

**NAME**

**krb5\_add\_et\_list**, **krb5\_add\_extra\_addresses**, **krb5\_add\_ignore\_addresses**, **krb5\_context**, **krb5\_free\_config\_files**, **krb5\_free\_context**, **krb5\_get\_default\_config\_files**, **krb5\_get\_dns\_canonize\_hostname**, **krb5\_get\_extra\_addresses**, **krb5\_get\_fcache\_version**, **krb5\_get\_ignore\_addresses**, **krb5\_get\_kdc\_sec\_offset**, **krb5\_get\_max\_time\_skew**, **krb5\_get\_use\_admin\_kdc**, **krb5\_init\_context**, **krb5\_init\_ets**, **krb5\_prepend\_config\_files**, **krb5\_prepend\_config\_files\_default**, **krb5\_set\_config\_files**, **krb5\_set\_dns\_canonize\_hostname**, **krb5\_set\_extra\_addresses**, **krb5\_set\_fcache\_version**, **krb5\_set\_ignore\_addresses**, **krb5\_set\_max\_time\_skew**, **krb5\_set\_use\_admin\_kdc**, - create, modify and delete **krb5\_context** structures

**LIBRARY**

Kerberos 5 Library (libkrb5, -lkrb5)

**SYNOPSIS**

```
#include <krb5.h>
```

```
struct krb5_context;
```

*krb5\_error\_code*

```
krb5_init_context(krb5_context *context);
```

*void*

```
krb5_free_context(krb5_context context);
```

*void*

```
krb5_init_ets(krb5_context context);
```

*krb5\_error\_code*

```
krb5_add_et_list(krb5_context context, void (*func)(struct et_list **));
```

*krb5\_error\_code*

```
krb5_add_extra_addresses(krb5_context context, krb5_addresses *addresses);
```

*krb5\_error\_code*

```
krb5_set_extra_addresses(krb5_context context, const krb5_addresses *addresses);
```

*krb5\_error\_code*

```
krb5_get_extra_addresses(krb5_context context, krb5_addresses *addresses);
```

*krb5\_error\_code*

**krb5\_add\_ignore\_addresses**(*krb5\_context context, krb5\_addresses \*addresses*);

*krb5\_error\_code*

**krb5\_set\_ignore\_addresses**(*krb5\_context context, const krb5\_addresses \*addresses*);

*krb5\_error\_code*

**krb5\_get\_ignore\_addresses**(*krb5\_context context, krb5\_addresses \*addresses*);

*krb5\_error\_code*

**krb5\_set\_fcache\_version**(*krb5\_context context, int version*);

*krb5\_error\_code*

**krb5\_get\_fcache\_version**(*krb5\_context context, int \*version*);

*void*

**krb5\_set\_dns\_canonize\_hostname**(*krb5\_context context, krb5\_boolean flag*);

*krb5\_boolean*

**krb5\_get\_dns\_canonize\_hostname**(*krb5\_context context*);

*krb5\_error\_code*

**krb5\_get\_kdc\_sec\_offset**(*krb5\_context context, int32\_t \*sec, int32\_t \*usec*);

*krb5\_error\_code*

**krb5\_set\_config\_files**(*krb5\_context context, char \*\*filenames*);

*krb5\_error\_code*

**krb5\_prepend\_config\_files**(*const char \*filelist, char \*\*pq, char \*\*\*ret\_pp*);

*krb5\_error\_code*

**krb5\_prepend\_config\_files\_default**(*const char \*filelist, char \*\*\*pfilenames*);

*krb5\_error\_code*

**krb5\_get\_default\_config\_files**(*char \*\*\*pfilenames*);

*void*

**krb5\_free\_config\_files**(*char \*\*filenames*);

*void*

```
krb5_set_use_admin_kdc(krb5_context context, krb5_boolean flag);
```

*krb5\_boolean*

```
krb5_get_use_admin_kdc(krb5_context context);
```

*time\_t*

```
krb5_get_max_time_skew(krb5_context context);
```

*krb5\_error\_code*

```
krb5_set_max_time_skew(krb5_context context, time_t time);
```

## DESCRIPTION

The **krb5\_init\_context**() function 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.

**krb5\_init\_context**() 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\_init\_ets**() adds all com\_err(3) libs to *context*. This is done by **krb5\_init\_context**().

**krb5\_add\_et\_list**() adds a com\_err(3) error-code handler *func* to the specified *context*. The error handler must be generated by the re-entrant version of the compile\_et(1) program. **krb5\_add\_extra\_addresses**() add a list of addresses that should be added when requesting tickets.

**krb5\_add\_ignore\_addresses**() add a list of addresses that should be ignored when requesting tickets.

**krb5\_get\_extra\_addresses**() get the list of addresses that should be added when requesting tickets.

**krb5\_get\_ignore\_addresses**() get the list of addresses that should be ignored when requesting tickets.

**krb5\_set\_ignore\_addresses**() set the list of addresses that should be ignored when requesting tickets.

**krb5\_set\_extra\_addresses**() set the list of addresses that should be added when requesting tickets.

**krb5\_set\_fcache\_version**() sets the version of file credentials caches that should be used.

**krb5\_get\_fcache\_version**() gets the version of file credentials caches that should be used.

**krb5\_set\_dns\_canonize\_hostname()** sets if the context is configured to canonicalize hostnames using DNS.

**krb5\_get\_dns\_canonize\_hostname()** returns if the context is configured to canonicalize hostnames using DNS.

**krb5\_get\_kdc\_sec\_offset()** returns the offset between the localtime and the KDC's time. *sec* and *usec* are both optional argument and NULL can be passed in.

**krb5\_set\_config\_files()** set the list of configuration files to use and re-initialize the configuration from the files.

**krb5\_prepend\_config\_files()** parse the *filelist* and prepend the result to the already existing list *pp*. The result is returned in *ret\_pp* and should be freed with **krb5\_free\_config\_files()**.

**krb5\_prepend\_config\_files\_default()** parse the *filelist* and append that to the default list of configuration files.

**krb5\_get\_default\_config\_files()** get a list of default configuration files.

**krb5\_free\_config\_files()** free a list of configuration files returned by **krb5\_get\_default\_config\_files()**, **krb5\_prepend\_config\_files\_default()**, or **krb5\_prepend\_config\_files()**.

**krb5\_set\_use\_admin\_kdc()** sets if all KDC requests should go admin KDC.

**krb5\_get\_use\_admin\_kdc()** gets if all KDC requests should go admin KDC.

**krb5\_get\_max\_time\_skew()** and **krb5\_set\_max\_time\_skew()** get and sets the maximum allowed time skew between client and server.

## SEE ALSO

errno(2), krb5(3), krb5\_config(3), krb5\_context(3), kerberos(8)