

NAME

gss_wrap_size_limit - Determine maximum message sizes

SYNOPSIS

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

```
OM_uint32
```

```
gss_wrap_size_limit(OM_uint32 *minor_status, const gss_ctx_id_t context_handle, int conf_req_flag,
    gss_qop_t qop_req, OM_uint32 req_output_size, OM_uint32 *max_input_size);
```

DESCRIPTION

Allows an application to determine the maximum message size that, if presented to `gss_wrap(3)` with the same `conf_req_flag` and `qop_req` parameters, will result in an output token containing no more than `req_output_size` bytes.

This call is intended for use by applications that communicate over protocols that impose a maximum message size. It enables the application to fragment messages prior to applying protection.

GSS-API implementations are recommended but not required to detect invalid QOP values when `gss_wrap_size_limit()` is called. This routine guarantees only a maximum message size, not the availability of specific QOP values for message protection.

Successful completion of this call does not guarantee that `gss_wrap(3)` will be able to protect a message of length `max_input_size` bytes, since this ability may depend on the availability of system resources at the time that `gss_wrap(3)` is called. However, if the implementation itself imposes an upper limit on the length of messages that may be processed by `gss_wrap`, the implementation should not return a value via `max_input_bytes` that is greater than this length.

PARAMETERS

`minor_status` Mechanism specific status code.

`context_handle` A handle that refers to the security over which the messages will be sent.

`conf_req_flag` Indicates whether `gss_wrap(3)` will be asked to apply confidentiality protection in addition to integrity protection.

`qop_req` Indicates the level of protection that `gss_wrap(3)` will be asked to provide.

`req_output_size` The desired maximum size for tokens emitted by `gss_wrap(3)`.

`max_input_size` The maximum input message size that may be presented to `gss_wrap(3)` in order to guarantee that the emitted token shall be no larger than `req_output_size` bytes.

RETURN VALUES

`GSS_S_COMPLETE` Successful completion.

`GSS_S_NO_CONTEXT` The referenced context could not be accessed.

`GSS_S_CONTEXT_EXPIRED` The context has expired.

`GSS_S_BAD_QOP` The specified QOP is not supported by the mechanism.

SEE ALSO

`gss_wrap(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_wrap_size_limit` 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.