

VOMS C API

1.5.0

Generated by Doxygen 1.7.4

Mon May 30 2011 18:13:21

Contents

| | | |
|----------|--|----------|
| 1 | Data Structure Index | 1 |
| 1.1 | Data Structures | 1 |
| 2 | File Index | 3 |
| 2.1 | File List | 3 |
| 3 | Data Structure Documentation | 5 |
| 3.1 | attribute Struct Reference | 5 |
| 3.1.1 | Detailed Description | 5 |
| 3.1.2 | Field Documentation | 5 |
| 3.1.2.1 | name | 5 |
| 3.1.2.2 | qualifier | 5 |
| 3.1.2.3 | value | 5 |
| 3.2 | contactdata Struct Reference | 6 |
| 3.2.1 | Detailed Description | 6 |
| 3.2.2 | Field Documentation | 6 |
| 3.2.2.1 | contact | 6 |
| 3.2.2.2 | host | 6 |
| 3.2.2.3 | nick | 6 |
| 3.2.2.4 | port | 7 |
| 3.2.2.5 | reserved | 7 |
| 3.2.2.6 | version | 7 |
| 3.2.2.7 | vo | 7 |
| 3.3 | data Struct Reference | 7 |
| 3.3.1 | Detailed Description | 7 |
| 3.3.2 | Field Documentation | 8 |

| | | |
|----------|---------------------------|----|
| 3.3.2.1 | cap | 8 |
| 3.3.2.2 | group | 8 |
| 3.3.2.3 | role | 8 |
| 3.4 | voms Struct Reference | 8 |
| 3.4.1 | Detailed Description | 9 |
| 3.4.2 | Field Documentation | 9 |
| 3.4.2.1 | ac | 9 |
| 3.4.2.2 | custom | 9 |
| 3.4.2.3 | datalen | 9 |
| 3.4.2.4 | date1 | 9 |
| 3.4.2.5 | date2 | 9 |
| 3.4.2.6 | fqn | 9 |
| 3.4.2.7 | holder | 9 |
| 3.4.2.8 | serial | 10 |
| 3.4.2.9 | server | 10 |
| 3.4.2.10 | serverca | 10 |
| 3.4.2.11 | siglen | 10 |
| 3.4.2.12 | signature | 10 |
| 3.4.2.13 | std | 10 |
| 3.4.2.14 | type | 10 |
| 3.4.2.15 | uri | 10 |
| 3.4.2.16 | user | 11 |
| 3.4.2.17 | userca | 11 |
| 3.4.2.18 | version | 11 |
| 3.4.2.19 | voname | 11 |
| 3.5 | vomsdata Struct Reference | 11 |
| 3.5.1 | Detailed Description | 11 |
| 3.5.2 | Field Documentation | 12 |
| 3.5.2.1 | cdir | 12 |
| 3.5.2.2 | data | 12 |
| 3.5.2.3 | extra_data | 12 |
| 3.5.2.4 | extralen | 12 |
| 3.5.2.5 | real | 12 |
| 3.5.2.6 | vdir | 12 |

| | | |
|----------|----------------------------|-----------|
| 3.5.2.7 | volen | 12 |
| 3.5.2.8 | workvo | 12 |
| 4 | File Documentation | 15 |
| 4.1 | voms_apic.h File Reference | 15 |
| 4.1.1 | Define Documentation | 18 |
| 4.1.1.1 | NOGLOBUS | 18 |
| 4.1.1.2 | RECURSE_CHAIN | 18 |
| 4.1.1.3 | RECURSE_NONE | 18 |
| 4.1.1.4 | TYPE_CUSTOM | 18 |
| 4.1.1.5 | TYPE_NODATA | 18 |
| 4.1.1.6 | TYPE_STD | 18 |
| 4.1.1.7 | VERIFY_CERTLIST | 18 |
| 4.1.1.8 | VERIFY_DATE | 18 |
| 4.1.1.9 | VERIFY_FULL | 18 |
| 4.1.1.10 | VERIFY_ID | 19 |
| 4.1.1.11 | VERIFY_KEY | 19 |
| 4.1.1.12 | VERIFY_NONE | 19 |
| 4.1.1.13 | VERIFY_NOTARGET | 19 |
| 4.1.1.14 | VERIFY_ORDER | 19 |
| 4.1.1.15 | VERIFY_SIGN | 19 |
| 4.1.1.16 | VERR_COMM | 19 |
| 4.1.1.17 | VERR_DIR | 19 |
| 4.1.1.18 | VERR_EXTRAINFO | 19 |
| 4.1.1.19 | VERR_FORMAT | 20 |
| 4.1.1.20 | VERR_IDCHECK | 20 |
| 4.1.1.21 | VERR_MEM | 20 |
| 4.1.1.22 | VERR_NODATA | 20 |
| 4.1.1.23 | VERR_NOEXT | 20 |
| 4.1.1.24 | VERR_NOIDENT | 20 |
| 4.1.1.25 | VERR_NOINIT | 20 |
| 4.1.1.26 | VERR_NONE | 20 |
| 4.1.1.27 | VERR_NOSOCKET | 21 |
| 4.1.1.28 | VERR_NOTAVAIL | 21 |

| | | |
|----------|------------------------|----|
| 4.1.1.29 | VERR_ORDER | 21 |
| 4.1.1.30 | VERR_PARAM | 21 |
| 4.1.1.31 | VERR_PARSE | 21 |
| 4.1.1.32 | VERR_SERVER | 21 |
| 4.1.1.33 | VERR_SERVERCODE | 21 |
| 4.1.1.34 | VERR_SIGN | 21 |
| 4.1.1.35 | VERR_TIME | 22 |
| 4.1.1.36 | VERR_TYPE | 22 |
| 4.1.1.37 | VERR_VERIFY | 22 |
| 4.1.2 | Typedef Documentation | 22 |
| 4.1.2.1 | gss_cred_id_t | 22 |
| 4.1.2.2 | gss_ctx_id_t | 22 |
| 4.1.3 | Function Documentation | 22 |
| 4.1.3.1 | getMajorVersionNumber | 22 |
| 4.1.3.2 | getMinorVersionNumber | 22 |
| 4.1.3.3 | getPatchVersionNumber | 22 |
| 4.1.3.4 | VOMS_AddTarget | 22 |
| 4.1.3.5 | VOMS_Contact | 23 |
| 4.1.3.6 | VOMS_ContactRaw | 23 |
| 4.1.3.7 | VOMS_Copy | 24 |
| 4.1.3.8 | VOMS_CopyAll | 24 |
| 4.1.3.9 | VOMS_DefaultData | 24 |
| 4.1.3.10 | VOMS_Delete | 24 |
| 4.1.3.11 | VOMS_DeleteContacts | 25 |
| 4.1.3.12 | VOMS_Destroy | 25 |
| 4.1.3.13 | VOMS_Duplicate | 25 |
| 4.1.3.14 | VOMS_ErrorMessage | 25 |
| 4.1.3.15 | VOMS_Export | 25 |
| 4.1.3.16 | VOMS_FindByAlias | 26 |
| 4.1.3.17 | VOMS_FindByVO | 26 |
| 4.1.3.18 | VOMS_FreeTargets | 26 |
| 4.1.3.19 | VOMS_FreeTargetsList | 27 |
| 4.1.3.20 | VOMS_GetAC | 27 |
| 4.1.3.21 | VOMS_GetAttribute | 27 |

