

globus authz

0.7

Generated by Doxygen 1.7.4

Sat Apr 30 2011 06:02:19

Contents

1	Module Index	1
1.1	Modules	1
2	Module Documentation	1
2.1	GSI Authorization API	1
2.1.1	Function Documentation	2
2.2	GSI Credential Constants	3
2.2.1	Enumeration Type Documentation	3

1 Module Index

1.1 Modules

Here is a list of all modules:

GSI Authorization API	1
GSI Credential Constants	3

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 Function Documentation

2.1.1.1 `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:

2.1.1.2 `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:

2.1.1.3 `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

2.1.1.4 `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 Enumeration Type Documentation

2.2.1.1 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, [3](#)
- GLOBUS_GSI_AUTHZ_ERROR_CALLOUT
 - GSI Credential Constants, [3](#)
- GLOBUS_GSI_AUTHZ_ERROR_ERRNO
 - GSI Credential Constants, [3](#)
- GLOBUS_GSI_AUTHZ_ERROR_SUCCESS
 - GSI Credential Constants, [3](#)
- globus_gsi_authorize
 - GSI Authorization API, [2](#)
- globus_gsi_authz_error_t
 - GSI Credential Constants, [3](#)
- 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](#)
- 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, [2](#)
 - GLOBUS_GSI_AUTHZ_ERROR_BAD_PARAMETER, [3](#)
 - GLOBUS_GSI_AUTHZ_ERROR_CALLOUT, [3](#)
 - GLOBUS_GSI_AUTHZ_ERROR_ERRNO, [3](#)
 - GLOBUS_GSI_AUTHZ_ERROR_SUCCESS, [3](#)
 - globus_gsi_authz_error_t, [3](#)