NAME

krb5 - Heimdal Kerberos 5 library

SYNOPSIS

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_add_et_list** (krb5_context context, void(*func)(struct et_list **))

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_password** (krb5_context context, krb5_creds *creds, const char *newpw, krb5_principal targprinc, int *result_code, krb5_data *result_code_string, krb5_data *result_string)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_init_context** (krb5_context *context)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_copy_context** (krb5_context context, krb5_context *out)

KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_free_context** (krb5_context context)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_config_files** (krb5_context context, char **filenames)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_prepend_config_files_default** (const char *filelist, char ***pfilenames)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_get_default_config_files** (char ***pfilenames)

KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_free_config_files** (char **filenames)

KRB5_LIB_FUNCTION const krb5_enctype *KRB5_LIB_CALL **krb5_kerberos_enctypes** (krb5_context context)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_default_in_tkt_etypes** (krb5_context context, const krb5_enctype *etypes)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_get_default_in_tkt_etypes** (krb5_context context, krb5_pdu pdu_type, krb5_enctype **etypes)

KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_init_ets** (krb5_context context)

KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_set_use_admin_kdc** (krb5_context context, krb5_boolean flag)

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL **krb5_get_use_admin_kdc** (krb5_context context)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_add_extra_addresses** (krb5_context context, krb5_addresses *addresses)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_extra_addresses** (krb5_context context, const krb5_addresses *addresses)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_get_extra_addresses** (krb5_context context, krb5_addresses *addresses)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_add_ignore_addresses** (krb5_context context, krb5_addresses *addresses)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_ignore_addresses** (krb5_context

context, const krb5_addresses *addresses)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_get_ignore_addresses** (krb5_context context, krb5_addresses *addresses)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_fcache_version** (krb5_context context, int version)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_get_fcache_version** (krb5_context context, int *version)
KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL **krb5_is_thread_safe** (void)
KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_set_dns_canonicalize_hostname** (krb5_context context, krb5_boolean flag)
KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL **krb5_get_dns_canonicalize_hostname** (krb5_context context)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_get_kdc_sec_offset** (krb5_context context, int32_t *sec, int32_t *usec)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_kdc_sec_offset** (krb5_context context, int32_t sec, int32_t usec)
KRB5_LIB_FUNCTION time_t KRB5_LIB_CALL **krb5_get_max_time_skew** (krb5_context context)
KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_set_max_time_skew** (krb5_context context, time_t t)
KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL **krb5_set_home_dir_access** (krb5_context context, krb5_boolean allow)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_copy_host_realm** (krb5_context context, const krb5_realm *from, krb5_realm **to)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_free_cred_contents** (krb5_context context, krb5_creds *c)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_copy_creds_contents** (krb5_context context, const krb5_creds *incred, krb5_creds *c)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_copy_creds** (krb5_context context, const krb5_creds *incred, krb5_creds **outcred)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_free_creds** (krb5_context context, krb5_creds *c)
KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL **krb5_compare_creds** (krb5_context context, krb5_flags whichfields, const krb5_creds *mcreds, const krb5_creds *creds)
KRB5_LIB_FUNCTION unsigned long KRB5_LIB_CALL **krb5_creds_get_ticket_flags** (krb5_creds *creds)
KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_data_zero** (krb5_data *p)
KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_data_free** (krb5_data *p)
KRB5_LIB_FUNCTION void KRB5_LIB_CALL **krb5_free_data** (krb5_context context, krb5_data *p)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_data_alloc** (krb5_data *p, int len)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_data_realloc** (krb5_data *p, int len)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_data_copy** (krb5_data *p, const void *data, size_t len)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_copy_data** (krb5_context context, const krb5_data *indata, krb5_data **outdata)
KRB5_LIB_FUNCTION int KRB5_LIB_CALL **krb5_data_cmp** (const krb5_data *data1, const krb5_data *data2)
KRB5_LIB_FUNCTION int KRB5_LIB_CALL **krb5_data_ct_cmp** (const krb5_data *data1, const krb5_data *data2)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_krbhst_get_addrinfo** (krb5_context context, krb5_krbhst_info *host, struct addrinfo **ai)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_free_ticket** (krb5_context context, krb5_ticket *ticket)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_copy_ticket** (krb5_context context, const krb5_ticket *from, krb5_ticket **to)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_ticket_get_client** (krb5_context context, const krb5_ticket *ticket, krb5_principal *client)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_ticket_get_server** (krb5_context context, const krb5_ticket *ticket, krb5_principal *server)
KRB5_LIB_FUNCTION time_t KRB5_LIB_CALL **krb5_ticket_get_endtime** (krb5_context context, const krb5_ticket *ticket)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_ticket_get_authorization_data_type** (krb5_context context, krb5_ticket *ticket, int type, krb5_data *data)
KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_set_real_time** (krb5_context context, krb5_timestamp sec, int32_t usec)