| | | |
|----------|--------------------------------|----|
| 4.1.3.22 | VOMS_GetAttributeGrantor | 27 |
| 4.1.3.23 | VOMS_GetAttributesNumber | 27 |
| 4.1.3.24 | VOMS_GetAttributeSourceHandle | 27 |
| 4.1.3.25 | VOMS_GetAttributeSourcesNumber | 27 |
| 4.1.3.26 | VOMS_GetTargetsList | 27 |
| 4.1.3.27 | VOMS_Import | 27 |
| 4.1.3.28 | VOMS_Init | 27 |
| 4.1.3.29 | VOMS_ListTargets | 28 |
| 4.1.3.30 | VOMS_LoadCredentials | 28 |
| 4.1.3.31 | VOMS_Ordering | 28 |
| 4.1.3.32 | VOMS_ResetOrder | 28 |
| 4.1.3.33 | VOMS_Retrieve | 29 |
| 4.1.3.34 | VOMS_RetrieveEXT | 29 |
| 4.1.3.35 | VOMS_RetrieveFromAC | 29 |
| 4.1.3.36 | VOMS_RetrieveFromCred | 30 |
| 4.1.3.37 | VOMS_RetrieveFromCtx | 30 |
| 4.1.3.38 | VOMS_RetrieveFromFile | 30 |
| 4.1.3.39 | VOMS_RetrieveFromProxy | 31 |
| 4.1.3.40 | VOMS_SetLifetime | 31 |
| 4.1.3.41 | VOMS_SetTimeout | 31 |
| 4.1.3.42 | VOMS_SetVerificationTime | 31 |
| 4.1.3.43 | VOMS_SetVerificationType | 31 |

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

| | |
|---|----|
| attribute | 5 |
| contactdata (The type of data returned) | 6 |
| data (User's characteristics: can be repeated) | 7 |
| voms | 8 |
| vomsdata | 11 |

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

| | |
|-----------------------------|----|
| voms_apic.h | 15 |
|-----------------------------|----|

Chapter 3

Data Structure Documentation

3.1 attribute Struct Reference

```
#include <voms_apic.h>
```

Data Fields

- const char * [name](#)
- const char * [value](#)
- const char * [qualifier](#)

3.1.1 Detailed Description

Definition at line 54 of file voms_apic.h.

3.1.2 Field Documentation

3.1.2.1 const char* attribute::name

Definition at line 55 of file voms_apic.h.

3.1.2.2 const char* attribute::qualifier

Definition at line 57 of file voms_apic.h.

3.1.2.3 const char* attribute::value

Definition at line 56 of file voms_apic.h.

The documentation for this struct was generated from the following file:

- [voms_apic.h](#)

3.2 contactdata Struct Reference

The type of data returned.

```
#include <voms_apic.h>
```

Data Fields

- char * [nick](#)
- char * [host](#)
- char * [contact](#)
- char * [vo](#)
- int [port](#)
- char * [reserved](#)
- int [version](#)

3.2.1 Detailed Description

The type of data returned.

Definition at line 60 of file voms_apic.h.

3.2.2 Field Documentation

3.2.2.1 char* contactdata::contact

The subject of the server's certificate

Definition at line 65 of file voms_apic.h.

3.2.2.2 char* contactdata::host

The hostname of the server

Definition at line 64 of file voms_apic.h.

3.2.2.3 char* contactdata::nick

< You must never allocate directly this structure. Its sizeof() is subject to change without notice. The only supported way to obtain it is via the VOMS_FindBy* functions. The alias of the server

Definition at line 63 of file voms_apic.h.

3.2.2.4 int contactdata::port

The port on which the server is listening

Definition at line 67 of file voms_apic.h.

3.2.2.5 char* contactdata::reserved

HANDS OFF!

Definition at line 68 of file voms_apic.h.

3.2.2.6 int contactdata::version

The version of Globus on which this server runs.

Definition at line 69 of file voms_apic.h.

3.2.2.7 char* contactdata::vo

The VO served by this server

Definition at line 66 of file voms_apic.h.

The documentation for this struct was generated from the following file:

- [voms_apic.h](#)

3.3 data Struct Reference

User's characteristics: can be repeated.

```
#include <voms_apic.h>
```

Data Fields

- char * [group](#)
- char * [role](#)
- char * [cap](#)

3.3.1 Detailed Description

User's characteristics: can be repeated.

<

Definition at line 48 of file voms_apic.h.

3.3.2 Field Documentation

3.3.2.1 `char* data::cap`

user's capability

Definition at line 51 of file voms_apic.h.

3.3.2.2 `char* data::group`

user's group

Definition at line 49 of file voms_apic.h.

3.3.2.3 `char* data::role`

user's role

Definition at line 50 of file voms_apic.h.

The documentation for this struct was generated from the following file:

- [voms_apic.h](#)

3.4 voms Struct Reference

```
#include <voms_apic.h>
```

Data Fields

- int [siglen](#)
- char * [signature](#)
- char * [user](#)
- char * [userca](#)
- char * [server](#)
- char * [serverca](#)
- char * [voname](#)
- char * [uri](#)
- char * [date1](#)
- char * [date2](#)
- int [type](#)
- struct [data](#) ** [std](#)
- char * [custom](#)
- int [datalen](#)
- int [version](#)
- char ** [fqan](#)

- char * [serial](#)
- AC * [ac](#)
- X509 * [holder](#)

3.4.1 Detailed Description

Definition at line 80 of file voms_apic.h.

3.4.2 Field Documentation

3.4.2.1 AC* voms::ac

Definition at line 100 of file voms_apic.h.

3.4.2.2 char* voms::custom

The data returned by an S command

Definition at line 93 of file voms_apic.h.

3.4.2.3 int voms::datalen

Definition at line 94 of file voms_apic.h.

3.4.2.4 char* voms::date1

Beginning of validity of the user info

Definition at line 89 of file voms_apic.h.

3.4.2.5 char* voms::date2

End of validity of the user info

Definition at line 90 of file voms_apic.h.

3.4.2.6 char** voms::fqan

User's attributes in compact format

Definition at line 96 of file voms_apic.h.

