NAME

hx509 CA functions -

Functions

int **hx509_ca_tbs_init** (hx509_context context, hx509_ca_tbs *tbs)
void **hx509_ca_tbs_free** (hx509_ca_tbs *tbs)
int **hx509_ca_tbs_set_notBefore** (hx509_context context, hx509_ca_tbs tbs, time_t t)
int **hx509_ca_tbs_set_notAfter** (hx509_context context, hx509_ca_tbs tbs, time_t t)
int **hx509_ca_tbs_set_notAfter_lifetime** (hx509_context context, hx509_ca_tbs tbs, time_t delta)
struct units * **hx509_ca_tbs_template_units** (void)
int **hx509_ca_tbs_set_template** (hx509_context context, hx509_ca_tbs tbs, int flags, hx509_cert cert)
int **hx509_ca_tbs_set_ca** (hx509_context context, hx509_ca_tbs tbs, int pathLenConstraint)
int **hx509_ca_tbs_set_proxy** (hx509_context context, hx509_ca_tbs tbs, int pathLenConstraint)
int **hx509_ca_tbs_set_domaincontroller** (hx509_context context, hx509_ca_tbs tbs)
int **hx509_ca_tbs_set_spki** (hx509_context context, hx509_ca_tbs tbs, const SubjectPublicKeyInfo *spki)
int **hx509_ca_tbs_set_serialnumber** (hx509_context context, hx509_ca_tbs tbs, const heim_integer *serialNumber)
int **hx509_ca_tbs_add_eku** (hx509_context context, hx509_ca_tbs tbs, const heim_oid *oid)
int **hx509_ca_tbs_add_crl_dp_uri** (hx509_context context, hx509_ca_tbs tbs, const char *uri, hx509_name issuername)
int **hx509_ca_tbs_add_san_otherName** (hx509_context context, hx509_ca_tbs tbs, const heim_oid *oid, const heim_octet_string *os)
int **hx509_ca_tbs_add_san_pkinit** (hx509_context context, hx509_ca_tbs tbs, const char *principal)
int **hx509_ca_tbs_add_san_ms_upn** (hx509_context context, hx509_ca_tbs tbs, const char *principal)
int **hx509_ca_tbs_add_san_jid** (hx509_context context, hx509_ca_tbs tbs, const char *jid)
int **hx509_ca_tbs_add_san_hostname** (hx509_context context, hx509_ca_tbs tbs, const char *dnsname)
int **hx509_ca_tbs_add_san_rfc822name** (hx509_context context, hx509_ca_tbs tbs, const char *rfc822Name)
int **hx509_ca_tbs_set_subject** (hx509_context context, hx509_ca_tbs tbs, hx509_name subject)
int **hx509_ca_tbs_set_unique** (hx509_context context, hx509_ca_tbs tbs, const heim_bit_string *subjectUniqueID, const heim_bit_string *issuerUniqueID)
int **hx509_ca_tbs_subject_expand** (hx509_context context, hx509_ca_tbs tbs, hx509_env env)
int **hx509_ca_sign** (hx509_context context, hx509_ca_tbs tbs, hx509_cert signer, hx509_cert *certificate)
int **hx509_ca_sign_self** (hx509_context context, hx509_ca_tbs tbs, hx509_private_key signer, hx509_cert *certificate)

Detailed Description

See the **Hx509 CA functions** for description and examples.

Function Documentation

int hx509_ca_sign (hx509_context context, hx509_ca_tbs tbs, hx509_cert signer, hx509_cert * certificate)

Sign a to-be-signed certificate object with a issuer certificate.

The caller needs to at least have called the following functions on the to-be-signed certificate object:

- **hx509_ca_tbs_init()**
- **hx509_ca_tbs_set_subject()**
- **hx509_ca_tbs_set_spki()**

When done the to-be-signed certificate object should be freed with **hx509_ca_tbs_free()**.

When creating self-signed certificate use **hx509_ca_sign_self()** instead.

Parameters:

context A hx509 context.
tbs object to be signed.
signer the CA certificate object to sign with (need private key).
certificate return cerificate, free with **hx509_cert_free()**.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_sign_self (hx509_context context, hx509_ca_tbs tbs, hx509_private_key signer, hx509_cert * certificate)

Work just like **hx509_ca_sign()** but signs it-self.

Parameters:

context A hx509 context.
tbs object to be signed.
signer private key to sign with.
certificate return cerificate, free with **hx509_cert_free()**.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_add_crl_dp_uri (hx509_context context, hx509_ca_tbs tbs, const char * uri, hx509_name issuername)

Add CRL distribution point URI to the to-be-signed certificate object.

Parameters:

context A hx509 context.
tbs object to be signed.
uri uri to the CRL.
issuername name of the issuer.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

issuername not supported

int hx509_ca_tbs_add_eku (hx509_context context, hx509_ca_tbs tbs, const heim_oid * oid)

An an extended key usage to the to-be-signed certificate object. Duplicates will detected and not added.

Parameters:

context A hx509 context.
tbs object to be signed.
oid extended key usage to add.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_add_san_hostname (hx509_context context, hx509_ca_tbs tbs, const char * dnsname)

Add a Subject Alternative Name hostname to to-be-signed certificate object. A domain match starts with ., an exact match does not.

Example of a an domain match: .domain.se matches the hostname host.domain.se.

Parameters:

context A hx509 context.
tbs object to be signed.
dnsname a hostame.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_add_san_jid (hx509_context context, hx509_ca_tbs tbs, const char * jid)

Add a Jabber/XMPP jid Subject Alternative Name to the to-be-signed certificate object. The jid is an

UTF8 string.