Detailed Description

Function Documentation

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_add_et_list (krb5_context context, void(*)(struct et_list **) func)

Add a specified list of error messages to the et list in context. Call func (probably a comerr-generated function) with a pointer to the current et_list.

Parameters:

context A kerberos context.
func The generated com_err et function.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_add_extra_addresses (krb5_context context, krb5_addresses * addresses)
Add extra address to the address list that the library will add to the client's address list when communicating with the KDC.

Parameters:
context Kerberos 5 context.
addresses addreses to add

Returns:
Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_add_ignore_addresses (krb5_context context, krb5_addresses * addresses)
Add extra addresses to ignore when fetching addresses from the underlaying operating system.

Parameters:
context Kerberos 5 context.
addresses addreses to ignore

Returns:
Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL krb5_compare_creds (krb5_context context, krb5_flags whichfields, const krb5_creds * mcreds, const krb5_creds * creds)
Return TRUE if 'mcreds' and 'creds' are equal ('whichfields' determines what equal means).

The following flags, set in whichfields affects the comparison:

- KRB5_TC_MATCH_SRV_NAMEONLY Consider all realms equal when comparing the service principal.

- KRB5_TC_MATCH_KEYTYPE Compare enctypes.

- KRB5_TC_MATCH_FLAGS_EXACT Make sure that the ticket flags are identical.

- KRB5_TC_MATCH_FLAGS Make sure that all ticket flags set in mcreds are also present in creds .

- KRB5_TC_MATCH_TIMES_EXACT Compares the ticket times exactly.
- KRB5_TC_MATCH_TIMES Compares only the expiration times of the creds.
- KRB5_TC_MATCH_AUTHDATA Compares the authdata fields.
- KRB5_TC_MATCH_2ND_TKT Compares the second tickets (used by user-to-user authentication).
- KRB5_TC_MATCH_IS_SKEY Compares the existance of the second ticket.

Parameters:

context Kerberos 5 context.
whichfields which fields to compare.
mcreds cred to compare with.
creds cred to compare with.

Returns:

return TRUE if mcred and creds are equal, FALSE if not.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_copy_context (krb5_context context, krb5_context * out)

Make a copy for the Kerberos 5 context, the new krb5_context shoud be freed with **krb5_free_context()**.

Parameters:

context the Kerberos context to copy
out the copy of the Kerberos, set to NULL error.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_copy_creds (krb5_context context, const krb5_creds * incred, krb5_creds ** outcred)

Copy krb5_creds.

Parameters:

context Kerberos 5 context.
incred source credential
outcred destination credential, free with **krb5_free_creds()**.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_copy_creds_contents (krb5_context context, const krb5_creds * incred, krb5_creds * c)

Copy content of krb5_creds.

Parameters:

context Kerberos 5 context.
incred source credential
c destination credential, free with **krb5_free_cred_contents()**.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_copy_data (krb5_context context, const krb5_data * indata, krb5_data ** outdata)

Copy the data into a newly allocated krb5_data.

Parameters:

context Kerberos 5 context.
indata the krb5_data data to copy
outdata new krb5_date to copy too. Free with **krb5_free_data()**.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_copy_host_realm (krb5_context context, const krb5_realm * from, krb5_realm ** to)

Copy the list of realms from ‘from’ to ‘to’.