3.4.2.7 X509* voms::holder

Definition at line 101 of file voms_apic.h.

3.4.2.8 char* voms::serial

Serial number. Only significant if coming from AC. Null otherwise

Definition at line 97 of file voms_apic.h.

3.4.2.9 char* voms::server

The VOMS server DN, as from its certificate

Definition at line 85 of file voms_apic.h.

3.4.2.10 char* voms::serverca

The CA which signed the VOMS certificate

Definition at line 86 of file voms_apic.h.

3.4.2.11 int voms::siglen

The length of the VOMS server signature

Definition at line 81 of file voms_apic.h.

3.4.2.12 char* voms::signature

The VOMS server signature

Definition at line 82 of file voms_apic.h.

3.4.2.13 struct data voms::std**

User's characteristics

Definition at line 92 of file voms_apic.h.

3.4.2.14 int voms::type

The type of data returned

Definition at line 91 of file voms_apic.h.

3.4.2.15 char* voms::uri

The URI of the VOMS server

Definition at line 88 of file voms_apic.h.

3.4.2.16 char* voms::user

The user's DN, as from his certificate

Definition at line 83 of file voms_apic.h.

3.4.2.17 char* voms::userca

The CA which signed the user's certificate

Definition at line 84 of file voms_apic.h.

3.4.2.18 int voms::version

Definition at line 95 of file voms_apic.h.

3.4.2.19 char* voms::voname

The name of the VO to which the VOMS belongs

Definition at line 87 of file voms_apic.h.

The documentation for this struct was generated from the following file:

- [voms_apic.h](#)

3.5 vomsdata Struct Reference

```
#include <voms_apic.h>
```

Data Fields

- char * [cdir](#)
- char * [vdir](#)
- struct [voms](#) ** [data](#)
- char * [workvo](#)
- char * [extra_data](#)
- int [volen](#)
- int [extralen](#)
- struct [vomsdata](#) * [real](#)

3.5.1 Detailed Description

Definition at line 144 of file voms_apic.h.

3.5.2 Field Documentation

3.5.2.1 `char* vomsdata::cdir`

Definition at line 145 of file voms_apic.h.

3.5.2.2 `struct voms** vomsdata::data`

User's info, as in the certificate extension. It may contain data gathered from more than one VOMS server,

Definition at line 147 of file voms_apic.h.

3.5.2.3 `char* vomsdata::extra_data`

The data specified by the user with the --include switch. Note that this field doesn't contain the result of a request to the VOMS server, but instead data specified by the user. The reason for the introduction of this extension is to let a user include important data into his proxy certificate, like, for example, a kerberos ticket

Definition at line 152 of file voms_apic.h.

3.5.2.4 `int vomsdata::extralen`

Definition at line 162 of file voms_apic.h.

3.5.2.5 `struct vomsdata* vomsdata::real`

Definition at line 164 of file voms_apic.h.

3.5.2.6 `char* vomsdata::vdir`

Definition at line 146 of file voms_apic.h.

3.5.2.7 `int vomsdata::volen`

Definition at line 161 of file voms_apic.h.

3.5.2.8 `char* vomsdata::workvo`

The value of the -vo option of the voms-proxy-init command

Definition at line 150 of file voms_apic.h.

The documentation for this struct was generated from the following file:

- [voms_apic.h](#)

Chapter 4

File Documentation

4.1 voms_apic.h File Reference

```
#include <openssl/x509.h>
#include <time.h>
#include "newformat.h"
```

Data Structures

- struct [data](#)
User's characteristics: can be repeated.
- struct [attribute](#)
- struct [contactdata](#)
The type of data returned.
- struct [voms](#)
- struct [vomsdata](#)

Defines

- #define [NOGLOBUS](#)
- #define [TYPE_NODATA](#) 0
- #define [TYPE_STD](#) 1
- #define [TYPE_CUSTOM](#) 2
- #define [RECURSE_CHAIN](#) 0
- #define [RECURSE_NONE](#) 1
- #define [VERIFY_FULL](#) 0xffffffff
- #define [VERIFY_NONE](#) 0x00000000
- #define [VERIFY_DATE](#) 0x00000001
- #define [VERIFY_NOTARGET](#) 0x00000002
- #define [VERIFY_KEY](#) 0x00000004

- #define [VERIFY_SIGN](#) 0x00000008
- #define [VERIFY_ORDER](#) 0x00000010
- #define [VERIFY_ID](#) 0x00000020
- #define [VERIFY_CERTLIST](#) 0x00000040
- #define [VERR_NONE](#) 0

Error codes.

- #define [VERR_NOSOCKET](#) 1
- #define [VERR_NOIDENT](#) 2
- #define [VERR_COMM](#) 3
- #define [VERR_PARAM](#) 4
- #define [VERR_NOEXT](#) 5
- #define [VERR_NOINIT](#) 6
- #define [VERR_TIME](#) 7
- #define [VERR_IDCHECK](#) 8
- #define [VERR_EXTRAINFO](#) 9
- #define [VERR_FORMAT](#) 10
- #define [VERR_NODATA](#) 11
- #define [VERR_PARSE](#) 12
- #define [VERR_DIR](#) 13
- #define [VERR_SIGN](#) 14
- #define [VERR_SERVER](#) 15
- #define [VERR_MEM](#) 16
- #define [VERR_VERIFY](#) 17
- #define [VERR_TYPE](#) 18
- #define [VERR_ORDER](#) 19
- #define [VERR_SERVERCODE](#) 20
- #define [VERR_NOTAVAIL](#) 21

Typedefs

- typedef void * [gss_cred_id_t](#)
- typedef void * [gss_ctx_id_t](#)

Functions

- struct [contactdata](#) ** [VOMS_FindByAlias](#) (struct [vomsdata](#) *vd, char *alias, char *system, char *user, int *error)
- struct [contactdata](#) ** [VOMS_FindByVO](#) (struct [vomsdata](#) *vd, char *vo, char *system, char *user, int *error)
- void [VOMS_DeleteContacts](#) (struct [contactdata](#) **list)
- struct [vomsdata](#) * [VOMS_Init](#) (char *voms, char *cert)
- struct [voms](#) * [VOMS_Copy](#) (struct [voms](#) *v, int *error)
- struct [vomsdata](#) * [VOMS_CopyAll](#) (struct [vomsdata](#) *vd, int *error)
- void [VOMS_Delete](#) (struct [voms](#) *v)
- int [VOMS_AddTarget](#) (struct [vomsdata](#) *vd, char *target, int *error)

