

NAME

Heimdal Kerberos 5 keytab handling functions -

Functions

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_register** (krb5_context context, const krb5_kt_ops *ops)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_resolve** (krb5_context context, const char *name, krb5_keytab *id)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_default_name** (krb5_context context, char *name, size_t namesize)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_default_modify_name** (krb5_context context, char *name, size_t namesize)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_default** (krb5_context context, krb5_keytab *id)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_read_service_key** (krb5_context context, krb5_pointer keyprocarg, krb5_principal principal, krb5_kvno vno, krb5_etype etype, krb5_keyblock **key)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_get_type** (krb5_context context, krb5_keytab keytab, char *prefix, size_t prefixsize)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_get_name** (krb5_context context, krb5_keytab keytab, char *name, size_t namesize)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_get_full_name** (krb5_context context, krb5_keytab keytab, char **str)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_close** (krb5_context context, krb5_keytab id)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_destroy** (krb5_context context, krb5_keytab id)

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL **krb5_kt_compare** (krb5_context context, krb5_keytab_entry *entry, krb5_const_principal principal, krb5_kvno vno, krb5_etype etype)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_get_entry** (krb5_context context, krb5_keytab id, krb5_const_principal principal, krb5_kvno kvno, krb5_etype etype, krb5_keytab_entry *entry)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_copy_entry_contents** (krb5_context context, const krb5_keytab_entry *in, krb5_keytab_entry *out)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_free_entry** (krb5_context context, krb5_keytab_entry *entry)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_start_seq_get** (krb5_context context, krb5_keytab id, krb5_kt_cursor *cursor)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_next_entry** (krb5_context context, krb5_keytab id, krb5_keytab_entry *entry, krb5_kt_cursor *cursor)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_end_seq_get** (krb5_context context, krb5_keytab id, krb5_kt_cursor *cursor)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_add_entry** (krb5_context context, krb5_keytab id, krb5_keytab_entry *entry)

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL **krb5_kt_remove_entry** (krb5_context context, krb5_keytab id, krb5_keytab_entry *entry)

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL **krb5_kt_have_content** (krb5_context context, krb5_keytab id)

Detailed Description

Function Documentation

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_add_entry (krb5_context context, krb5_keytab id, krb5_keytab_entry * entry)

Add the entry in 'entry' to the keytab 'id'.

Parameters:

context a Keberos context.

id a keytab.

entry the entry to add

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_close (krb5_context context, krb5_keytab id)

Finish using the keytab in 'id'. All resources will be released, even on errors.

Parameters:

context a Keberos context.

id keytab to close.

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL krb5_kt_compare (krb5_context context, krb5_keytab_entry * entry, krb5_const_principal principal, krb5_kvno vno, krb5_etype enctype)

Compare 'entry' against 'principal, vno, enctype'. Any of 'principal, vno, enctype' might be 0 which acts as a wildcard. Return TRUE if they compare the same, FALSE otherwise.

Parameters:

context a Keberos context.
entry an entry to match with.
principal principal to match, NULL matches all principals.
vno key version to match, 0 matches all key version numbers.
enctype encryption type to match, 0 matches all encryption types.

Returns:

Return TRUE or match, FALSE if not matched.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_copy_entry_contents
(krb5_context context, const krb5_keytab_entry * in, krb5_keytab_entry * out)
 Copy the contents of 'in' into 'out'.

Parameters:

context a Keberos context.
in the keytab entry to copy.
out the copy of the keytab entry, free with **krb5_kt_free_entry()**.

Returns:

Return an error code or 0, see **krb5_get_error_message()**.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_default (**krb5_context context,**
krb5_keytab * id)
 Set 'id' to the default keytab.

Parameters:

context a Keberos context.
id the new default keytab.

Returns:

Return an error code or 0, see **krb5_get_error_message()**.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_default_modify_name
(krb5_context context, char * name, size_t namesize)
 Copy the name of the default modify keytab into 'name'.

Parameters:

context a Keberos context.
name buffer where the name will be written
namesize length of name

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION `krb5_error_code` **KRB5_LIB_CALL** `krb5_kt_default_name` (`krb5_context` `context`, `char * name`, `size_t namesize`)

copy the name of the default keytab into 'name'.

Parameters:

context a Keberos context.

name buffer where the name will be written

namesize length of name

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION `krb5_error_code` **KRB5_LIB_CALL** `krb5_kt_destroy` (`krb5_context` `context`, `krb5_keytab` `id`)

Destroy (remove) the keytab in 'id'. All resources will be released, even on errors, does the equivalent of `krb5_kt_close()` on the resources.

Parameters:

context a Keberos context.

id keytab to destroy.

