

# globus gsi cert utils Reference Manual

## 5.5

Generated by Doxygen 1.5.3

Fri Jun 26 23:12:43 2009

# Contents

<b>1</b>	<b><a href="#">Globus GSI Certificate Handling Utilities</a></b>	<b>1</b>
<b>2</b>	<b><a href="#">globus gsi cert utils Module Index</a></b>	<b>1</b>
<b>3</b>	<b><a href="#">globus gsi cert utils Module Documentation</a></b>	<b>1</b>

## 1 Globus GSI Certificate Handling Utilities

The Globus GSI Certificate Handling Utilities library. This library contains helper functions for dealing with certificates.

- [Activation](#)
- [Cert Utils Functions](#)
- [Cert Utils Constants](#)

## 2 globus gsi cert utils Module Index

### 2.1 globus gsi cert utils Modules

Here is a list of all modules:

<b>Activation</b>	<b>1</b>
<b>Cert Utils Functions</b>	<b>2</b>
<b>Cert Utils Constants</b>	<b>2</b>

## 3 globus gsi cert utils Module Documentation

### 3.1 Activation

Globus GSI Cert Utils uses standard Globus module activation and deactivation.

#### Defines

- #define [GLOBUS\\_GSI\\_CERT\\_UTILS\\_MODULE](#)

#### 3.1.1 Detailed Description

Globus GSI Cert Utils uses standard Globus module activation and deactivation.

Before any Globus GSI Cert Utils functions are called, the following function must be called:

```
globus_module_activate(GLOBUS_GSI_CERT_UTILS_MODULE)
```

This function returns `GLOBUS_SUCCESS` if Globus GSI Credential was successfully initialized, and you are therefore allowed to subsequently call Globus GSI Cert Utils functions. Otherwise, an error code is returned, and Globus GSI Cert Utils functions should not be subsequently called. This function may be called multiple times.

To deactivate Globus GSI Cert Utils, the following function must be called:

```
globus_module_deactivate(GLOBUS_GSI_CERT_UTILS_MODULE)
```

This function should be called once for each time Globus GSI Cert Utils was activated.

### 3.1.2 Define Documentation

#### 3.1.2.1 #define GLOBUS\_GSI\_CERT\_UTILS\_MODULE

Module descriptor.

## 3.2 Cert Utils Functions

A generic set of utility functions for manipulating OpenSSL objects, such as X509 certificates.

### Functions

- `globus_result_t globus_gsi_cert_utils_get_eec` (`STACK_OF(X509)*cert_chain`, `X509 **eec`)

#### 3.2.1 Detailed Description

A generic set of utility functions for manipulating OpenSSL objects, such as X509 certificates.

#### 3.2.2 Function Documentation

##### 3.2.2.1 `globus_result_t globus_gsi_cert_utils_get_eec` (`STACK_OF(X509)* cert_chain`, `X509 ** eec`)

Get the end-entity certificate associated with a certificate chain.

### Parameters:

*cert\_chain* Certificate chain to inspect.

*eec* Pointer to be set to the EEC value from within the cert chain. Must freed by the caller.

## 3.3 Cert Utils Constants

### Typedefs

- typedef enum `globus_gsi_cert_utils_cert_type_e` `globus_gsi_cert_utils_cert_type_t`