Parameters:

context A hx509 context.
tbs object to be signed.
jid string of an a jabber id in UTF8.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_add_san_ms_upn (hx509_context context, hx509_ca_tbs tbs, const char * principal)

Add Microsoft UPN Subject Alternative Name to the to-be-signed certificate object. The principal string is a UTF8 string.

Parameters:

context A hx509 context.
tbs object to be signed.
principal Microsoft UPN string.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_add_san_otherName (hx509_context context, hx509_ca_tbs tbs, const heim_oid * oid, const heim_octet_string * os)

Add Subject Alternative Name otherName to the to-be-signed certificate object.

Parameters:

context A hx509 context.
tbs object to be signed.
oid the oid of the OtherName.
os data in the other name.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_add_san_pkinit (hx509_context context, hx509_ca_tbs tbs, const char * principal)

Add Kerberos Subject Alternative Name to the to-be-signed certificate object. The principal string is a UTF8 string.

Parameters:

context A hx509 context.
tbs object to be signed.
principal Kerberos principal to add to the certificate.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_add_san_rfc822name (hx509_context context, hx509_ca_tbs tbs, const char * rfc822Name)
Add a Subject Alternative Name rfc822 (email address) to to-be-signed certificate object.

Parameters:
context A hx509 context.
tbs object to be signed.
rfc822Name a string to a email address.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

void hx509_ca_tbs_free (hx509_ca_tbs * tbs)
Free an To Be Signed object.

Parameters:
tbs object to free.

int hx509_ca_tbs_init (hx509_context context, hx509_ca_tbs * tbs)
Allocate an to-be-signed certificate object that will be converted into an certificate.

Parameters:
context A hx509 context.
tbs returned to-be-signed certicate object, free with **hx509_ca_tbs_free()**.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_ca (hx509_context context, hx509_ca_tbs tbs, int pathLenConstraint)
Make the to-be-signed certificate object a CA certificate. If the pathLenConstraint is negative path length constraint is used.

Parameters:

context A hx509 context.
tbs object to be signed.
pathLenConstraint path length constraint, negative, no constraint.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_domaincontroller (hx509_context context, hx509_ca_tbs tbs)
Make the to-be-signed certificate object a windows domain controller certificate.

Parameters:
context A hx509 context.
tbs object to be signed.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_notAfter (hx509_context context, hx509_ca_tbs tbs, time_t t)
Set the absolute time when the certificate is valid to.

Parameters:
context A hx509 context.
tbs object to be signed.
t time when the certificate will expire

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_notAfter_lifetime (hx509_context context, hx509_ca_tbs tbs, time_t delta)
Set the relative time when the certificiate is going to expire.

Parameters:
context A hx509 context.
tbs object to be signed.
delta seconds to the certificate is going to expire.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_notBefore (hx509_context context, hx509_ca_tbs tbs, time_t t)

Set the absolute time when the certificate is valid from. If not set the current time will be used.

Parameters:

context A hx509 context.
tbs object to be signed.
t time the certificated will start to be valid

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_proxy (hx509_context context, hx509_ca_tbs tbs, int pathLenConstraint)

Make the to-be-signed certificate object a proxy certificate. If the pathLenConstraint is negative path length constraint is used.

Parameters:

context A hx509 context.
tbs object to be signed.
pathLenConstraint path length constraint, negative, no constraint.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_serialnumber (hx509_context context, hx509_ca_tbs tbs, const heim_integer * serialNumber)

Set the serial number to use for to-be-signed certificate object.

Parameters:

context A hx509 context.
tbs object to be signed.
serialNumber serial number to use for the to-be-signed certificate object.

Returns:

An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_spki (hx509_context context, hx509_ca_tbs tbs, const SubjectPublicKeyInfo * spki)

Set the subject public key info (SPKI) in the to-be-signed certificate object. SPKI is the public key and key related parameters in the certificate.

Parameters:

context A hx509 context.
tbs object to be signed.
spki subject public key info to use for the to-be-signed certificate object.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_subject (hx509_context context, hx509_ca_tbs tbs, hx509_name subject)
Set the subject name of a to-be-signed certificate object.

Parameters:
context A hx509 context.
tbs object to be signed.
subject the name to set a subject.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_template (hx509_context context, hx509_ca_tbs tbs, int flags, hx509_cert cert)
Initialize the to-be-signed certificate object from a template certifiate.

Parameters:
context A hx509 context.
tbs object to be signed.
flags bit field selecting what to copy from the template certifiate.
cert template certificate.

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_set_unique (hx509_context context, hx509_ca_tbs tbs, const heim_bit_string * subjectUniqueID, const heim_bit_string * issuerUniqueID)
Set the issuerUniqueID and subjectUniqueID

These are only supposed to be used considered with version 2 certificates, replaced by the two extensions SubjectKeyIdentifier and IssuerKeyIdentifier. This function is to allow application using legacy protocol to issue them.

Parameters:
context A hx509 context.

tbs object to be signed.
issuerUniqueID to be set
subjectUniqueID to be set

Returns:
An hx509 error code, see **hx509_get_error_string()**.

int hx509_ca_tbs_subject_expand (hx509_context context, hx509_ca_tbs tbs, hx509_env env)
Expand the the subject name in the to-be-signed certificate object using **hx509_name_expand()**.

Parameters:
context A hx509 context.
tbs object to be signed.
env enviroment variable to expand variables in the subject name, see hx509_env_init().

Returns:
An hx509 error code, see **hx509_get_error_string()**.

struct units* hx509_ca_tbs_template_units (void) [read]
Make of template units, use to build flags argument to **hx509_ca_tbs_set_template()** with parse_units().

Returns:
an units structure.