- void [VOMS_FreeTargets](#) (struct [vomsdata](#) *vd, int *error)
- char * [VOMS_ListTargets](#) (struct [vomsdata](#) *vd, int *error)
- int [VOMS_SetVerificationType](#) (int type, struct [vomsdata](#) *vd, int *error)
- int [VOMS_SetLifetime](#) (int length, struct [vomsdata](#) *vd, int *error)
- void [VOMS_Destroy](#) (struct [vomsdata](#) *vd)
- int [VOMS_ResetOrder](#) (struct [vomsdata](#) *vd, int *error)
- int [VOMS_Ordering](#) (char *order, struct [vomsdata](#) *vd, int *error)
- int [VOMS_Contact](#) (char *hostname, int port, char *servsubject, char *command, struct [vomsdata](#) *vd, int *error)
- int [VOMS_ContactRaw](#) (char *hostname, int port, char *servsubject, char *command, void **data, int *datalen, int *version, struct [vomsdata](#) *vd, int *error)
- int [VOMS_Retrieve](#) (X509 *cert, STACK_OF(X509)*chain, int how, struct [vomsdata](#) *vd, int *error)
- int [VOMS_Import](#) (char *buffer, int buflen, struct [vomsdata](#) *vd, int *error)
- int [VOMS_Export](#) (char **buffer, int *buflen, struct [vomsdata](#) *vd, int *error)
- struct [voms](#) * [VOMS_DefaultData](#) (struct [vomsdata](#) *vd, int *error)
- char * [VOMS_ErrorMessage](#) (struct [vomsdata](#) *vd, int error, char *buffer, int len)
- int [VOMS_RetrieveEXT](#) (X509_EXTENSION *ext, struct [vomsdata](#) *vd, int *error)
- int [VOMS_RetrieveFromCred](#) (gss_cred_id_t cred, int how, struct [vomsdata](#) *vd, int *error)
- int [VOMS_RetrieveFromFile](#) (FILE *file, int how, struct [vomsdata](#) *vd, int *error)
- int [VOMS_RetrieveFromCtx](#) (gss_ctx_id_t ctx, int how, struct [vomsdata](#) *vd, int *error)
- int [VOMS_RetrieveFromProxy](#) (int how, struct [vomsdata](#) *vd, int *error)
- int [VOMS_RetrieveFromAC](#) (AC *ac, struct [vomsdata](#) *vd, int *error)
- int [VOMS_GetAttributeSourcesNumber](#) (struct [voms](#) *v, struct [vomsdata](#) *vd, int *error)
- int [VOMS_GetAttributeSourceHandle](#) (struct [voms](#) *v, int num, struct [vomsdata](#) *vd, int *error)
- const char * [VOMS_GetAttributeGrantor](#) (struct [voms](#) *v, int handle, struct [vomsdata](#) *vd, int *error)
- int [VOMS_GetAttributesNumber](#) (struct [voms](#) *v, int handle, struct [vomsdata](#) *vd, int *error)
- int [VOMS_GetAttribute](#) (struct [voms](#) *v, int handle, int num, struct [attribute](#) *at, struct [vomsdata](#) *vd, int *error)
- struct [vomsdata](#) * [VOMS_Duplicate](#) (struct [vomsdata](#) *vd)
- AC * [VOMS_GetAC](#) (struct [voms](#) *v)
- int [getMajorVersionNumber](#) (void)
- int [getMinorVersionNumber](#) (void)
- int [getPatchVersionNumber](#) (void)
- int [VOMS_SetVerificationTime](#) (time_t verificationtime, struct [vomsdata](#) *vd, int *error)
- char ** [VOMS_GetTargetsList](#) (struct [voms](#) *v, struct [vomsdata](#) *vd, int *error)
- void [VOMS_FreeTargetsList](#) (char **)
- int [VOMS_SetTimeout](#) (int t, struct [vomsdata](#) *vd, int *error)
- int [VOMS_LoadCredentials](#) (X509 *cert, EVP_PKEY *pkey, STACK_OF(X509)*chain, struct [vomsdata](#) *vd, int *error)

4.1.1 Define Documentation

4.1.1.1 `#define NOGLOBUS`

Definition at line 33 of file voms_apic.h.

4.1.1.2 `#define RECURSE_CHAIN 0`

Definition at line 105 of file voms_apic.h.

4.1.1.3 `#define RECURSE_NONE 1`

Definition at line 106 of file voms_apic.h.

4.1.1.4 `#define TYPE_CUSTOM 2`

result of an S command

Definition at line 77 of file voms_apic.h.

4.1.1.5 `#define TYPE_NODATA 0`

no data

Definition at line 75 of file voms_apic.h.

4.1.1.6 `#define TYPE_STD 1`

group, role, capability triplet

Definition at line 76 of file voms_apic.h.

4.1.1.7 `#define VERIFY_CERTLIST 0x00000040`

Definition at line 116 of file voms_apic.h.

4.1.1.8 `#define VERIFY_DATE 0x00000001`

Definition at line 110 of file voms_apic.h.

4.1.1.9 `#define VERIFY_FULL 0xffffffff`

Definition at line 108 of file voms_apic.h.

4.1.1.10 #define VERIFY_ID 0x00000020

Definition at line 115 of file voms_apic.h.

4.1.1.11 #define VERIFY_KEY 0x00000004

Definition at line 112 of file voms_apic.h.

4.1.1.12 #define VERIFY_NONE 0x00000000

Definition at line 109 of file voms_apic.h.

4.1.1.13 #define VERIFY_NOTARGET 0x00000002

Definition at line 111 of file voms_apic.h.

4.1.1.14 #define VERIFY_ORDER 0x00000010

Definition at line 114 of file voms_apic.h.

4.1.1.15 #define VERIFY_SIGN 0x00000008

Definition at line 113 of file voms_apic.h.

4.1.1.16 #define VERR_COMM 3

Server problem

Definition at line 123 of file voms_apic.h.

4.1.1.17 #define VERR_DIR 13

Directory error

Definition at line 134 of file voms_apic.h.

4.1.1.18 #define VERR_EXTRAINFO 9

VO name and URI missing

Definition at line 130 of file voms_apic.h.

4.1.1.19 #define VERR_FORMAT 10

Wrong data format

Definition at line 131 of file voms_apic.h.

4.1.1.20 #define VERR_IDCHECK 8

User data in extension different from the real ones

Definition at line 128 of file voms_apic.h.

4.1.1.21 #define VERR_MEM 16

Memory problems

Definition at line 137 of file voms_apic.h.

4.1.1.22 #define VERR_NODATA 11

Empty extension

Definition at line 132 of file voms_apic.h.

4.1.1.23 #define VERR_NOEXT 5

VOMS extension missing

Definition at line 125 of file voms_apic.h.

4.1.1.24 #define VERR_NOIDENT 2

Cannot identify itself (certificate problem)

Definition at line 122 of file voms_apic.h.

4.1.1.25 #define VERR_NOINIT 6

Initialization error

Definition at line 126 of file voms_apic.h.

4.1.1.26 #define VERR_NONE 0

Error codes.

