NAME

hx509 certificate store functions -

Functions

int **hx509_certs_init** (hx509_context context, const char *name, int flags, hx509_lock lock, hx509_certs *certs)
int **hx509_certs_store** (hx509_context context, hx509_certs certs, int flags, hx509_lock lock)
void **hx509_certs_free** (hx509_certs *certs)
int **hx509_certs_start_seq** (hx509_context context, hx509_certs certs, hx509_cursor *cursor)
int **hx509_certs_next_cert** (hx509_context context, hx509_certs certs, hx509_cursor cursor, hx509_cert *cert)
int **hx509_certs_end_seq** (hx509_context context, hx509_certs certs, hx509_cursor cursor)
int **hx509_certs_iter_f** (hx509_context context, hx509_certs certs, int(*func)(hx509_context, void *, hx509_cert), void *ctx)
int **hx509_ci_print_names** (hx509_context context, void *ctx, hx509_cert c)
int **hx509_certs_add** (hx509_context context, hx509_certs certs, hx509_cert cert)
int **hx509_certs_find** (hx509_context context, hx509_certs certs, const hx509_query *q, hx509_cert *r)
int **hx509_certs_filter** (hx509_context context, hx509_certs certs, const hx509_query *q, hx509_certs *result)
int **hx509_certs_merge** (hx509_context context, hx509_certs to, hx509_certs from)
int **hx509_certs_append** (hx509_context context, hx509_certs to, hx509_lock lock, const char *name)
int **hx509_get_one_cert** (hx509_context context, hx509_certs certs, hx509_cert *c)
int **hx509_certs_info** (hx509_context context, hx509_certs certs, int(*func)(void *, const char *), void *ctx)

Detailed Description

See the **Certificate store operations** for description and examples.

Function Documentation

int hx509_certs_add (hx509_context context, hx509_certs certs, hx509_cert cert)

Add a certificate to the certificiate store.

The receiving keyset certs will either increase reference counter of the cert or make a deep copy, either way, the caller needs to free the cert itself.

Parameters:

context a hx509 context.
certs certificate store to add the certificate to.
cert certificate to add.

Returns:

Returns an hx509 error code.

int hx509_certs_append (hx509_context context, hx509_certs to, hx509_lock lock, const char * name)

Same a **hx509_certs_merge()** but use a lock and name to describe the from source.

Parameters:

context a hx509 context.
to the store to merge into.
lock a lock that unlocks the certificates store, use NULL to select no password/certifictes/prompt lock (see **Locking and unlocking certificates and encrypted data.**).
name name of the source store

Returns:

Returns an hx509 error code.

int hx509_certs_end_seq (hx509_context context, hx509_certs certs, hx509_cursor cursor)

End the iteration over certificates.

Parameters:

context a hx509 context.
certs certificate store to iterate over.
cursor cursor that will keep track of progress, freed.

Returns:

Returns an hx509 error code.

int hx509_certs_filter (hx509_context context, hx509_certs certs, const hx509_query * q, hx509_certs * result)

Filter certificate matching the query.

Parameters:

context a hx509 context.
certs certificate store to search.
q query allocated with **hx509 query functions** functions.
result the filtered certificate store, caller must free with **hx509_certs_free()**.

Returns:

Returns an hx509 error code.

Return HX509_CERT_NOT_FOUND if no certificate in certs matched the query.

int hx509_certs_find (hx509_context context, hx509_certs certs, const hx509_query * q, hx509_cert * r)

Find a certificate matching the query.

Parameters:

context a hx509 context.
certs certificate store to search.
q query allocated with **hx509 query functions** functions.
r return certificate (or NULL on error), should be freed with **hx509_cert_free()**.

Returns:

Returns an hx509 error code.

Return HX509_CERT_NOT_FOUND if no certificate in certs matched the query.

void hx509_certs_free (hx509_certs * certs)

Free a certificate store.

Parameters:

certs certificate store to free.

int hx509_certs_info (hx509_context context, hx509_certs certs, int(*)(void *, const char *) func, void * ctx)

Print some info about the certificate store.