### Enumerations

- enum `globus_gsi_cert_utils_error_t` {  
    `GLOBUS_GSI_CERT_UTILS_ERROR_SUCCESS` = 0,  
    `GLOBUS_GSI_CERT_UTILS_ERROR_GETTING_NAME_ENTRY_OF_SUBJECT` = 1,  
    `GLOBUS_GSI_CERT_UTILS_ERROR_COPYING_SUBJECT` = 2,

```

GLOBUS_GSI_CERT_UTILS_ERROR_GETTING_CN_ENTRY = 3,
GLOBUS_GSI_CERT_UTILS_ERROR_ADDING_CN_TO_SUBJECT = 4,
GLOBUS_GSI_CERT_UTILS_ERROR_OUT_OF_MEMORY = 5,
GLOBUS_GSI_CERT_UTILS_ERROR_UNEXPECTED_FORMAT = 6,
GLOBUS_GSI_CERT_UTILS_ERROR_NON_COMPLIANT_PROXY = 7,
GLOBUS_GSI_CERT_UTILS_ERROR_DETERMINING_CERT_TYPE = 8,
GLOBUS_GSI_CERT_UTILS_ERROR_LAST = 9 }
• enum globus_gsi_cert_utils_cert_type_e {
    GLOBUS_GSI_CERT_UTILS_TYPE_DEFAULT = 0,
    GLOBUS_GSI_CERT_UTILS_TYPE_EEC = (1 << 0),
    GLOBUS_GSI_CERT_UTILS_TYPE_CA = (1 << 1),
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_2 = (1 << 2),
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_3 = (1 << 3),
    GLOBUS_GSI_CERT_UTILS_TYPE_RFC = (1 << 4),
    GLOBUS_GSI_CERT_UTILS_TYPE_FORMAT_MASK,
    GLOBUS_GSI_CERT_UTILS_TYPE_IMPERSONATION_PROXY = (1 << 5),
    GLOBUS_GSI_CERT_UTILS_TYPE_LIMITED_PROXY = (1 << 6),
    GLOBUS_GSI_CERT_UTILS_TYPE_RESTRICTED_PROXY = (1 << 7),
    GLOBUS_GSI_CERT_UTILS_TYPE_INDEPENDENT_PROXY = (1 << 8),
    GLOBUS_GSI_CERT_UTILS_TYPE_PROXY_MASK,
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_3_IMPERSONATION_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_3_INDEPENDENT_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_3_LIMITED_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_3_RESTRICTED_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_2_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_GSI_2_LIMITED_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_RFC_IMPERSONATION_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_RFC_INDEPENDENT_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_RFC_LIMITED_PROXY,
    GLOBUS_GSI_CERT_UTILS_TYPE_RFC_RESTRICTED_PROXY }

```

### 3.3.1 Typedef Documentation

#### 3.3.1.1 typedef enum globus\_gsi\_cert\_utils\_cert\_type\_e globus\_gsi\_cert\_utils\_cert\_type\_t

Certificate Types.

These certificate types are used to describe some properties of a certificate and to specify what type of proxy should be generated in the proxy core code. There are two non-proxy types of certificates understood by Globus: EEC (End-Entity Certificate) and CA (Certificate Authority Certificates), three proxy formats (GSI 2 "legacy" proxies, GSI 3 "Draft" proxies, and RFC 3820-compliant proxies), and four types of proxy (limited, impersonation "full", restricted, and independent). The latter two types are not expressible in the GSI 2 format.

In addition to enumerations for the concrete renderings of certificate format and type combined, there are default, formats-without-types and types-without-formats so that application logic which uses the proxy library can request default proxy formats which are compatible with the issuing certificate.

### 3.3.2 Enumeration Type Documentation

#### 3.3.2.1 enum globus\_gsi\_cert\_utils\_error\_t

Cert Utils Error Codes.

##### Enumerator:

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_SUCCESS** Success - never used.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_GETTING\_NAME\_ENTRY\_OF\_SUBJECT** Failed to retrieve a subcomponent of the subject.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_COPYING\_SUBJECT** A error occurred while trying to copy a X.509 subject.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_GETTING\_CN\_ENTRY** Failed to retrieve a CN subcomponent of the subject.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_ADDING\_CN\_TO\_SUBJECT** Failed to add a CN component to a X.509 subject name.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_OUT\_OF\_MEMORY** Out of memory.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_UNEXPECTED\_FORMAT** Something unexpected happen while converting a string subject to a X509\_NAME structure.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_NON\_COMPLIANT\_PROXY** Proxy does not comply with the expected format.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_DETERMINING\_CERT\_TYPE** Couldn't determine the certificate type.

**GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_LAST** Last marker - never used.

#### 3.3.2.2 enum globus\_gsi\_cert\_utils\_cert\_type\_e

Certificate Types.

These certificate types are used to describe some properties of a certificate and to specify what type of proxy should be generated in the proxy core code. There are two non-proxy types of certificates understood by Globus: EEC (End-Entity Certificate) and CA (Certificate Authority Certificates), three proxy formats (GSI 2 "legacy" proxies, GSI 3 "Draft" proxies, and RFC 3820-compliant proxies), and four types of proxy (limited, impersonation "full", restricted, and independent). The latter two types are not expressible in the GSI 2 format.

In addition to enumerations for the concrete renderings of certificate format and type combined, there are default, formats-without-types and types-without-formats so that application logic which uses the proxy library can request default proxy formats which are compatible with the issuing certificate.

##### Enumerator:

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_DEFAULT** Default proxy type.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_EEC** A end entity certificate.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_CA** A CA certificate.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2** Legacy Proxy Format.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3** X.509 Proxy Certificate Profile (draft) Proxy Format.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC** X.509 Proxy Certificate Profile Compliant Proxy Format.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_FORMAT\_MASK** Proxy certificate formats mask.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_IMPERSONATION\_PROXY** Impersonation proxy type.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_LIMITED\_PROXY** Limited proxy type.

**GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RESTRICTED\_PROXY** Restricted proxy type.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_INDEPENDENT\_PROXY*** Independent proxy type.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_PROXY\_MASK*** Proxy types mask.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_IMPERSONATION\_PROXY*** A X.509 Proxy Certificate Profile (pre-RFC) compliant impersonation proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_INDEPENDENT\_PROXY*** A X.509 Proxy Certificate Profile (pre-RFC) compliant independent proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_LIMITED\_PROXY*** A X.509 Proxy Certificate Profile (pre-RFC) compliant limited proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_RESTRICTED\_PROXY*** A X.509 Proxy Certificate Profile (pre-RFC) compliant restricted proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2\_PROXY*** A legacy Globus impersonation proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2\_LIMITED\_PROXY*** A legacy Globus limited impersonation proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_IMPERSONATION\_PROXY*** A X.509 Proxy Certificate Profile RFC compliant impersonation proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_INDEPENDENT\_PROXY*** A X.509 Proxy Certificate Profile RFC compliant independent proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_LIMITED\_PROXY*** A X.509 Proxy Certificate Profile RFC compliant limited proxy.

***GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_RESTRICTED\_PROXY*** A X.509 Proxy Certificate Profile RFC compliant restricted proxy.

## Index

Activation, [1](#)

Cert Utils Constants, [2](#)

Cert Utils Functions, [2](#)

globus\_gsi\_cert\_utils

    globus\_gsi\_cert\_utils\_get\_eec, [2](#)

globus\_gsi\_cert\_utils\_activation

    GLOBUS\_GSI\_CERT\_UTILS\_MODULE, [2](#)

globus\_gsi\_cert\_utils\_cert\_type\_e

    globus\_gsi\_cert\_utils\_constants, [4](#)

globus\_gsi\_cert\_utils\_cert\_type\_t

    globus\_gsi\_cert\_utils\_constants, [3](#)

globus\_gsi\_cert\_utils\_constants

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        ADDING\_CN\_TO\_SUBJECT, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        COPYING\_SUBJECT, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        DETERMINING\_CERT\_TYPE, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        GETTING\_CN\_ENTRY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        GETTING\_NAME\_ENTRY\_OF\_-

        SUBJECT, [3](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_LAST,

[4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_NON\_-

        COMPLIANT\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_OUT\_-

        OF\_MEMORY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        SUCCESS, [3](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        UNEXPECTED\_FORMAT, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_CA, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-

        DEFAULT, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_EEC, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-