Definition at line 120 of file voms_apic.h.

4.1.1.27 #define VERR_NOSOCKET 1

Socket problem

Definition at line 121 of file voms_apic.h.

4.1.1.28 #define VERR_NOTAVAIL 21

Method not available

Definition at line 142 of file voms_apic.h.

4.1.1.29 #define VERR_ORDER 19

Ordering different than required

Definition at line 140 of file voms_apic.h.

4.1.1.30 #define VERR_PARAM 4

Wrong parameters

Definition at line 124 of file voms_apic.h.

4.1.1.31 #define VERR_PARSE 12

Parse error

Definition at line 133 of file voms_apic.h.

4.1.1.32 #define VERR_SERVER 15

Unidentifiable VOMS server

Definition at line 136 of file voms_apic.h.

4.1.1.33 #define VERR_SERVERCODE 20

Error from the server

Definition at line 141 of file voms_apic.h.

4.1.1.34 #define VERR_SIGN 14

Signature error

Definition at line 135 of file voms_apic.h.

4.1.1.35 #define VERR_TIME 7

Error in time checking

Definition at line 127 of file voms_apic.h.

4.1.1.36 #define VERR_TYPE 18

Returned data of unknown type

Definition at line 139 of file voms_apic.h.

4.1.1.37 #define VERR_VERIFY 17

Generic verification error

Definition at line 138 of file voms_apic.h.

4.1.2 Typedef Documentation

4.1.2.1 typedef void* gss_cred_id_t

Definition at line 35 of file voms_apic.h.

4.1.2.2 typedef void* gss_ctx_id_t

Definition at line 36 of file voms_apic.h.

4.1.3 Function Documentation

4.1.3.1 int getMajorVersionNumber (void)

4.1.3.2 int getMinorVersionNumber (void)

4.1.3.3 int getPatchVersionNumber (void)

4.1.3.4 int VOMS_AddTarget (struct vomsdata * *vd*, char * *target*, int * *error*)

Adds a target to the AC.

Parameters

| | |
|---------------|--|
| <i>vd</i> | The vomsdata structure. |
| <i>target</i> | The target to add. It should be a FQDN. |
| <i>error</i> | RETURN PARAMETER: qualifies the error message. |

Returns

failure (0) or success (<>0)

4.1.3.5 `int VOMS.Contact (char * hostname, int port, char * servsubject, char * command, struct vomsdata * vd, int * error)`

Contacts a VOMS server to get a certificate

It is the equivalent of the voms_proxy_init command, but without the --include functionality.

Parameters

| | |
|--------------------|--|
| <i>hostname</i> | FQDN of the VOMS server |
| <i>port</i> | the port on which the VOMS server is listening |
| <i>servsubject</i> | the subject of the server's certificate |
| <i>command</i> | Command |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.6 `int VOMS.ContactRaw (char * hostname, int port, char * servsubject, char * command, void ** data, int * datalen, int * version, struct vomsdata * vd, int * error)`

The same as VOMS_Contract, except that instead of starting the verification process, the data is returned as is in the

Parameters

| | |
|--------------------|--|
| <i>data</i> | and |
| <i>datalen</i> | fields. |
| <i>hostname</i> | FQDN of the VOMS server |
| <i>port</i> | the port on which the VOMS server is listening |
| <i>servsubject</i> | the subject of the server's certificate |
| <i>command</i> | the command sent to the server |
| <i>version</i> | is the version number of the data. |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.7 struct voms* VOMS_Copy (struct voms * *v*, int * *error*) [read]

Copies a voms structure. N.B: This is the ONLY way to correctly initialize a voms structure as a copy of another voms structure.

Parameters

| | |
|--------------|--|
| <i>v</i> | The structure to copy. |
| <i>error</i> | RETURN PARAMETER: qualifies the error message. |

Returns

NULL (error) or the new voms structure.

4.1.3.8 struct vomsdata* VOMS_CopyAll (struct vomsdata * *vd*, int * *error*) [read]

Copies a vomsdata structure. N.B: This is the ONLY way to correctly initialize a vomsdata structure as a copy of another vomsdata structure.

Parameters

| | |
|--------------|--|
| <i>vd</i> | The structure to copy. |
| <i>error</i> | RETURN PARAMETER: qualifies the error message. |

Returns

NULL (error) or the new vomsdata structure.

4.1.3.9 struct voms* VOMS_DefaultData (struct vomsdata * *vd*, int * *error*) [read]

Gets the default attributes from a vomsdata structure.

Parameters

| | |
|--------------|---|
| <i>vd</i> | the vomsdata structure to analyze |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

a pointer to the relevant voms structure. DO NOT modify the fields.

4.1.3.10 void VOMS_Delete (struct voms * *v*)

Deletes a voms structure

Parameters

| | |
|----------|-------------------------------------|
| <i>v</i> | Pointer to the structure to delete. |
|----------|-------------------------------------|

4.1.3.11 void VOMS_DeleteContacts (struct contactdata ** *list*)

Frees a contactdata vector.

Parameters

| | |
|-------------|---------------------|
| <i>list</i> | The vector to free. |
|-------------|---------------------|

Returns

NONE

4.1.3.12 void VOMS_Destroy (struct vomsdata * *vd*)

Destroys a proper vomsdata structure /param *vd* The structure to deallocate.

4.1.3.13 struct vomsdata* VOMS_Duplicate (struct vomsdata * *vd*) [read]**4.1.3.14 char* VOMS_ErrorMessage (struct vomsdata * *vd*, int *error*, char * *buffer*, int *len*)**

Gets a textual description of the error.

Parameters

| | |
|---------------|---|
| <i>vd</i> | The vomsdata structure to analyze |
| <i>error</i> | The error returned by the last function |
| <i>buffer</i> | A pointer to a buffer where the error message will be written. If NULL, then memory is allocated by the function, and will have to be free()ed by the caller. |
| <i>len</i> | The length of the memory pointed to by the buffer parameter. |

Returns

A pointer to the error message.

4.1.3.15 int VOMS_Export (char ** *buffer*, int * *buflen*, struct vomsdata * *vd*, int * *error*)

Converts data into a test format

Parameters

| | |
|---------------|--|
| <i>buffer</i> | OUTPUT PARAMETER contains the converted data |
| <i>buflen</i> | OUTPUT PARAMETER contains the length of buffer |
| <i>vd</i> | contains the data to convert |
| <i>error</i> | RETURN PARAMETER Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.16 `struct contactdata** VOMS_FindByAlias (struct vomsdata * vd, char * alias,
char * system, char * user, int * error)` [read]

Gets a list of VOMS servers which share an alias.