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION `krb5_error_code` **KRB5_LIB_CALL** `krb5_kt_end_seq_get` (`krb5_context` `context`, `krb5_keytab` `id`, `krb5_kt_cursor * cursor`)

Release all resources associated with 'cursor'.

Parameters:

context a Keberos context.

id a keytab.

cursor the cursor to free.

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION `krb5_error_code` **KRB5_LIB_CALL** `krb5_kt_free_entry` (`krb5_context` `context`,

krb5_keytab_entry * entry)

Free the contents of 'entry'.

Parameters:

context a Keberos context.

entry the entry to free

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_get_entry (krb5_context context, krb5_keytab id, krb5_const_principal principal, krb5_kvno kvno, krb5_etype enctype, krb5_keytab_entry * entry)

Retrieve the keytab entry for 'principal, kvno, enctype' into 'entry' from the keytab 'id'. Matching is done like `krb5_kt_compare()`.

Parameters:

context a Keberos context.

id a keytab.

principal principal to match, NULL matches all principals.

kvno key version to match, 0 matches all key version numbers.

enctype encryption type to match, 0 matches all encryption types.

entry the returned entry, free with `krb5_kt_free_entry()`.

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_get_full_name (krb5_context context, krb5_keytab keytab, char ** str)

Retrieve the full name of the keytab 'keytab' and store the name in 'str'.

Parameters:

context a Keberos context.

keytab keytab to get name for.

str the name of the keytab name, use `krb5_xfree()` to free the string. On error, *str is set to NULL.

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_get_name (krb5_context context, krb5_keytab keytab, char * name, size_t namesize)

Retrieve the name of the keytab 'keytab' into 'name', 'namesize'

Parameters:

context a Keberos context.

keytab the keytab to get the name for.

name name buffer.

namesize size of name buffer.

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_get_type (krb5_context context, krb5_keytab keytab, char * prefix, size_t prefixsize)

Return the type of the 'keytab' in the string 'prefix' of length 'prefixsize'.

Parameters:

context a Keberos context.

keytab the keytab to get the prefix for

prefix prefix buffer

prefixsize length of prefix buffer

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_boolean KRB5_LIB_CALL krb5_kt_have_content (krb5_context context, krb5_keytab id)

Return true if the keytab exists and have entries

Parameters:

context a Keberos context.

id a keytab.

Returns:

Return an error code or 0, see `krb5_get_error_message()`.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_next_entry (krb5_context context, krb5_keytab id, krb5_keytab_entry * entry, krb5_kt_cursor * cursor)

Get the next entry from keytab, advance the cursor. On last entry the function will return

KRB5_KT_END.

Parameters:

context a Keberos context.

id a keytab.

entry the returned entry, free with **krb5_kt_free_entry()**.

cursor the cursor of the iteration.

Returns:

Return an error code or 0, see **krb5_get_error_message()**.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_read_service_key (krb5_context context, krb5_pointer keyprocarg, krb5_principal principal, krb5_kvno vno, krb5_etype etype, krb5_keyblock ** key)

Read the key identified by '(principal, vno, etype)' from the keytab in 'keyprocarg' (the default if == NULL) into '*key'.

Parameters:

context a Keberos context.

keyprocarg

principal

vno

etype

key

Returns:

Return an error code or 0, see **krb5_get_error_message()**.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_register (krb5_context context, const krb5_kt_ops * ops)

Register a new keytab backend.

Parameters:

context a Keberos context.

ops a backend to register.

Returns:

Return an error code or 0, see **krb5_get_error_message()**.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_remove_entry (krb5_context

context, krb5_keytab id, krb5_keytab_entry * entry)

Remove an entry from the keytab, matching is done using **krb5_kt_compare()**.

Parameters:

context a Keberos context.

id a keytab.

entry the entry to remove

Returns:

Return an error code or 0, see **krb5_get_error_message()**.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_resolve (krb5_context context, const char * name, krb5_keytab * id)

Resolve the keytab name (of the form 'type:residual') in 'name' into a keytab in 'id'.

Parameters:

context a Keberos context.

name name to resolve

id resulting keytab, free with **krb5_kt_close()**.

Returns:

Return an error code or 0, see **krb5_get_error_message()**.

KRB5_LIB_FUNCTION krb5_error_code KRB5_LIB_CALL krb5_kt_start_seq_get (krb5_context context, krb5_keytab id, krb5_kt_cursor * cursor)

Set 'cursor' to point at the beginning of 'id'.

Parameters:

context a Keberos context.

id a keytab.

cursor a newly allocated cursor, free with **krb5_kt_end_seq_get()**.

Returns:

Return an error code or 0, see **krb5_get_error_message()**.