Parameters:

context a hx509 context.
certs certificate store to print information about.
func function that will get each line of the information, if NULL is used the data is printed on a FILE descriptor that should be passed in ctx, if ctx also is NULL, stdout is used.
ctx parameter to func.

Returns:

Returns an hx509 error code.

int hx509_certs_init (hx509_context context, const char * name, int flags, hx509_lock lock, hx509_certs * certs)

Open or creates a new hx509 certificate store.

Parameters:

context A hx509 context

name name of the store, format is TYPE:type-specific-string, if NULL is used the MEMORY store is used.
flags list of flags:

- HX509_CERTS_CREATE create a new keystore of the specific TYPE.

- HX509_CERTS_UNPROTECT_ALL fails if any private key failed to be extracted.

lock a lock that unlocks the certificates store, use NULL to select no password/certifictes/prompt lock (see **Locking and unlocking certificates and encrypted data.**).
certs return pointer, free with **hx509_certs_free()**.

int hx509_certs_iter_f (hx509_context context, hx509_certs certs, int(*)(hx509_context, void *, hx509_cert) func, void * ctx)

Iterate over all certificates in a keystore and call an function for each fo them.

Parameters:

context a hx509 context.
certs certificate store to iterate over.
func function to call for each certificate. The function should return non-zero to abort the iteration, that value is passed back to the caller of **hx509_certs_iter_f()**.
ctx context variable that will passed to the function.

Returns:

Returns an hx509 error code.

int hx509_certs_merge (hx509_context context, hx509_certs to, hx509_certs from)

Merge a certificate store into another. The from store is keep intact.

Parameters:

context a hx509 context.
to the store to merge into.
from the store to copy the object from.

Returns:

Returns an hx509 error code.

int hx509_certs_next_cert (hx509_context context, hx509_certs certs, hx509_cursor cursor, hx509_cert * cert)

Get next ceritificate from the certificate keystore pointed out by cursor.

Parameters:

context a hx509 context.
certs certificate store to iterate over.
cursor cursor that keeps track of progress.
cert return certificate next in store, NULL if the store contains no more certificates. Free with **hx509_cert_free()**.

Returns:

Returns an hx509 error code.

int hx509_certs_start_seq (hx509_context context, hx509_certs certs, hx509_cursor * cursor)

Start the integration

Parameters:

context a hx509 context.
certs certificate store to iterate over
cursor cursor that will keep track of progress, free with **hx509_certs_end_seq()**.

Returns:

Returns an hx509 error code. HX509_UNSUPPORTED_OPERATION is returned if the certificate store doesn't support the iteration operation.

int hx509_certs_store (hx509_context context, hx509_certs certs, int flags, hx509_lock lock)

Write the certificate store to stable storage.

Parameters:

context A hx509 context.
certs a certificate store to store.
flags currently unused, use 0.
lock a lock that unlocks the certificates store, use NULL to select no password/certifictes/prompt lock (see **Locking and unlocking certificates and encrypted data.**).

Returns:

Returns an hx509 error code. HX509_UNSUPPORTED_OPERATION if the certificate store doesn't support the store operation.

int hx509_ci_print_names (hx509_context context, void * ctx, hx509_cert c)

Iterate over all certificates in a keystore and call an function for each fo them.

Parameters:

context a hx509 context.
certs certificate store to iterate over.
func function to call for each certificate. The function should return non-zero to abort the iteration, that value is passed back to the caller of hx509_certs_iter().

Returns:
Returns an hx509 error code. Function to use to **hx509_certs_iter_f()** as a function argument, the ctx variable to **hx509_certs_iter_f()** should be a FILE file descriptor.

Parameters:
context a hx509 context.
ctx used by **hx509_certs_iter_f()**.
c a certificate

Returns:
Returns an hx509 error code.

int hx509_get_one_cert (hx509_context context, hx509_certs certs, hx509_cert * c)
Get one random certificate from the certificate store.

Parameters:
context a hx509 context.
certs a certificate store to get the certificate from.
c return certificate, should be freed with **hx509_cert_free()**.

Returns:
Returns an hx509 error code.