NAME

gss_display_status - Convert a GSS-API status code to text

SYNOPSIS

#include <gssapi/gssapi.h>

OM_uint32

gss_display_status(*OM_uint32 *minor_status*, *OM_uint32 status_value*, *int status_type*, *const gss_OID mech_type*, *OM_uint32 *message_context*, *gss_buffer_t status_string*);

DESCRIPTION

Allows an application to obtain a textual representation of a GSS-API status code, for display to the user or for logging purposes. Since some status values may indicate multiple conditions, applications may need to call **gss_display_status**() multiple times, each call generating a single text string. The *message_context* parameter is used by **gss_display_status**() to store state information about which error messages have already been extracted from a given *status_value*; *message_context* must be initialized to zero by the application prior to the first call, and **gss_display_status**() will return a non-zero value in this parameter if there are further messages to extract.

The *message_context* parameter contains all state information required by **gss_display_status**() in order to extract further messages from the *status_value*; even when a non-zero value is returned in this parameter, the application is not required to call **gss_display_status**() again unless subsequent messages are desired. The following code extracts all messages from a given status code and prints them to stderr:

```
OM_uint32 message_context;
OM_uint32 status_code;
OM_uint32 maj_status;
OM_uint32 min_status;
gss_buffer_desc status_string;

    ...

message_context = 0;

do {

  maj_status = gss_display_status (
                &min_status,
                status_code,
                GSS_C_GSS_CODE,
```

```
                     GSS_C_NO_OID,
                     &message_context,
                     &status_string)

  fprintf(stderr,
            "%.*s\n",
          (int)status_string.length,
          (char *)status_string.value);

  gss_release_buffer(&min_status, &status_string);

} while (message_context != 0);
```

PARAMETERS

minor_status Mechanism specific status code.

status_value Status value to be converted

status_type

> GSS_C_GSS_CODE   *status_value* is a GSS status code

> GSS_C_MECH_CODE *status_value* is a mechanism status code

mech_type Underlying mechanism (used to interpret a minor status value). Supply GSS_C_NO_OID to obtain the system default.

message_context
> Should be initialized to zero by the application prior to the first call. On return from **gss_display_status**(), a non-zero status_value parameter indicates that additional messages may be extracted from the status code via subsequent calls to **gss_display_status**(), passing the same *status_value*, *status_type*, *mech_type*, and *message_context* parameters.

status_string Textual interpretation of the *status_value*. Storage associated with this parameter must be freed by the application after use with a call to **gss_release_buffer**().

RETURN VALUES

GSS_S_COMPLETE   Successful completion

GSS_S_BAD_MECH   Indicates that translation in accordance with an unsupported mechanism type

was requested

GSS_S_BAD_STATUS The status value was not recognized, or the status type was neither
                  GSS_C_GSS_CODE nor GSS_C_MECH_CODE.

SEE ALSO
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_display_status** 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.