Parameters

| | |
|---------------|---|
| <i>vd</i> | The correctly initialized vomsdata structured. |
| <i>alias</i> | The alias to look for. |
| <i>system</i> | The directory in which to look for the system configuration files. If NULL, defaults to /opt/edc/etc/vomses |
| <i>user</i> | The directory in which to look for the user configuration files. Defaults to \$VOMS_USERCONF if NULL. Again defaults to \$HOME/.edg/vomses if the latter is NULL, or to ~/.edg/vomses as a last resort. |
| <i>error</i> | RETURN PARAMETER: qualifies the error message. |

Returns

NULL, or a NULL-terminated vector of contactdata structures. The only supported way to free this array is via the VOMS_DeleteContacts function. Note also that the order in which the servers are returned is unspecified.

4.1.3.17 `struct contactdata** VOMS_FindByVO (struct vomsdata * vd, char * vo, char *
system, char * user, int * error)` [read]

Gets a list of VOMS servers which serve the same VO.

Parameters

| | |
|---------------|---|
| <i>vd</i> | The correctly initialized vomsdata structured. |
| <i>vo</i> | The VO to look for. |
| <i>system</i> | The directory in which to look for the system configuration files. If NULL, defaults to /opt/edc/etc/vomses |
| <i>user</i> | The directory in which to look for the user configuration files. Defaults to \$VOMS_USERCONF if NULL. Again defaults to \$HOME/.edg/vomses if the latter is NULL, or to ~/.edg/vomses as a last resort. |
| <i>error</i> | RETURN PARAMETER: qualifies the error message. |

Returns

NULL, or a NULL-terminated vector of contactdata structures. The only supported way to free this array is via the VOMS_DeleteContacts function. Note also that the order in which the servers are returned is unspecified.

4.1.3.18 `void VOMS_FreeTargets (struct vomsdata * vd, int * error)`

Delete the targets from the AC.

Parameters

| | |
|--------------|--|
| <i>vd</i> | The vomsdata structure. |
| <i>error</i> | RETURN PARAMETER: qualifies the error message. |

4.1.3.19 void VOMS_FreeTargetsList (char **)

4.1.3.20 AC* VOMS_GetAC (struct voms * v)

4.1.3.21 int VOMS_GetAttribute (struct voms * v, int *handle*, int *num*, struct attribute * *at*, struct vomsdata * *vd*, int * *error*)

4.1.3.22 const char* VOMS_GetAttributeGrantor (struct voms * v, int *handle*, struct vomsdata * *vd*, int * *error*)

4.1.3.23 int VOMS_GetAttributesNumber (struct voms * v, int *handle*, struct vomsdata * *vd*, int * *error*)

4.1.3.24 int VOMS_GetAttributeSourceHandle (struct voms * v, int *num*, struct vomsdata * *vd*, int * *error*)

4.1.3.25 int VOMS_GetAttributeSourcesNumber (struct voms * v, struct vomsdata * *vd*, int * *error*)

4.1.3.26 char** VOMS_GetTargetsList (struct voms * v, struct vomsdata * *vd*, int * *error*)

4.1.3.27 int VOMS_Import (char * *buffer*, int *buflen*, struct vomsdata * *vd*, int * *error*)

Converts data from the format used for inclusion into a certificate to the internal format
The function does verify the data.

Parameters

| | |
|---------------|--|
| <i>buffer</i> | contains the data to be converted |
| <i>buflen</i> | contains the length of buffer |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.28 struct vomsdata* VOMS_Init (char * *voms*, char * *cert*) [read]

Initializes a vomsdata structure for use by the other functions. N.B: This is the ONLY way to correctly initialize a vomsdata structure. It is also forbidden to directly take the sizeof() of this structure.

Parameters

| | |
|-------------|---|
| <i>voms</i> | The directory which contains the certificates of the VOMS servers |
| <i>cert</i> | The directory which contains the CA certificates |

If voms_dir is empty, the value of the environment variable X509_VOMS_DIR is taken

If cert_dir is empty, the value of the environment variable X509_CERT_DIR is taken

Returns

NULL for failure, or a pointer to a properly initialized structure.

4.1.3.29 `char* VOMS_ListTargets (struct vomsdata * vd, int * error)`

4.1.3.30 `int VOMS_LoadCredentials (X509 * cert, EVP_PKEY * pkey, STACK_OF(X509)* chain, struct vomsdata * vd, int * error)`

4.1.3.31 `int VOMS_Ordering (char * order, struct vomsdata * vd, int * error)`

Further specified the order of the returned attributes. Please do note that calls are cumulative unless [VOMS_ResetOrder\(\)](#) is called.

Parameters

| | |
|--------------|---|
| <i>order</i> | the group:role attribute. |
| <i>vd</i> | RETURN PARAMETER: contains the modified data. |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.32 `int VOMS_ResetOrder (struct vomsdata * vd, int * error)`

Unsets the return order of the attributes.

Parameters

| | |
|--------------|---|
| <i>vd</i> | RETURN PARAMETER: contains the modified data. |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.33 int VOMS_Retrieve (X509 * *cert*, STACK_OF(X509)* *chain*, int *how*, struct vomsdata * *vd*, int * *error*)

Extracts the VOMS extension from an X.509 certificate.

The function doesn't check the validity of the certificates, but it does check the content of the user data.

Parameters

| | |
|--------------|---|
| <i>cert</i> | The certificate with the VOMS extensions |
| <i>chain</i> | The chain of the validation certificates (only the intermediate ones) |
| <i>how</i> | Recursion type |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.34 int VOMS_RetrieveEXT (X509_EXTENSION * *ext*, struct vomsdata * *vd*, int * *error*)

Gets VOMS information from the given extension

Parameters

| | |
|--------------|--|
| <i>ext</i> | The extension to parse. |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.35 int VOMS_RetrieveFromAC (AC * *ac*, struct vomsdata * *vd*, int * *error*)

Gets VOMS information from an existing globus proxy

Parameters

| | |
|--------------|--|
| <i>ac</i> | AC from which to get the credentials |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.36 `int VOMS.RetrieveFromCred (gss_cred_id_t cred, int how, struct vomsdata * vd, int * error)`

Gets VOMS information from the given globus credential

Parameters

| | |
|--------------|--|
| <i>cred</i> | The credential from which to retrieve the certificate. |
| <i>how</i> | Recursion type |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.37 `int VOMS.RetrieveFromCtx (gss_ctx_id_t ctx, int how, struct vomsdata * vd, int * error)`

Gets VOMS information from the given globus context

Parameters

| | |
|--------------|--|
| <i>ctx</i> | The context from which to retrieve the certificate. |
| <i>how</i> | Recursion type |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.38 `int VOMS.RetrieveFromFile (FILE * file, int how, struct vomsdata * vd, int * error)`

Gets VOMS information from the given globus credential

Parameters

| | |
|--------------|--|
| <i>file</i> | The file from which to retrieve the certificate. |
| <i>how</i> | Recursion type |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.39 int VOMS_RetrieveFromProxy (int *how*, struct vomsdata * *vd*, int * *error*)

Gets VOMS information from an existing globus proxy

Parameters

| | |
|--------------|--|
| <i>how</i> | Recursion type |
| <i>vd</i> | RETURN PARAMETER: contains the data returned by the connection |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.40 int VOMS_SetLifetime (int *length*, struct vomsdata * *vd*, int * *error*)

Set requested lifetime for [VOMS_Contact\(\)](#) calls.

