

NAME

gss_export_sec_context - Transfer a security context to another process

SYNOPSIS

```
#include <gssapi/gssapi.h>
```

```
OM_uint32
```

```
gss_export_sec_context(OM_uint32 *minor_status, gss_ctx_id_t *context_handle,  
    gss_buffer_t interprocess_token);
```

DESCRIPTION

Provided to support the sharing of work between multiple processes. This routine will typically be used by the context-acceptor, in an application where a single process receives incoming connection requests and accepts security contexts over them, then passes the established context to one or more other processes for message exchange. **gss_export_sec_context()** deactivates the security context for the calling process and creates an interprocess token which, when passed to **gss_import_sec_context()** in another process, will re-activate the context in the second process. Only a single instantiation of a given context may be active at any one time; a subsequent attempt by a context exporter to access the exported security context will fail.

The implementation may constrain the set of processes by which the interprocess token may be imported, either as a function of local security policy, or as a result of implementation decisions. For example, some implementations may constrain contexts to be passed only between processes that run under the same account, or which are part of the same process group.

The interprocess token may contain security-sensitive information (for example cryptographic keys). While mechanisms are encouraged to either avoid placing such sensitive information within interprocess tokens, or to encrypt the token before returning it to the application, in a typical object-library GSS-API implementation this may not be possible. Thus the application must take care to protect the interprocess token, and ensure that any process to which the token is transferred is trustworthy.

If creation of the interprocess token is successful, the implementation shall deallocate all process-wide resources associated with the security context, and set the *context_handle* to `GSS_C_NO_CONTEXT`. In the event of an error that makes it impossible to complete the export of the security context, the implementation must not return an interprocess token, and should strive to leave the security context referenced by the *context_handle* parameter untouched. If this is impossible, it is permissible for the implementation to delete the security context, providing it also sets the *context_handle* parameter to `GSS_C_NO_CONTEXT`.

PARAMETERS

`minor_status` Mechanism specific status code.

`context_handle` Context handle identifying the context to transfer.

`interprocess_token` Token to be transferred to target process. Storage associated with this token must be freed by the application after use with a call to `gss_release_buffer()`.

RETURN VALUES

`GSS_S_COMPLETE` Successful completion

`GSS_S_CONTEXT_EXPIRED` The context has expired

`GSS_S_NO_CONTEXT` The context was invalid

`GSS_S_UNAVAILABLE` The operation is not supported

SEE ALSO

`gss_import_sec_context(3)`, `gss_release_buffer(3)`

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 `gss_export_sec_context` function first appeared in FreeBSD 7.0.

AUTHORS

John Wray, Iris Associates

COPYRIGHT

Copyright (C) The Internet Society (2000). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be

followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.