OpenSSL - ca

Application CA minimaliste. Peut être utilisé pour signer des requêtes de certificats et générer des CRL. Maintient également une base des certificats délivrés et leur status.

Options CA

-config filename Spécifie le fichier de configuration à utiliser

-name section Spécifie la section du fichier de configuration à utiliser

-in filename Fichier source contenant une requête de certificat à signer

-ss_cert filename un certificat auto-signé à signer par la CA

-spkac filename Un fichier contenant une clé publique signée Netscape, un challenge et des valeurs de champs assitionnels à signer par la CA.

-infiles Si présent doit être la dernière option. Tous les arguments qui suivent sont traités comme des noms de fichier contenant des requêtes.

-out filename Fichier de sortie. (défaut : stdout)

-outdir directory Répertoire où placer les certificats de sortie. Le certificat sera nommé avec un numéro hexa et l'extension pem.

-cert Fichier du certificat CA

-keyfile filename La clé privée à utiliser pour signer la requête

-key password Le mot de passe à utiliser pour chiffrer la clé privée

-selfsign indique que les certificats délivrés doivent être signés avec la clé qui a servit à signer la requête (spécifié avec -keyfile). Les requêtes signées avec un certificat différent sont ignorés. si spkac, ss_cert ou gencrl sont spécifiés, cette option est ignorée.

-passin arg La source du mot de passe.

-verbose mode verbeux

-notext N'affiche pas la forme texte dans le fichier de sortie

-startdate date Permet de définir la date de début au format YYMMDDHHMMSSZ (structure ASN1 UTCTime)

-enddate date Permet de définir la date d'expiration au format YYMMDDHHMMSSZ (structure ASN1 UTCTime)

-days arg Le nombre de jours pendant lesquels certifier ce certificat

-md alg Le message digest à utiliser (md5 sha1 et mdc2)

-policy arg Spécifie la stratégie CA à utiliser. C'est une section dans le fichier de configuration

-msie_hack Permet un fonctionnemnt avec les très vieux contrôle d'enrollement de certificat IE 'certenr3'.

-preserveDN Normalement l'ordre du DN est le même que l'ordre des champs dans la section de stratégie. Avec cette option, l'ordre est le même que dans la requête

-noemailDN Le DN d'un certificat peut contenir le champ email s'il est présent dans le DN de la requête ; cependantil est préférable d'avoir l'email dans l'extnsion altName. Avec cette option, le champs email est supprimé du sujet du certificat et définis dans les extensions si présent.

-batch Dans ce mode tous les certificats sont certifiés automatiquement sans poser de questions.

-extensions section La section du fichier de configuration contenant les extensions à ajouter quand un certificat est délivré. Sans extention, un V1 est créé, si une extension est présente, même vide, un v3 est créé.

-extfile file Un fichier de configuration additionnel contenant les extensions à ajouter.

-engine id Spécifie le type de moteur. peut être définis à default pour tous les algorithmes disponibles

-subj arg Remplace le sujet dans la reque^te. doitêtre au format /type0=value0/type1=value1/type2=...

-utf8 Les valeurs dns les champs sont interprétés comme chaînes UTF-8 (par défaut en ASCII)

-multivalue-rdn l'argument -subj est interprété comme RDN multivalué. ex : /DC=org/DC=OpenSSL/DC=users/UID=123456+CN=John Doe (sans cette option : 123456+CN=John Doe)

Options CRL

-gencrl Génère une CRL basé sur les informations dans le fichier d'index

-crldays num Nombre de jours avant la prochaine CRL. valeur placée dans le champ nextUpdate de la CRL.

-crlhours num Nombre d'heure avant la prochaine CRL.

-revoke filename Un nom de fichier contenant un certificat à révoquer.

-crl_reason reason Raison dela révocation. Peut être (unspecified, keyCompromise, CACompromise, affiliationChanged, superseded, cessationOfOperation, certificateHold ou removeFromCRL). Définis une raison de révocation va créer une CRL v2.

