

globus authz

2.2

Generated by Doxygen 1.8.1.1

Fri Aug 3 2012 03:21:12

# Contents

<b>1</b>	<b>Module Index</b>	<b>1</b>
1.1	Modules . . . . .	1
<b>2</b>	<b>Module Documentation</b>	<b>1</b>
2.1	GSI Authorization API . . . . .	1
2.1.1	Detailed Description . . . . .	2
2.1.2	Function Documentation . . . . .	2
2.2	GSI Credential Constants . . . . .	4
2.2.1	Detailed Description . . . . .	4
2.2.2	Enumeration Type Documentation . . . . .	4

## 1 Module Index

### 1.1 Modules

Here is a list of all modules:

<b>GSI Authorization API</b>	<b>1</b>
<b>GSI Credential Constants</b>	<b>4</b>

## 2 Module Documentation

### 2.1 GSI Authorization API

#### Initialize Handle

- globus\_result\_t **globus\_gsi\_authz\_handle\_init** (globus\_gsi\_authz\_handle\_t \*handle, const char \*service\_name, const gss\_ctx\_id\_t context, globus\_gsi\_authz\_cb\_t callback, void \*callback\_arg)

#### Authorization decision made here

- globus\_result\_t **globus\_gsi\_authorize** (globus\_gsi\_authz\_handle\_t handle, const void \*action, const void \*object, globus\_gsi\_authz\_cb\_t callback, void \*callback\_arg)

#### Destroy Handle

- globus\_result\_t **globus\_gsi\_authz\_handle\_destroy** (globus\_gsi\_authz\_handle\_t handle, globus\_gsi\_authz\_cb\_t callback, void \*callback\_arg)

#### Query for authorization identity

- globus\_result\_t **globus\_gsi\_authz\_get\_authorization\_identity** (globus\_gsi\_authz\_handle\_t handle, char \*\*identity\_ptr, globus\_gsi\_authz\_cb\_t callback, void \*callback\_arg)

### 2.1.1 Detailed Description

### 2.1.2 Function Documentation

2.1.2.1 `globus_result_t globus_gsi_authz_handle_init ( globus_gsi_authz_handle_t * handle, const char * service_name, const gss_ctx_id_t context, globus_gsi_authz_cb_t callback, void * callback_arg )`

Initializes a handle.

#### Parameters

<i>handle</i>	Pointer to the handle that is to be initialized
<i>service_name</i>	Service to authorize access to
<i>context</i>	Security context used to contact the service
<i>callback</i>	Callback function to call when authz handle init completes
<i>callback_arg</i>	Argument to callback function

#### Returns

GLOBUS\_SUCCESS if successful A Globus error object on failure:

2.1.2.2 `globus_result_t globus_gsi_authorize ( globus_gsi_authz_handle_t handle, const void * action, const void * object, globus_gsi_authz_cb_t callback, void * callback_arg )`

Authorization decision made here.

#### Parameters

<i>handle</i>	Pointer to the handle that is to be initialized
<i>action</i>	Action to authorize
<i>object</i>	Object that the action pertains to.
<i>callback</i>	Callback function to call when authorization completes
<i>callback_arg</i>	Argument to callback function

#### Returns

GLOBUS\_SUCCESS if successful A Globus error object on failure:

2.1.2.3 `globus_result_t globus_gsi_authz_handle_destroy ( globus_gsi_authz_handle_t handle, globus_gsi_authz_cb_t callback, void * callback_arg )`

Destroy a Globus GSI authz handle.

#### Parameters

<i>handle</i>	The handle that is to be destroyed
<i>callback</i>	Callback function to call when handle is destroyed
<i>callback_arg</i>	Argument to callback function

#### Returns

GLOBUS\_SUCCESS

2.1.2.4 `globus_result_t globus_gsi_authz_get_authorization_identity ( globus_gsi_authz_handle_t handle, char ** identity_ptr,  
globus_gsi_authz_cb_t callback, void * callback_arg )`

Query for authorization identity.

#### Parameters

<i>handle</i>	The handle that is to be used for the identity check.
<i>identity_ptr</i>	The authorization identity determined by the authorization handle. This is must be freed by the caller. If the value is NULL (and this function returned GLOBUS_SUCCESS), the caller should use the authenticated identity.
<i>callback</i>	Callback function to call when identity is determined.
<i>callback_arg</i>	Argument to callback function.

#### Returns

GLOBUS\_SUCCESS

## 2.2 GSI Credential Constants

### Enumerations

- enum **globus\_gsi\_authz\_error\_t** {  
    **GLOBUS\_GSI\_AUTHZ\_ERROR\_SUCCESS** = 0,  
    **GLOBUS\_GSI\_AUTHZ\_ERROR\_ERRNO** = 1,  
    **GLOBUS\_GSI\_AUTHZ\_ERROR\_BAD\_PARAMETER** = 2,  
    **GLOBUS\_GSI\_AUTHZ\_ERROR\_CALLOUT** = 3 }

#### 2.2.1 Detailed Description

#### 2.2.2 Enumeration Type Documentation

##### 2.2.2.1 enum globus\_gsi\_authz\_error\_t

GSI Authz Error codes.

Enumerator:

- GLOBUS\_GSI\_AUTHZ\_ERROR\_SUCCESS** Success.
- GLOBUS\_GSI\_AUTHZ\_ERROR\_ERRNO** Error with system call.
- GLOBUS\_GSI\_AUTHZ\_ERROR\_BAD\_PARAMETER** Invalid parameter.
- GLOBUS\_GSI\_AUTHZ\_ERROR\_CALLOUT** Callout returned an error.

## Index

- GLOBUS\_GSI\_AUTHZ\_ERROR\_BAD\_PARAMETER
  - GSI Credential Constants, 4
- GLOBUS\_GSI\_AUTHZ\_ERROR\_CALLOUT
  - GSI Credential Constants, 4
- GLOBUS\_GSI\_AUTHZ\_ERROR\_ERRNO
  - GSI Credential Constants, 4
- GLOBUS\_GSI\_AUTHZ\_ERROR\_SUCCESS
  - GSI Credential Constants, 4
- GSI Authorization API, 1
  - globus\_gsi\_authorize, 2
  - globus\_gsi\_authz\_get\_authorization\_identity, 2
  - globus\_gsi\_authz\_handle\_destroy, 2
  - globus\_gsi\_authz\_handle\_init, 1
- GSI Credential Constants, 4
  - GLOBUS\_GSI\_AUTHZ\_ERROR\_BAD\_PARAMETER, 4
  - GLOBUS\_GSI\_AUTHZ\_ERROR\_CALLOUT, 4
  - GLOBUS\_GSI\_AUTHZ\_ERROR\_ERRNO, 4
  - GLOBUS\_GSI\_AUTHZ\_ERROR\_SUCCESS, 4
  - globus\_gsi\_authz\_error\_t, 4
- globus\_gsi\_authorize
  - GSI Authorization API, 2
- globus\_gsi\_authz\_error\_t
  - GSI Credential Constants, 4
- globus\_gsi\_authz\_get\_authorization\_identity
  - GSI Authorization API, 2
- globus\_gsi\_authz\_handle\_destroy
  - GSI Authorization API, 2
- globus\_gsi\_authz\_handle\_init
  - GSI Authorization API, 1