller d'une copie que j'avais fait, pour les binaires d'OpenSSL, ça se télécharge simplement à la source: <http://www.openssl.org/>

<http://www.chez.com/winterminator/x509.html>

COMMENT CREER VOTRE PROPRE CERTIFICAT NUMERIQUE X509!

Par WinTerMiNator

Copyright (c) 2000-2002 Michel Nallino aka Winterterminator.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with NO Invariant Sections, with NO Front-Cover Texts, and with NO Back-Cover Texts.

A copy of the license is included here.

Obtenir un certificat numérique?

C'est en général cher (150 F pour 1 an à Certinomis) et compliqué. Si c'est gratuit, il faut donner des renseignements confidentiels (Thawte), ou bien c'est limité dans le temps.

Alors, voici le "livre de cuisine" pour créer votre propre CA (Autorité de Certification) et votre propre certificat numérique X 509.

Quelle est sa valeur? La même qu'une clé PGP / GnuPG non certifiée.

Une clé PGP / GnuPG vous permet d'utiliser PGP / GnuPG, de signer vos mails et de recevoir des mails chiffrés, elle ne garantit rien quant à votre identité. Un certificat numérique non signé par une CA vous permet d'utiliser S/MIME, de signer vos mails et de recevoir des mails chiffrés avec votre logiciel client de messagerie, il ne garantit rien quant à votre identité.

Ce qui suit repose sur OpenSSL, un logiciel libre très puissant mais réputé pour être quasi incompréhensible (plus d'une centaine de commandes, 15 à 20 options par commande, un help dans lequel on se noie, une doc jamais terminée...). Je ne prétends donc pas vous expliquer OpenSSL, mais seulement comment fabriquer votre certificat.

Je me suis inspiré des scripts de Yeak Nai Siew, que j'ai simplifiés et adaptés à Windows.

1. Pour Unix / Linux:

Sur le site <http://www.openssl.org>, Télécharger le kit d'installation d'OpenSSL. Dans la rubrique "contributions", télécharger ssl.ca-0.1.tar.gz (par Yeak Nai Siew).

Aspirer la doc, depuis la page "documents". (autre doc intéressante: Open-source PKI Book, <http://ospkibook.sourceforge.net>)

Compiler et installer OpenSSL. Décompresser ssl.ca-0.1 et suivre les instructions du Readme qu'il contient.

2. Pour Windows :

Il faut d'abord télécharger une version d'OpenSSL compilée pour Windows!

J'ai utilisé avec succès la version 0.9.6g, qui se télécharge depuis <http://curl.haxx.se/download.html> ou <http://www.stunnel.org>. Vous pouvez, bien entendu, utiliser une version plus récente (de temps en temps, une vérification sur l'un de ces sites permettra de suivre les évolutions d'OpenSSL et d'être toujours à jour).

Créer ensuite un répertoire (où l'on veut), par exemple C:\OpenSSL.

Puis y copier:

"openssl.exe" et ses deux dll,

les sept fichiers batchs et de configuration obtenus par copier / coller à partir de cette page (voir ANNEXE)

un sous-répertoire vide, à créer, "newcerts"

un fichier vide, créé avec notepad, "index.txt"

un fichier, créé avec notepad, qui contient le numéro: 01, "serial.txt"

un fichier à créer, nommé "seed.rnd", qui doit contenir des données pseudo-aléatoires;

quelques méthodes d'obtention: copier le "random.seed" généré par GnuPG à la première génération de clé, ou le "randseed.rnd" créé par PGP, et le renommer "seed.rnd", enregistrer le bruit électronique de votre carte son (tous amplis à fond, microphone débranché), et renommer le fichier wav en "seed.rnd", prendre un gros fichier, le compresser avec Winzip / PKZIP ou le chiffrer avec un algorithme de chiffrement symétrique (rester en binaire), le découper en tranches, prendre une tranche du milieu (sans en-tête ni fin de fichier) et en faire "seed.rnd", ouvrir Notepad et taper n'importe quoi, enregistrer le fichier et le nommer "seed.rnd".

Pour aller plus vite, vous pouvez télécharger et installer VotreCA.zip, prêt à fonctionner, qui contient tous les fichiers nécessaires! (essayez quand même de créer votre propre "seed.rnd").

Personnaliser les fichiers suivants (les modifier avec un éditeur de texte, Notepad ou Wordpad par exemple):

root-ca.bat (personnaliser taille de clé RSA de CA en bits, durée du certificat CA en jours)

user-cert.bat (personnaliser taille de clé RSA d'utilisateur, en bits)

ca-sign.cnf (personnaliser durée du certificat utilisateur, en jours)

p12.bat (personnaliser User name, CA NAME).

