OpenSSL - req

Utilitaire pour créer et traiter des requêtes pkcs#10

OPTIONS

-inform PEM|DER Format du fichier d'entrée

-outform PEM|DER Format du fichier de sortie

-in filename Fichier d'entrée

-passin arg source du mot de passe du fichier d'entrée

-out filename Fichier de sortie

-passout arg Source du mot de passe pour le fichier de sortie

-text Affiche la requête en clair

-subject Affiche le sujet de la requête

-pubkey Affiche la clé publique

-noout N'affiche pas la version encodée de la requête

-modulus Affiche le modulo de la clé publique contenue dans la requête

-verify Vérifie la signature de la requête

-new Génère une nouvelle requête de certificat.

-subj arg Remplace le champ sujet dans la requête d'entrée doit être au format /type0=value0/type1=value1/typeN=valueN...

-rand file(s) Fichier(s) contenant les données aléatoire pour le générateur de nombre aléatoire.

-newkey arg Crée une nouvelle requête de certificat et une nouvelle clé privée. arg peut être sous la forme rsa :nbits où nbits est la longueur de la clé RSA. Tous les autres algorithmes supportent la forme alg :file où file est un fichier de paramètre. param :file utilise le fichier de paramètre ou le certificat spécifié par file. L'algorithme est déterminé par les paramètres.

-pkeyopt opt :value Définie une options d'algorithme. (Voir genpkey)

-key filename Spécifie le fichier contenant la clé privée à lire. Accepte le format PKCS#8.

-keyform PEM-DER Le format du fichier de clé privée.

-keyout filename Nom du fichier où écrire la clé privée.

-nodes Si cette option est spécifiée, la clé privée créée n'est pas chiffrée

-[digest] Spécifie le message digest à utiliser pour signer la requête.

-multivalue-rdn Permet d'interpréter -subj au format RDN multi-valué (exemple : /DC=org/DC=OpenSSL/DC=users/UID=123456+CN=John Doe, sinon le format sera : 123456+CN=John Doe)

-x509 Sort un certificat auto-signé au lieu d'une requête de certificat.

-days -n avec -x509, spécifie la durée de validité du certificat (défaut : 30jours)

-set_serial n Définis le numéro de série du certificat auto-signé. Peut-être en décimal ou en hexa.

-extensions section

-reqexts section Spécifient les sections alternatives à inclure dans la requête.

-utf8 interprète les valeurs de champs au format UTF8

-nameopt option détermine comment le sujet et l'issuer sont affichés (voir x509)

-repopt Personnalise le format de sortie avec -text. l'argument peut être une simple options ou plusieurs, séparés par des ',' (voir x509)

-asn1-kludge Par défaut, req sort les requêtes de certificat ne contenant pas d'attributs dans le format PKCS#10 correct. Cependant, certaines CA acceptent uniquement les requêtes ne contenant pas d'attributs dans une forme invalide : cette options produit ce format invalide (les attributs dans une requête PKCS#10 sont définis comme un set d'attribut SET OF. Ils ne sont pas optionnels donc si aucun attribut n'est présent, il devrait être encodé comme un SET OF vide, une forme invalide ne contient pas ce SET OF vide)

-no-asn1-kludge Inverse l'effet de -asn1-kludge

-newhdr Ajoute le mot NEW dans l'en-tête et pied de page PEM.

-batch Mode non-interactif

-verbose Affiche les détails sur l'opération courante

-engine id req va tenter d'obtenir une référence fonctionnelle du moteur spécifié

-keygen_engine id Spécifie un moteur qui devrait être utilisé pour les opérations de génération de clé.

Format du fichier de configuration

input_password output_password Les mots de passe pour les fichier de clé privée d'entrée et de sortie (remplace passin et passout)

default_keyfile Taille en bits de la clé (défaut 512, remplace -newkey)

oid_file Spécifie un fichier contenant des OID additionnels. Chaque ligne consiste de l'oid suivi d'un espace blanc, suivi du nom cours, suivi par un blanc et suivi par un nom long.

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

RANDFILE Spécifie un nom de fichier contenant les données aléatoires à utiliser par le moteur de génération de nombres pseudo-aléatoires

encrypt_key à no, la clé privée n'est pas chiffrée. (Remplace -nodes)

default_md Spécifie l'algorithme digest à utiliser. (md5, sha1 ou mdc2, défaut : md5)

string_mask Masque l'utilisation de certaines chaines dans certains champs. default utilise PrintableStrings, T61Strings et BMPStrings.

pkix utilise PrintableStrings et BMPStrings. utf8only utilise UTF8Strings. nombstr utilise PrintableStrings et T61Strings.

req_extensions Spécifie la section du fichier de configuration contenant la liste des extensions à ajouter à une requête (remplace -reqexts)

x509_extensions Spécifie la section du fichier de configuration contenant un liste des extensions à ajouter au certificat généré avec -x509 (remplace -extensions)

prompt à no désactive la demande des champs du certificat et prend les valeurs dans le fichier de configuration

utf8 à yes les valeurs de champs sont interprétés en utf8 (ASCII par défaut)

attributes Spécifie la section contenant les attributs de requête. Identique à distinguished_name. Actuellement ignoré par OpenSSL.

distinguished_name Spécifie la section contenant les champs dn à demander pour générer un certificat ou une requête.