Parameters:

context Kerberos 5 context.
from list of realms to copy from.
to list of realms to copy to, free list of **krb5_free_host_realm()**.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_copy_ticket (krb5_context context, const krb5_ticket * from, krb5_ticket ** to)

Copy ticket and content

Parameters:

context a Kerberos 5 context
from ticket to copy
to new copy of ticket, free with **krb5_free_ticket()**

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION unsigned long KRB5_LIB_CALL krb5_creds_get_ticket_flags (krb5_creds * creds)

Returns the ticket flags for the credentials in creds. See also **krb5_ticket_get_flags()**.

Parameters:

creds credential to get ticket flags from

Returns:

ticket flags

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_data_alloc (krb5_data * p, int len)

Allocate data of and krb5_data.

Parameters:

p krb5_data to allocate.
len size to allocate.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned.

KRB5_LIB_FUNCTION int KRB5_LIB_CALL krb5_data_cmp (const krb5_data * data1, const krb5_data * data2)

Compare to data.

Parameters:

data1 krb5_data to compare
data2 krb5_data to compare

Returns:
return the same way as memcmp(), useful when sorting.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_data_copy (krb5_data * p, const void * data, size_t len)
Copy the data of len into the krb5_data.

Parameters:
p krb5_data to copy into.
data data to copy..
len new size.

Returns:
Returns 0 to indicate success. Otherwise an kerberos et error code is returned.

KRB5_LIB_FUNCTION int KRB5_LIB_CALL krb5_data_ct_cmp (const krb5_data * data1, const krb5_data * data2)
Compare to data not exposing timing information from the checksum data

Parameters:
data1 krb5_data to compare
data2 krb5_data to compare

Returns:
returns zero for same data, otherwise non zero.

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_data_free (krb5_data * p)
Free the content of krb5_data structure, its ok to free a zeroed structure (with memset() or **krb5_data_zero()**). When done, the structure will be zeroed. The same function is called **krb5_free_data_contents()** in MIT Kerberos.

Parameters:
p krb5_data to free.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_data_realloc (krb5_data * p, int len)
Grow (or shrink) the content of krb5_data to a new size.

Parameters:

p krb5_data to free.
len new size.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned.

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_data_zero (krb5_data * p)

Reset the (potentially uninitalized) krb5_data structure.

Parameters:

p krb5_data to reset.

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_free_config_files (char ** filenames)

Free a list of configuration files.

Parameters:

filenames list, terminated with a NULL pointer, to be freed. NULL is an valid argument.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_free_context (krb5_context context)

Frees the krb5_context allocated by **krb5_init_context()**.

Parameters:

context context to be freed.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_free_cred_contents (krb5_context context, krb5_creds * c)

Free content of krb5_creds.

Parameters:

context Kerberos 5 context.
c krb5_creds to free.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_free_creds (krb5_context context, krb5_creds * c)
Free krb5_creds.

Parameters:
context Kerberos 5 context.
c krb5_creds to free.

Returns:
Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_free_data (krb5_context context, krb5_data * p)
Free krb5_data (and its content).

Parameters:
context Kerberos 5 context.
p krb5_data to free.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_free_ticket (krb5_context context, krb5_ticket * ticket)
Free ticket and content

Parameters:
context a Kerberos 5 context
ticket ticket to free

Returns:
Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_get_default_config_files (char * pfilenames)**
Get the global configuration list.

Parameters:
pfilenames return array of filenames, should be freed with **krb5_free_config_files()**.

Returns:
Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see

krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_get_default_in_tkt_etypes (krb5_context context, krb5_pdu pdu_type, krb5_enctype ** etypes)

Get the default encryption types that will be use in communcation with the KDC, clients and servers.

Parameters:

context Kerberos 5 context.
etypes Encryption types, array terminated with ETYPE_NULL(0), caller should free array with krb5_xfree():

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL krb5_get_dns_canonicalize_hostname (krb5_context context)

Get if the library uses DNS to canonicalize hostnames.

Parameters:

context Kerberos 5 context.