La réalisation et l'installation d'un certificat numérique X509 se fait ensuite en plusieurs étapes:

a) Création d'une CA:

En mode "ligne de commandes", sous Windows, dans une "fenêtre MS-DOS", exécuter "root-ca.bat". Répondre aux questions. En final on obtient:

"ca.crt", certificat de CA,

"ca.key", clé privée (cryptée) associée.

b) Création d'un formulaire de demande de certificat:

En mode "ligne de commandes", sous Windows, dans une "fenêtre MS-DOS", exécuter "user-cert.bat". Répondre aux questions. En final on obtient:

"user.csr", formulaire de demande de certificat utilisateur,

"user.key", clé privée associée, non cryptée (car elle devra être packagée en PKCS#12).

c) Signature du certificat par la CA:

En mode "ligne de commandes", sous Windows, dans une "fenêtre MS-DOS", exécuter "ca-sign.bat". Répondre aux questions. En final on obtient:

"user.crt", certificat utilisateur.

d) Création d'un PKCS#12:

Vérifier que l'on a bien mis dans "p12.bat" les mêmes noms User name et CA NAME que dans le certificat d'utilisateur et celui de la CA. (Il faut laisser les guillemets).

En mode "ligne de commandes", sous Windows, dans une "fenêtre MS-DOS", exécuter "p12.bat". Répondre aux questions. En final on obtient:

"user.p12", certificat utilisateur avec clé privée et arborescence de la certification.

e) Importation dans Internet Explorer:

Depuis Explorer, double-cliquer sur "user.p12", laisser faire ensuite l'assistant d'importation.

Ne pas oublier de cocher:

activer la protection de clé renforcée,
rendre la clé privée exportable.

Et de choisir le mode "Haute sécurité" pour l'accès à votre certificat (par défaut, c'est "Sécurité moyenne").

Votre certificat de CA va se trouver ajouté à la liste des "Autorités Principales de Confiance". (Internet Explorer / click droit / Propriétés / Contenu / Certificats (cliquer deux fois) / onglet "Autorités Principales de Confiance")

Votre certificat d'utilisateur va se trouver ajouté dans la liste de vos certificats personnels (Internet Explorer / click droit / Propriétés / Contenu / Certificats (cliquer deux fois) / onglet "Certificats Personnels")

f) Utilisation dans Outlook Express:

Menu Outils / Comptes, sélectionner le compte ayant la même adresse e-mail que votre certificat d'utilisateur, puis Propriétés / Sécurité.

Choisir alors un certificat de signature et un de chiffrement (2 différents si vous en avez généré 2, sinon le même). Sélectionner le mode de chiffrement préférentiel (3DES).

g) Paramètres régionaux:

Démarrer / Paramètres / Panneau de Configuration / Paramètres régionaux, choisir "Français (Luxembourg)". Ceci est nécessaire pour enlever les limitations mises par

Microsoft sur les versions françaises d'Outlook Express. L'autre solution, permettant de garder les paramètres régionaux français, est d'exécuter le correctif "francfix.exe" de Microsoft. Il est possible que, pour les versions à venir d'Outlook Express, ceci ne soit, un jour, plus nécessaire. A vous de vérifier si votre version d'Outlook Express chiffre en 128 bits avec des paramètres régionaux français!

h) Utilisation dans d'autres navigateurs / logiciels de courrier électronique:

Suivre les instructions spécifiques de votre navigateur / logiciel de courrier électronique pour importer le certificat "user.p12". En particulier, si votre navigateur favori est le navigateur par défaut, double-cliquer sur "user.p12" devrait déclencher l'importation du certificat.

Pour une utilisation sûre du chiffrement S/MIME, je vous recommande l'utilisation de Mozilla, la version libre (OpenSource) de Netscape, qui est le seul navigateur / logiciel de courrier électronique pour lequel l'ensemble du code-source, y compris la partie cryptographique, est public. C'est le seul moyen de garantir l'absence de "trappes", en particulier dans la génération des clés de chiffrement. La version française de Mozilla se télécharge depuis:

<http://frenchmozilla.sourceforge.net/>.

3. Statut Légal:

L'utilisation de OpenSSL 0.9.6g est légale en France. La Free Software Foundation Europe Chapter France a obtenu de la DCSSI, en date du 15/07/02, une autorisation générale de fourniture en vue de l'utilisation et de l'importation de OpenSSL 0.9.6d et versions suivantes, et une dispense de licence des Douanes pour l'exportation.

Plus de détails sur le site de la FSFE France:

<http://france.fsfeurope.org/dcssi/dcssi.fr.html#dossiers> et

<http://france.fsfeurope.org/crypto/>.

Vous voilà prêt à signer des mails! Et à recevoir des mails chiffrés, après avoir préalablement envoyé un mail signé à votre correspondant (pour qu'il ait la partie publique de votre certificat).

Faites parvenir à vos correspondants une copie de votre "ca.crt". Une fois qu'ils l'auront installé dans leur magasin de certificats, ils accepteront tous les certificats venant de vous.

Ne dépréciez pas votre CA en signant n'importe quel certificat! Et ne dépréciez pas cet outil en créant de faux certificats de CA imitant ceux "des grands": de toute manière, la clé RSA associée ne sera pas identique... Votre faux ne résistera pas longtemps à l'analyse.