Format des sections Distinguished Name et Attribute

Il y'a 2 formats pour ces sections. Si l'option prompt est à no, ces sections consistent de noms de champs et de valeur. Par exemple :

```
CN=My Name
OU=My Organization
emailAddress=someone@somewhere.org
```

Si prompt est absent ou à yes, ces sections contiennent la liste des champs à demander :

```
fieldName="prompt"
fieldName_default="default field value"
fieldName_min= 2
fieldName_max= 4
```

fieldname est le nom d'un champs, par exemple commonName ou CN. "prompt" est utilisé pour demander à l'utilisateur d'entrer les informations du champ. Si l'utilisateur n'entre rien, ce champ est omis.

- Si une valeur par défaut est définie, elle sera utilisée si l'utilisateur n'entre rien. L'utilisateur peut outrepasser cette valeur par défaut en entrant le caractère '.'
- Les limites min et max peuvent être utilisé pour limiter les valeurs de champs. Par exemple, countryName peut seulement avoir 2 caractères et doivent être correspondre à un PrintableString
- Certains champs comme organizationName peuvent être utilisé plus d'une fois dans un DN. Un second organizationName peut être entré en l'appelant 1.organizationName
- Les noms de champs permis sont des noms cours ou long d'object identifier. Incluant : commonName, countryName, localityName, organizationName, organizationUnitName, stateOrProvinceName. emailAddress, name, surname, givenName, dnQualifier.
- Des object identifier additionnels peuvent être définis avec les options oid_file et oid_section dans le fichier de configuration. Ces champs additionnels seront traités comme si c'était un DirectoryString

Exemples

Examiner et vérifier une requête de certificat :
openssl req -in req.pem -text -verify -noout
Créer une clé privée et générer une requête de certificat :
openssl genrsa -out key.pem 1024
openssl req -new -key key.pem -out req.pem
Idem en utilisant uniquement req :
openssl req -newkey rsa :1024 -keyout key.pem -out req.pem
Générer un certificat root auto-signé :
openssl req -x509 -newkey rsa :1024 -keyout key.pem -out req.pem
Exemple de fichier pointé par l'option oid_file :
1.2.3.4 shortName A longer Name
1.2.3.6 otherName Other longer Name
Exemple d'une section pointée par oid_section :
testoid1=1.2.3.5
testoid2=$testoid1.6

Exemple de fichier de configuration demandant les valeurs de champs

```
[ req ]
default_bits = 1024
default_keyfile = privkey.pem
distinguished_name = req_distinguished_name
attributes = req_attributes
x509_extensions = v3_ca

dirstring_type = nobmp

[ req_distinguished_name ]
countryName = Country Name (2 letter code)
countryName_default = AU
countryName_min = 2
countryName_max = 2

localityName = Locality Name (eg, city)

organizationalUnitName = Organizational Unit Name (eg, section)
```

```
commonName = Common Name (eg, YOUR name)
commonName_max = 64

emailAddress = Email Address
emailAddress_max = 40

[ req_attributes ]
challengePassword = A challenge password
challengePassword_min = 4
challengePassword_max = 20

[ v3_ca ]

subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid:always,issuer:always
basicConstraints = CA:true
```

Exemple de configuration contenant toutes les valeurs de champs

```
RANDFILE = $ENV ::HOME/.rnd

[ req ]
default_bits = 1024
default_keyfile = keyfile.pem
distinguished_name = req_distinguished_name
attributes = req_attributes
prompt = no
output_password = mypass

[ req_distinguished_name ]
C = GB
ST = Test State or Province
L = Test Locality
O = Organization Name
OU = Organizational Unit Name
CN = Common Name
emailAddress = test@email.address

[ req_attributes ]
challengePassword = A challenge password
```

Notes

Le format PEM inclus :

```
---BEGIN CERTIFICATE REQUEST---
---END CERTIFICATE REQUEST---
```

Certains logiciels comme Netscape certificate server nécessitent :

```
---BEGIN NEW CERTIFICATE REQUEST---
---END NEW CERTIFICATE REQUEST---
```

Variables d'environnement

OPENSSL_CONF peut être utilisé comme emplacement de fichier de configuration alternatif.