Returns:

return non zero if the library uses DNS to canonicalize hostnames.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_get_extra_addresses (krb5_context context, krb5_addresses * addresses)

Get extra address to the address list that the library will add to the client's address list when communicating with the KDC.

Parameters:

context Kerberos 5 context.
addresses addreses to set

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_get_fcache_version (krb5_context context, int * version)

Get version of fcache that the library should use.

Parameters:

context Kerberos 5 context.
version version number.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_get_ignore_addresses (krb5_context context, krb5_addresses * addresses)
Get extra addresses to ignore when fetching addresses from the underlaying operating system.

Parameters:

context Kerberos 5 context.
addresses list addreses ignored

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_get_kdc_sec_offset (krb5_context context, int32_t * sec, int32_t * usec)
Get current offset in time to the KDC.

Parameters:

context Kerberos 5 context.
sec seconds part of offset.
usec micro seconds part of offset.

Returns:

returns zero

KRB5_LIB_FUNCTION time_t KRB5_LIB_CALL krb5_get_max_time_skew (krb5_context context)
Get max time skew allowed.

Parameters:

context Kerberos 5 context.

Returns:

timeskew in seconds.

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL krb5_get_use_admin_kdc (krb5_context context)

Make the kerberos library default to the admin KDC.

Parameters:

context Kerberos 5 context.

Returns:

boolean flag to telling the context will use admin KDC as the default KDC.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_init_context (krb5_context * context)

Initializes the context structure and reads the configuration file /etc/krb5.conf. The structure should be freed by calling **krb5_free_context()** when it is no longer being used.

Parameters:

context pointer to returned context

Returns:

Returns 0 to indicate success. Otherwise an errno code is returned. Failure means either that something bad happened during initialization (typically ENOMEM) or that Kerberos should not be used ENXIO.

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_init_ets (krb5_context context)

Init the built-in ets in the Kerberos library.

Parameters:

context kerberos context to add the ets too

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL krb5_is_thread_safe (void)

Runtime check if the Kerberos library was complied with thread support.

Returns:

TRUE if the library was compiled with thread support, FALSE if not.

KRB5_LIB_FUNCTION const krb5_enctype* KRB5_LIB_CALL krb5_kerberos_enctypes (krb5_context context)

Returns the list of Kerberos encryption types sorted in order of most preferred to least preferred encryption type. Note that some encryption types might be disabled, so you need to check with **krb5_enctype_valid()** before using the encryption type.

Returns:

list of enctypes, terminated with ETYPE_NULL. Its a static array completed into the Kerberos library so the content doesn't need to be freed.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_krbhst_get_addrinfo (krb5_context context, krb5_krbhst_info * host, struct addrinfo ** ai)

Return an 'struct addrinfo *' for a KDC host.

Returns an the struct addrinfo in in that corresponds to the information in 'host'. free:ing is handled by krb5_krbhst_free, so the returned ai must not be released.

First try this as an IP address, this allows us to add a dot at the end to stop using the search domains.

If the hostname contains a dot, assumes it's a FQDN and don't use search domains since that might be painfully slow when machine is disconnected from that network.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_prepend_config_files_default (const char * filelist, char * pfilenames)**

Prepend the filename to the global configuration list.

Parameters:

filelist a filename to add to the default list of filename

pfilenames return array of filenames, should be freed with **krb5_free_config_files()**.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_config_files (krb5_context context, char ** filenames)

Reinit the context from a new set of filenames.

Parameters:

context context to add configuration too.

filenames array of filenames, end of list is indicated with a NULL filename.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_default_in_tkt_etypes (krb5_context context, const krb5_enctype * etypes)

Set the default encryption types that will be use in communcation with the KDC, clients and servers.

Parameters:

context Kerberos 5 context.
etypes Encryption types, array terminated with ETYPE_NULL (0).

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_set_dns_canonicalize_hostname (krb5_context context, krb5_boolean flag)

Set if the library should use DNS to canonicalize hostnames.

Parameters:

context Kerberos 5 context.
flag if its dns canonicalizion is used or not.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_extra_addresses (krb5_context context, const krb5_addresses * addresses)

Set extra address to the address list that the library will add to the client's address list when communicating with the KDC.