        FORMAT\_MASK, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2\_-

        LIMITED\_PROXY, [5](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2\_-

        PROXY, [5](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-

        IMPERSONATION\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-

        INDEPENDENT\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-

        LIMITED\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-

        RESTRICTED\_PROXY, [5](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-

        IMPERSONATION\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-

        INDEPENDENT\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-

        LIMITED\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_PROXY\_-

        MASK, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-

        RESTRICTED\_PROXY, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-

        IMPERSONATION\_PROXY, [5](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-

        INDEPENDENT\_PROXY, [5](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-

        LIMITED\_PROXY, [5](#)

    GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-

        RESTRICTED\_PROXY, [5](#)

globus\_gsi\_cert\_utils\_constants

    globus\_gsi\_cert\_utils\_cert\_type\_e, [4](#)

    globus\_gsi\_cert\_utils\_cert\_type\_t, [3](#)

    globus\_gsi\_cert\_utils\_error\_t, [3](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_ADDING\_-

        CN\_TO\_SUBJECT

    globus\_gsi\_cert\_utils\_constants, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_COPYING\_-

        SUBJECT

    globus\_gsi\_cert\_utils\_constants, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-

        DETERMINING\_CERT\_TYPE

    globus\_gsi\_cert\_utils\_constants, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_GETTING\_-

        CN\_ENTRY

    globus\_gsi\_cert\_utils\_constants, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_GETTING\_-

        NAME\_ENTRY\_OF\_SUBJECT

    globus\_gsi\_cert\_utils\_constants, [3](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_LAST

    globus\_gsi\_cert\_utils\_constants, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_NON\_-

        COMPLIANT\_PROXY

    globus\_gsi\_cert\_utils\_constants, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_OUT\_OF\_-

        MEMORY

    globus\_gsi\_cert\_utils\_constants, [4](#)

    GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_SUCCESS

    globus\_gsi\_cert\_utils\_constants, [3](#)

    globus\_gsi\_cert\_utils\_error\_t

    globus\_gsi\_cert\_utils\_constants, [3](#)

GLOBUS\_GSI\_CERT\_UTILS\_ERROR\_-  
UNEXPECTED\_FORMAT  
globus\_gsi\_cert\_utils\_constants, 4  
globus\_gsi\_cert\_utils\_get\_eec  
globus\_gsi\_cert\_utils, 2  
GLOBUS\_GSI\_CERT\_UTILS\_MODULE  
globus\_gsi\_cert\_utils\_activation, 2  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_CA  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_DEFAULT  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_EEC  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_FORMAT\_-  
MASK  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2\_-  
LIMITED\_PROXY  
globus\_gsi\_cert\_utils\_constants, 5  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_2\_-  
PROXY  
globus\_gsi\_cert\_utils\_constants, 5  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-  
IMPERSONATION\_PROXY  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-  
INDEPENDENT\_PROXY  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-  
LIMITED\_PROXY  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_GSI\_3\_-  
RESTRICTED\_PROXY  
globus\_gsi\_cert\_utils\_constants, 5  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-  
IMPERSONATION\_PROXY  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-  
INDEPENDENT\_PROXY  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_LIMITED\_-  
PROXY  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_PROXY\_-  
MASK  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_-  
RESTRICTED\_PROXY  
globus\_gsi\_cert\_utils\_constants, 4  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC  
globus\_gsi\_cert\_utils\_constants, 4

GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-  
IMPERSONATION\_PROXY  
globus\_gsi\_cert\_utils\_constants, 5  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-  
INDEPENDENT\_PROXY  
globus\_gsi\_cert\_utils\_constants, 5  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-  
LIMITED\_PROXY  
globus\_gsi\_cert\_utils\_constants, 5  
GLOBUS\_GSI\_CERT\_UTILS\_TYPE\_RFC\_-  
RESTRICTED\_PROXY  
globus\_gsi\_cert\_utils\_constants, 5