ANNEXE:

Les fichiers suivants sont à copier depuis cette page, coller dans un éditeur de texte, enregistrer et mettre dans votre répertoire OpenSSL.

ROOT-CA.BAT:

```
REM "root-ca.bat"
REM Usage: creer la CA
REM Creer la cle private de CA. A ne faire qu'une fois. Personnaliser la taille de
cle (1024, 2048)
openssl genrsa -des3 -out ca.key -rand seed.rnd 2048
REM Autosignature de la clé. Personnaliser la durée du certificat de CA (3650,
7300), en jours.
echo "Autosignature de la clé de CA"
openssl req -new -x509 -days 7300 -config root-ca.cnf -key ca.key -out ca.crt
```

ROOT-CA.CNF:

```
[ req ]
default_bits = 1024
default_keyfile = ca.key
distinguished_name = req_distinguished_name
x509_extensions = v3_ca
string_mask = nombstr
req_extensions = v3_req
[ req_distinguished_name ]
countryName = Country Name (2 letter code)
countryName_default = FR
countryName_min = 2
countryName_max = 2
stateOrProvinceName = State or Province Name (full name)
stateOrProvinceName_default = FRANCE
```

```
localityName = Locality Name (eg, city)
localityName_default = PARIS
0.organizationName = Organization Name (eg, company)
0.organizationName_default = Ma CA
organizationalUnitName = Organizational Unit Name (eg, section)
organizationalUnitName_default = Service de Certification
commonName = Common Name (eg, Mon Autorité de Certification)
commonName_max = 64
emailAddress = Email Address
emailAddress_max = 40
[ v3_ca ]
basicConstraints = critical,CA:true
subjectKeyIdentifier = hash
[ v3_req ]
nsCertType = objsign,email,server
```

USER-CERT.BAT

```
REM "user-cert.bat"
REM Usage: creer le formulaire de demande de certificat utilisateur.
REM Creation de la cle. A faire pour chaque certificat. Personnaliser la taille de
cle (1024, 2048).
openssl genrsa -out user.key -rand seed.rnd 2048
REM Remplir les donnees du certificat, nom d'utilisateur et adresse e-mail
echo "Remplir les donnees du certificat"
openssl req -new -config user-cert.cnf -key user.key -out user.csr
echo " "
echo "Vous pouvez maintenant lancer ca-sign.bat pour faire signer votre certificat"
echo " "
```

USER-CERT.CNF:

```
[ req ]
default_bits = 1024
default_keyfile = user.key
distinguished_name = req_distinguished_name
string_mask = nombstr
req_extensions = v3_req
[ req_distinguished_name ]
commonName = Common Name (eg, Jean MARTIN)
commonName_max = 64
emailAddress = Email Address
emailAddress_max = 40
[ v3_req ]
nsCertType = client,email
basicConstraints = critical,CA:false
```

CA-SIGN.BAT:

```

REM "ca-sign.bat"
REM Usage: signer un certificat d'utilisateur avec la cle de CA
REM Utiliser le mot de passe entre lors de la creation de CA
echo "signature par la CA: user.csr -> user.crt:"
openssl ca -config ca-sign.cnf -out user.crt -batch -infiles user.csr
echo "verification par la CA user.crt <-> CA cert"
openssl verify -CAfile ca.crt user.crt
CA-SIGN.CNF:
[ ca ]
default_ca = default_CA
[ default_CA ]
dir = .
certs = $dir
new_certs_dir = $dir/newcerts
database = index.txt
serial = serial.txt
RANDFILE = seed.rnd
certificate = ca.crt
private_key = ca.key
default_days = 3650
default_crl_days = 30
default_md = md5
preserve = yes
x509_extensions = user_cert
policy = policy_anything
[ policy_anything ]
commonName = supplied
emailAddress = supplied
[ user_cert ]
subjectAltName = email:copy
basicConstraints = critical,CA:false
authorityKeyIdentifier = keyid:always
extendedKeyUsage = clientAuth,emailProtection
P12.BAT:
REM "p12.bat"
REM Usage: rassemble les elements du certificat utilisateur et les met au format
PKCS#12
REM Remplacer Username et CA NAME par les noms que vous avez entres lors de la
creation des certificats user et CA
REM Laisser les guillemets autour des noms
openssl pkcs12 -export -in user.crt -inkey user.key -certfile ca.crt
-name "Username" -caname "CA NAME" -out user.p12
echo " "
echo "Votre certificat utilisateur a ete cree au format PCKS#12"
echo "Vous pouvez l'importer dans votre navigateur"
echo " "

```

Copyright WinTerMiNator 2000-2002.

Diffusé sous licence GNU Free Documentation

Mise à jour le 01/12/2002

--

@+

Ascadix

adresse @mail valide, mais ajoutez "sesame" dans l'objet pour que ça arrive.