Parameters:

context Kerberos 5 context.
addresses addreses to set

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_fcache_version (krb5_context context, int version)

Set version of fcache that the library should use.

Parameters:

context Kerberos 5 context.
version version number.

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL krb5_set_home_dir_access (krb5_context context, krb5_boolean allow)

Enable and disable home directory access on either the global state or the krb5_context state. By calling **krb5_set_home_dir_access()** with context set to NULL, the global state is configured otherwise the state for the krb5_context is modified.

For home directory access to be allowed, both the global state and the krb5_context state have to be allowed.

Administrator (root user), never uses the home directory.

Parameters:

context a Kerberos 5 context or NULL
allow allow if TRUE home directory

Returns:

the old value

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_ignore_addresses (krb5_context context, const krb5_addresses * addresses)

Set extra addresses to ignore when fetching addresses from the underlaying operating system.

Parameters:

context Kerberos 5 context.
addresses addreses to ignore

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_kdc_sec_offset (krb5_context context, int32_t sec, int32_t usec)

Set current offset in time to the KDC.

Parameters:

context Kerberos 5 context.
sec seconds part of offset.
usec micro seconds part of offset.

Returns:

returns zero

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_set_max_time_skew (krb5_context context, time_t t)
Set max time skew allowed.

Parameters:

context Kerberos 5 context.
t timeskew in seconds.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_password (krb5_context context, krb5_creds * creds, const char * newpw, krb5_principal targprinc, int * result_code, krb5_data * result_code_string, krb5_data * result_string)
Change password using creds.

Parameters:

context a Keberos context
creds The initial kadmin/passwd for the principal or an admin principal
newpw The new password to set
targprinc if unset, the default principal is used.
result_code Result code, KRB5_KPASSWD_SUCCESS is when password is changed.
result_code_string binary message from the server, contains at least the result_code.
result_string A message from the kpasswd service or the library in human printable form. The string is NUL terminated.

Returns:

On sucess and *result_code is KRB5_KPASSWD_SUCCESS, the password is changed.

@

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_set_real_time (krb5_context context, krb5_timestamp sec, int32_t usec)

Set the absolute time that the caller knows the kdc has so the kerberos library can calculate the relative diffrence beteen the KDC time and local system time.

Parameters:

context Keberos 5 context.
sec The applications new of 'now' in seconds
usec The applications new of 'now' in micro seconds

Returns:

Kerberos 5 error code, see krb5_get_error_message().

If the caller passes in a negative usec, its assumed to be unknown and the function will use the current time usec.

KRB5_LIB_FUNCTION void KRB5_LIB_CALL krb5_set_use_admin_kdc (krb5_context context, krb5_boolean flag)

Make the kerberos library default to the admin KDC.

Parameters:

context Kerberos 5 context.
flag boolean flag to select if the use the admin KDC or not.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_ticket_get_authorization_data_type (krb5_context context, krb5_ticket * ticket, int type, krb5_data * data)

Extract the authorization data type of type from the ticket. Store the field in data. This function is to use for kerberos applications.

Parameters:

context a Kerberos 5 context
ticket Kerberos ticket
type type to fetch
data returned data, free with **krb5_data_free()**

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_ticket_get_client (krb5_context context, const krb5_ticket * ticket, krb5_principal * client)

Return client principal in ticket

Parameters:

context a Kerberos 5 context
ticket ticket to copy

client client principal, free with **krb5_free_principal()**

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().

KRB5_LIB_FUNCTION time_t KRB5_LIB_CALL krb5_ticket_get_endtime (krb5_context context, const krb5_ticket * ticket)

Return end time of ticket

Parameters:

context a Kerberos 5 context
ticket ticket to copy

Returns:

end time of ticket

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_ticket_get_server (krb5_context context, const krb5_ticket * ticket, krb5_principal * server)

Return server principal in ticket

Parameters:

context a Kerberos 5 context
ticket ticket to copy
server server principal, free with **krb5_free_principal()**

Returns:

Returns 0 to indicate success. Otherwise an kerberos et error code is returned, see krb5_get_error_message().