Parameters

| | |
|---------------|---|
| <i>length</i> | Lifetime requested. |
| <i>vd</i> | RETURN PARAMETER: contains the modified data. |
| <i>error</i> | RETURN PARAMETER: Qualifies the error message |

Returns

failure (0) or success (<>0)

4.1.3.41 int VOMS_SetTimeout (int *t*, struct vomsdata * *vd*, int * *error*)**4.1.3.42 int VOMS_SetVerificationTime (time_t *verificationtime*, struct vomsdata * *vd*, int * *error*)****4.1.3.43 int VOMS_SetVerificationType (int *type*, struct vomsdata * *vd*, int * *error*)**

Index

ac
 voms, [9](#)
attribute, [5](#)
 name, [5](#)
 qualifier, [5](#)
 value, [5](#)

cap
 data, [8](#)
cdir
 vomldata, [12](#)
contact
 contactdata, [6](#)
contactdata, [6](#)
 contact, [6](#)
 host, [6](#)
 nick, [6](#)
 port, [6](#)
 reserved, [7](#)
 version, [7](#)
 vo, [7](#)
custom
 voms, [9](#)

data, [7](#)
 cap, [8](#)
 group, [8](#)
 role, [8](#)
 vomldata, [12](#)
datalen
 voms, [9](#)
date1
 voms, [9](#)
date2
 voms, [9](#)

extra_data
 vomldata, [12](#)
extralen
 vomldata, [12](#)

fqan
 voms, [9](#)

getMajorVersionNumber
 voms_apic.h, [22](#)
getMinorVersionNumber
 voms_apic.h, [22](#)
getPatchVersionNumber
 voms_apic.h, [22](#)
group
 data, [8](#)
gss_cred_id_t
 voms_apic.h, [22](#)
gss_ctx_id_t
 voms_apic.h, [22](#)

holder
 voms, [9](#)
host
 contactdata, [6](#)

name
 attribute, [5](#)
nick
 contactdata, [6](#)
NOGLOBUS
 voms_apic.h, [18](#)

port
 contactdata, [6](#)

qualifier
 attribute, [5](#)

real
 vomldata, [12](#)
RECURSE_CHAIN
 voms_apic.h, [18](#)
RECURSE_NONE
 voms_apic.h, [18](#)
reserved
 contactdata, [7](#)
role

- data, [8](#)
- serial
 - voms, [9](#)
- server
 - voms, [10](#)
- serverca
 - voms, [10](#)
- siglen
 - voms, [10](#)
- signature
 - voms, [10](#)
- std
 - voms, [10](#)
- type
 - voms, [10](#)
- TYPE_CUSTOM
 - voms_apic.h, [18](#)
- TYPE_NODATA
 - voms_apic.h, [18](#)
- TYPE_STD
 - voms_apic.h, [18](#)
- uri
 - voms, [10](#)
- user
 - voms, [10](#)
- userca
 - voms, [11](#)
- value
 - attribute, [5](#)
- vdir
 - vomsdata, [12](#)
- VERIFY_CERTLIST
 - voms_apic.h, [18](#)
- VERIFY_DATE
 - voms_apic.h, [18](#)
- VERIFY_FULL
 - voms_apic.h, [18](#)
- VERIFY_ID
 - voms_apic.h, [18](#)
- VERIFY_KEY
 - voms_apic.h, [19](#)
- VERIFY_NONE
 - voms_apic.h, [19](#)
- VERIFY_NOTARGET
 - voms_apic.h, [19](#)
- VERIFY_ORDER
 - voms_apic.h, [19](#)
- VERIFY_SIGN
 - voms_apic.h, [19](#)
- VERR_COMM
 - voms_apic.h, [19](#)
- VERR_DIR
 - voms_apic.h, [19](#)
- VERR_EXTRINFO
 - voms_apic.h, [19](#)
- VERR_FORMAT
 - voms_apic.h, [19](#)
- VERR_IDCHECK
 - voms_apic.h, [20](#)
- VERR_MEM
 - voms_apic.h, [20](#)
- VERR_NODATA
 - voms_apic.h, [20](#)
- VERR_NOEXT
 - voms_apic.h, [20](#)
- VERR_NOIDENT
 - voms_apic.h, [20](#)
- VERR_NOINIT
 - voms_apic.h, [20](#)
- VERR_NONE
 - voms_apic.h, [20](#)
- VERR_NOCKET
 - voms_apic.h, [20](#)
- VERR_NOTAVAIL
 - voms_apic.h, [21](#)
- VERR_ORDER
 - voms_apic.h, [21](#)
- VERR_PARAM
 - voms_apic.h, [21](#)
- VERR_PARSE
 - voms_apic.h, [21](#)
- VERR_SERVER
 - voms_apic.h, [21](#)
- VERR_SERVERCODE
 - voms_apic.h, [21](#)
- VERR_SIGN
 - voms_apic.h, [21](#)
- VERR_TIME
 - voms_apic.h, [21](#)
- VERR_TYPE
 - voms_apic.h, [22](#)
- VERR_VERIFY
 - voms_apic.h, [22](#)
- version
 - contactdata, [7](#)
 - voms, [11](#)