-crl_hold instruction Définis la raison de révocation à certificateHold et l'instruction doit etre un OID (pet être à holdInstructioNone (=obsolete), holdInstructionCallIssuer ou HoldInstructionReject)

-crl_compromise time Définis la raison de révocation keyCompromise et le temps au format YYYYMMDDHHMMSSZ.

-crl_CA_compromise time Idem pour CACompromise

-crlexts section Section dans le fichier de configuration contenant les extensions CRL à inclure. Sans extension, une CRL v1 st générée.

Options du fichier de configuration

La section par défaut doit être nommée dans l'option default_ca de la section ca (ou de la section par défaut).

oid_file Spécifie le fichier contenant des oid additionnels, un par ligne.

oid_section Spécifie la section dans le fichier de configuration contenant les oid additionnels.

new_certs_dir Idem à -outdir

certificate idem à -cert

private_key idem à -keyfile

RANDFILE un fichier utilisé pour lire et écrire des informations de nombre aléatoire.

default_startdate idem à -startdate

default_enddate idem à -enddate

default_crl_hours idem à -crlhours

default_crl_days idem à -crldays

default_md idem à -md

database le fichier de données à utiliser

unique_subject à yes, les certificats doivent avoir un sujet unique

serial Fichier texte contenant le prochain numéro de CRL en hexa à utiliser.

x509_extensions idem à -extensions

crl_extensions idem à -crlexts

preserve idem à -preserveDN

email_in_dn idem à -noemailDN

msie_hack idem à -msie_hack

policy idem à -policy

name_opt, cert_opt Permettent au format utilisé d'afficher les détails du certificat lorsqu'il est demandé à l'utilisateur de confirmer la signature.

copy_extensions Détermine comment les extensions dans la requête sont manipulés. non spécifié ou à none, lesextensions sont ignorés. à copy, les extensions non encore présentes sont copiées dans le certificat. a copyall, toutes les extensions sont copiées en supprimant celles déjà présentes

Exemples

Signer une requête de certificat :
openssl ca -in req.pem -out newcert.pem
signer une requête e certificat, en utilisant les extensions de CA :
openssl ca -in req.pem -extensions v3_ca -out newcert.pem
Générer une CRL :
openssl ca -gencrl -out crl.pem
signer plusieurs requêtes :
openssl ca -infiles req1.pem req2.pem req3.pem
Certifier un SPKAC Netscape :
openssl ca -spkac spkac.txt
exemple de fichier SPKAC (tronquée pour plus de clarté) :
SPKAC=MIG0MGAwXDANBgkqhkiG9w0BAQEFAANLADBIAkEAn7PDhCeV/xIxUg8V70YRxK2A5
CN=Steve Test
emailAddress=steve@openssl.org
0.OU=OpenSSL Group
1.OU=Another Group

Exemple de section ca dans la configuration

```
[ ca ]
default_ca = CA_default # Section ca par défaut

[ CA_default ]
dir = ./demoCA # Répertoire de base
database = $dir/index.txt # fichier d'index
new_certs_dir = $dir/newcerts # Répertoire pour les nouveaux certificats délivrés

certificate = $dir/cacert.pem # Certificat racine
serial = $dir/serial # fichier serial
private_key = $dir/private/cakey.pem# Clé privée de la ca
RANDFILE = $dir/private/.rand # Fichier de nombres aléatoire

default_days = 365 # Durée de validité du certificat
default_crl_days= 30 # Durée de la CRL
default_md = md5 # md à utiliser

policy = policy_any # Stratégie par défaut
email_in_dn = no # Ne pas ajouter le mail dans le DN

name_opt = ca_default # Option d'affichage du nom du sujet
cert_opt = ca_default # Option d'affichage du certificat
copy_extensions = none # Ne pas copier les extensions depuis la requête

[ policy_any ]
countryName = supplied
stateOrProvinceName = optional
organizationName = optional
organizationalUnitName = optional
commonName = supplied
```

```
emailAddress = optional
```

Variables d'environnement

OPENSSL_CONF Réflète l'emplacement du fichier de configuration maître (idem à -config)