NAME

gssapi - Generic Security Services API

LIBRARY

GSS-API Library (libgssapi, -lgssapi)

SYNOPSIS

#include <gssapi/gssapi.h>

DESCRIPTION

The Generic Security Service Application Programming Interface provides security services to its callers, and is intended for implementation atop a variety of underlying cryptographic mechanisms. Typically, GSS-API callers will be application protocols into which security enhancements are integrated through invocation of services provided by the GSS-API. The GSS-API allows a caller application to authenticate a principal identity associated with a peer application, to delegate rights to a peer, and to apply security services such as confidentiality and integrity on a per-message basis.

There are four stages to using the GSS-API:

a) The application acquires a set of credentials with which it may prove its identity to other processes. The application's credentials vouch for its global identity, which may or may not be related to any local username under which it may be running.

b)

A pair of communicating applications establish a joint security context using their credentials. The security context is a pair of GSS-API data structures that contain shared state information, which is required in order that per-message security services may be provided. Examples of state that might be shared between applications as part of a security context are cryptographic keys, and message sequence numbers. As part of the establishment of a security context, the context initiator is authenticated to the responder, and may require that the responder is authenticated in turn. The initiator may optionally give the responder the right to initiate further security contexts, acting as an agent or delegate of the initiator. This transfer of rights is termed delegation, and is achieved by creating a set of credentials, similar to those used by the initiating application, but which may be used by the responder.

To establish and maintain the shared information that makes up the security context, certain GSS-API calls will return a token data structure, which is an opaque data type that may contain cryptographically protected data. The caller of such a GSS-API routine is responsible for transferring the token to the peer application, encapsulated if necessary in an application protocol. On receipt of such a token, the peer application should pass it to a corresponding GSS-API routine which will

decode the token and extract the information, updating the security context state information accordingly.

c) Per-message services are invoked to apply either:

integrity and data origin authentication, or confidentiality, integrity and data origin authentication to application data, which are treated by GSS-API as arbitrary octet-strings. An application transmitting a message that it wishes to protect will call the appropriate GSS-API routine (gss_get_mic or gss_wrap) to apply protection, specifying the appropriate security context, and send the resulting token to the receiving application. The receiver will pass the received token (and, in the case of data protected by gss_get_mic, the accompanying message-data) to the corresponding decoding routine (gss_verify_mic or gss_unwrap) to remove the protection and validate the data.

d)
At the completion of a communications session (which may extend across several transport connections), each application calls a GSS-API routine to delete the security context. Multiple contexts may also be used (either successively or simultaneously) within a single communications association, at the option of the applications.

GSS-API ROUTINES

This section lists the routines that make up the GSS-API, and offers a brief description of the purpose of each routine.

GSS-API Credential-management Routines:

gss_acquire_cred	Assume a global identity; Obtain a GSS-API credential handle for pre-existing credentials.
gss_add_cred	Construct credentials incrementally
gss_inquire_cred	Obtain information about a credential
gss_inquire_cred_by_mech	Obtain per-mechanism information about a credential.
gss_release_cred	Discard a credential handle.

GSS-API Context-Level Routines:

gss_init_sec_context	Initiate a security context with a peer application

gss_accept_sec_context	Accept a security context initiated by a peer application
gss_delete_sec_context	Discard a security context
gss_process_context_token	Process a token on a security context from a peer application
gss_context_time	Determine for how long a context will remain valid
gss_inquire_context	Obtain information about a security context
gss_wrap_size_limit	Determine token-size limit for gss_wrap(3) on a context
gss_export_sec_context	Transfer a security context to another process
gss_import_sec_context	Import a transferred context

GSS-API Per-message Routines:

gss_get_mic	Calculate a cryptographic message integrity code (MIC) for a message; integrity service
gss_verify_mic	Check a MIC against a message; verify integrity of a received message
gss_wrap	Attach a MIC to a message, and optionally encrypt the message content; confidentiality service
gss_unwrap	Verify a message with attached MIC, and decrypt message content if necessary.

GSS-API Name manipulation Routines:

gss_import_name	Convert a contiguous string name to internal-form
gss_display_name	Convert internal-form name to text
gss_compare_name	Compare two internal-form names
gss_release_name	Discard an internal-form name

gss_inquire_names_for_mech

List the name-types supported by the specified mechanism

gss_inquire_mechs_for_name
List mechanisms that support the specified name-type

gss_canonicalize_name Convert an internal name to an MN

gss_export_name Convert an MN to export form

gss_duplicate_name Create a copy of an internal name

GSS-API Miscellaneous Routines

gss_add_oid_set_member Add an object identifier to a set

gss_display_status Convert a GSS-API status code to text

gss_indicate_mechs Determine available underlying authentication mechanisms

gss_release_buffer Discard a buffer

gss_release_oid_set Discard a set of object identifiers

gss_create_empty_oid_set Create a set containing no object identifiers

gss_test_oid_set_member Determines whether an object identifier is a member of a set.

Individual GSS-API implementations may augment these routines by providing additional mechanism-specific routines if required functionality is not available from the generic forms. Applications are encouraged to use the generic routines wherever possible on portability grounds.

STANDARDS

RFC 2743 Generic Security Service Application Program Interface Version 2, Update 1

RFC 2744 Generic Security Service API Version 2 : C-bindings

HISTORY

The **gssapi** library first appeared in FreeBSD 7.0.

AUTHORS

John Wray, Iris Associates

COPYRIGHT