- vo
 - contactdata, [7](#)
- volen
 - vomsdata, [12](#)
- voms, [8](#)
 - ac, [9](#)
 - custom, [9](#)
 - datalen, [9](#)
 - date1, [9](#)
 - date2, [9](#)
 - fqn, [9](#)
 - holder, [9](#)
 - serial, [9](#)
 - server, [10](#)
 - serverca, [10](#)
 - siglen, [10](#)
 - signature, [10](#)
 - std, [10](#)
 - type, [10](#)
 - uri, [10](#)
 - user, [10](#)
 - userca, [11](#)
 - version, [11](#)
 - voname, [11](#)
- VOMS_AddTarget
 - voms_apic.h, [22](#)
- voms_apic.h, [15](#)
 - getMajorVersionNumber, [22](#)
 - getMinorVersionNumber, [22](#)
 - getPatchVersionNumber, [22](#)
 - gss_cred_id_t, [22](#)
 - gss_ctx_id_t, [22](#)
 - NOGLOBUS, [18](#)
 - RECURSE_CHAIN, [18](#)
 - RECURSE_NONE, [18](#)
 - TYPE_CUSTOM, [18](#)
 - TYPE_NODATA, [18](#)
 - TYPE_STD, [18](#)
 - VERIFY_CERTLIST, [18](#)
 - VERIFY_DATE, [18](#)
 - VERIFY_FULL, [18](#)
 - VERIFY_ID, [18](#)
 - VERIFY_KEY, [19](#)
 - VERIFY_NONE, [19](#)
 - VERIFY_NOTARGET, [19](#)
 - VERIFY_ORDER, [19](#)
 - VERIFY_SIGN, [19](#)
 - VERR_COMM, [19](#)
 - VERR_DIR, [19](#)
 - VERR_EXTRAINFO, [19](#)
 - VERR_FORMAT, [19](#)
 - VERR_IDCHECK, [20](#)
 - VERR_MEM, [20](#)
 - VERR_NODATA, [20](#)
 - VERR_NOEXT, [20](#)
 - VERR_NOIDENT, [20](#)
 - VERR_NOINIT, [20](#)
 - VERR_NONE, [20](#)
 - VERR_NOSOCKET, [20](#)
 - VERR_NOTAVAIL, [21](#)
 - VERR_ORDER, [21](#)
 - VERR_PARAM, [21](#)
 - VERR_PARSE, [21](#)
 - VERR_SERVER, [21](#)
 - VERR_SERVERCODE, [21](#)
 - VERR_SIGN, [21](#)
 - VERR_TIME, [21](#)
 - VERR_TYPE, [22](#)
 - VERR_VERIFY, [22](#)
 - VOMS_AddTarget, [22](#)
 - VOMS_Contact, [23](#)
 - VOMS_ContactRaw, [23](#)
 - VOMS_Copy, [23](#)
 - VOMS_CopyAll, [24](#)
 - VOMS_DefaultData, [24](#)
 - VOMS_Delete, [24](#)
 - VOMS_DeleteContacts, [24](#)
 - VOMS_Destroy, [25](#)
 - VOMS_Duplicate, [25](#)
 - VOMS_ErrorMessage, [25](#)
 - VOMS_Export, [25](#)
 - VOMS_FindByAlias, [25](#)
 - VOMS_FindByVO, [26](#)
 - VOMS_FreeTargets, [26](#)
 - VOMS_FreeTargetsList, [27](#)
 - VOMS_GetAC, [27](#)
 - VOMS_GetAttribute, [27](#)
 - VOMS_GetAttributeGrantor, [27](#)
 - VOMS_GetAttributesNumber, [27](#)
 - VOMS_GetAttributeSourceHandle, [27](#)
 - VOMS_GetAttributeSourcesNumber, [27](#)
 - VOMS_GetTargetsList, [27](#)
 - VOMS_Import, [27](#)
 - VOMS_Init, [27](#)
 - VOMS_ListTargets, [28](#)
 - VOMS_LoadCredentials, [28](#)
 - VOMS_Ordering, [28](#)
 - VOMS_ResetOrder, [28](#)
 - VOMS_Retrieve, [28](#)
 - VOMS_RetrieveEXT, [29](#)

- VOMS_RetrieveFromAC, [29](#)
- VOMS_RetrieveFromCred, [29](#)
- VOMS_RetrieveFromCtx, [30](#)
- VOMS_RetrieveFromFile, [30](#)
- VOMS_RetrieveFromProxy, [30](#)
- VOMS_SetLifetime, [31](#)
- VOMS_SetTimeout, [31](#)
- VOMS_SetVerificationTime, [31](#)
- VOMS_SetVerificationType, [31](#)
- VOMS_Contact
 - voms_apic.h, [23](#)
- VOMS_ContactRaw
 - voms_apic.h, [23](#)
- VOMS_Copy
 - voms_apic.h, [23](#)
- VOMS_CopyAll
 - voms_apic.h, [24](#)
- VOMS_DefaultData
 - voms_apic.h, [24](#)
- VOMS_Delete
 - voms_apic.h, [24](#)
- VOMS_DeleteContacts
 - voms_apic.h, [24](#)
- VOMS_Destroy
 - voms_apic.h, [25](#)
- VOMS_Duplicate
 - voms_apic.h, [25](#)
- VOMS_ErrorMessage
 - voms_apic.h, [25](#)
- VOMS_Export
 - voms_apic.h, [25](#)
- VOMS_FindByAlias
 - voms_apic.h, [25](#)
- VOMS_FindByVO
 - voms_apic.h, [26](#)
- VOMS_FreeTargets
 - voms_apic.h, [26](#)
- VOMS_FreeTargetsList
 - voms_apic.h, [27](#)
- VOMS_GetAC
 - voms_apic.h, [27](#)
- VOMS_GetAttribute
 - voms_apic.h, [27](#)
- VOMS_GetAttributeGrantor
 - voms_apic.h, [27](#)
- VOMS_GetAttributesNumber
 - voms_apic.h, [27](#)
- VOMS_GetAttributeSourceHandle
 - voms_apic.h, [27](#)
- VOMS_GetAttributeSourcesNumber
 - voms_apic.h, [27](#)
- voms_apic.h, [27](#)
- VOMS_GetTargetsList
 - voms_apic.h, [27](#)
- VOMS_Import
 - voms_apic.h, [27](#)
- VOMS_Init
 - voms_apic.h, [27](#)
- VOMS_ListTargets
 - voms_apic.h, [28](#)
- VOMS_LoadCredentials
 - voms_apic.h, [28](#)
- VOMS_Ordering
 - voms_apic.h, [28](#)
- VOMS_ResetOrder
 - voms_apic.h, [28](#)
- VOMS_Retrieve
 - voms_apic.h, [28](#)
- VOMS_RetrieveEXT
 - voms_apic.h, [29](#)
- VOMS_RetrieveFromAC
 - voms_apic.h, [29](#)
- VOMS_RetrieveFromCred
 - voms_apic.h, [29](#)
- VOMS_RetrieveFromCtx
 - voms_apic.h, [30](#)
- VOMS_RetrieveFromFile
 - voms_apic.h, [30](#)
- VOMS_RetrieveFromProxy
 - voms_apic.h, [30](#)
- VOMS_SetLifetime
 - voms_apic.h, [31](#)
- VOMS_SetTimeout
 - voms_apic.h, [31](#)
- VOMS_SetVerificationTime
 - voms_apic.h, [31](#)
- VOMS_SetVerificationType
 - voms_apic.h, [31](#)
- vomsdata, [11](#)
 - cdir, [12](#)
 - data, [12](#)
 - extra_data, [12](#)
 - extralen, [12](#)
 - real, [12](#)
 - vdir, [12](#)
 - volen, [12](#)
 - workvo, [12](#)
- voname
 - voms, [11](#)
- workvo

vomsdata, [12](#)