

# VOMS C API

1.5.0

Generated by Doxygen 1.8.4

Thu Aug 8 2013 12:41:05



# Contents

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Data Structure Index</b>            | <b>1</b> |
| 1.1      | Data Structures . . . . .              | 1        |
| <b>2</b> | <b>File Index</b>                      | <b>3</b> |
| 2.1      | File List . . . . .                    | 3        |
| <b>3</b> | <b>Data Structure Documentation</b>    | <b>5</b> |
| 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 . . . . . | 5        |
| 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 . . . . .                         | 6        |
| 3.2.2.5  | reserved . . . . .                     | 6        |
| 3.2.2.6  | version . . . . .                      | 6        |
| 3.2.2.7  | vo . . . . .                           | 7        |
| 3.3      | data Struct Reference . . . . .        | 7        |
| 3.3.1    | Detailed Description . . . . .         | 7        |
| 3.3.2    | Field Documentation . . . . .          | 7        |
| 3.3.2.1  | cap . . . . .                          | 7        |
| 3.3.2.2  | group . . . . .                        | 7        |
| 3.3.2.3  | role . . . . .                         | 7        |
| 3.4      | voms Struct Reference . . . . .        | 8        |
| 3.4.1    | Detailed Description . . . . .         | 8        |
| 3.4.2    | Field Documentation . . . . .          | 8        |

|          |                                |           |
|----------|--------------------------------|-----------|
| 3.4.2.1  | ac                             | 8         |
| 3.4.2.2  | custom                         | 8         |
| 3.4.2.3  | datalen                        | 8         |
| 3.4.2.4  | date1                          | 8         |
| 3.4.2.5  | date2                          | 9         |
| 3.4.2.6  | fqan                           | 9         |
| 3.4.2.7  | holder                         | 9         |
| 3.4.2.8  | serial                         | 9         |
| 3.4.2.9  | server                         | 9         |
| 3.4.2.10 | serverca                       | 9         |
| 3.4.2.11 | siglen                         | 9         |
| 3.4.2.12 | signature                      | 9         |
| 3.4.2.13 | std                            | 9         |
| 3.4.2.14 | type                           | 10        |
| 3.4.2.15 | uri                            | 10        |
| 3.4.2.16 | user                           | 10        |
| 3.4.2.17 | userca                         | 10        |
| 3.4.2.18 | version                        | 10        |
| 3.4.2.19 | voname                         | 10        |
| 3.5      | vomsdata Struct Reference      | 10        |
| 3.5.1    | Detailed Description           | 11        |
| 3.5.2    | Field Documentation            | 11        |
| 3.5.2.1  | cdir                           | 11        |
| 3.5.2.2  | data                           | 11        |
| 3.5.2.3  | extra_data                     | 11        |
| 3.5.2.4  | extralen                       | 11        |
| 3.5.2.5  | real                           | 11        |
| 3.5.2.6  | vdir                           | 11        |
| 3.5.2.7  | volen                          | 11        |
| 3.5.2.8  | workvo                         | 11        |
| <b>4</b> | <b>File Documentation</b>      | <b>13</b> |
| 4.1      | voms_apic.h File Reference     | 13        |
| 4.1.1    | Macro Definition Documentation | 15        |
| 4.1.1.1  | NOGLOBUS                       | 15        |
| 4.1.1.2  | RECURSE_CHAIN                  | 15        |
| 4.1.1.3  | RECURSE_NONE                   | 15        |
| 4.1.1.4  | TYPE_CUSTOM                    | 15        |
| 4.1.1.5  | TYPE_NODATA                    | 15        |
| 4.1.1.6  | TYPE_STD                       | 15        |

|          |                                  |    |
|----------|----------------------------------|----|
| 4.1.1.7  | VERIFY_CERTLIST . . . . .        | 16 |
| 4.1.1.8  | VERIFY_DATE . . . . .            | 16 |
| 4.1.1.9  | VERIFY_FULL . . . . .            | 16 |
| 4.1.1.10 | VERIFY_ID . . . . .              | 16 |
| 4.1.1.11 | VERIFY_KEY . . . . .             | 16 |
| 4.1.1.12 | VERIFY_NONE . . . . .            | 16 |
| 4.1.1.13 | VERIFY_NOTARGET . . . . .        | 16 |
| 4.1.1.14 | VERIFY_ORDER . . . . .           | 16 |
| 4.1.1.15 | VERIFY_SIGN . . . . .            | 16 |
| 4.1.1.16 | VERR_COMM . . . . .              | 16 |
| 4.1.1.17 | VERR_DIR . . . . .               | 16 |
| 4.1.1.18 | VERR_EXTRINFO . . . . .          | 17 |
| 4.1.1.19 | VERR_FORMAT . . . . .            | 17 |
| 4.1.1.20 | VERR_IDCHECK . . . . .           | 17 |
| 4.1.1.21 | VERR_MEM . . . . .               | 17 |
| 4.1.1.22 | VERR_NODATA . . . . .            | 17 |
| 4.1.1.23 | VERR_NOEXT . . . . .             | 17 |
| 4.1.1.24 | VERR_NOIDENT . . . . .           | 17 |
| 4.1.1.25 | VERR_NOINIT . . . . .            | 17 |
| 4.1.1.26 | VERR_NONE . . . . .              | 17 |
| 4.1.1.27 | VERR_NOSOCKET . . . . .          | 18 |
| 4.1.1.28 | VERR_NOTAVAIL . . . . .          | 18 |
| 4.1.1.29 | VERR_ORDER . . . . .             | 18 |
| 4.1.1.30 | VERR_PARAM . . . . .             | 18 |
| 4.1.1.31 | VERR_PARSE . . . . .             | 18 |
| 4.1.1.32 | VERR_SERVER . . . . .            | 18 |
| 4.1.1.33 | VERR_SERVERCODE . . . . .        | 18 |
| 4.1.1.34 | VERR_SIGN . . . . .              | 18 |
| 4.1.1.35 | VERR_TIME . . . . .              | 18 |
| 4.1.1.36 | VERR_TYPE . . . . .              | 19 |
| 4.1.1.37 | VERR_VERIFY . . . . .            | 19 |
| 4.1.2    | Typedef Documentation . . . . .  | 19 |
| 4.1.2.1  | gss_cred_id_t . . . . .          | 19 |
| 4.1.2.2  | gss_ctx_id_t . . . . .           | 19 |
| 4.1.3    | Function Documentation . . . . . | 19 |
| 4.1.3.1  | getMajorVersionNumber . . . . .  | 19 |
| 4.1.3.2  | getMinorVersionNumber . . . . .  | 19 |
| 4.1.3.3  | getPatchVersionNumber . . . . .  | 19 |
| 4.1.3.4  | VOMS_AddTarget . . . . .         | 19 |
| 4.1.3.5  | VOMS_Contact . . . . .           | 19 |

|          |  |    |
|----------|--|----|
| 4.1.3.6  | <a href="#">VOMS_ContactRaw</a>                | 20 |
| 4.1.3.7  | <a href="#">VOMS_Copy</a>                      | 20 |
| 4.1.3.8  | <a href="#">VOMS_CopyAll</a>                   | 20 |
| 4.1.3.9  | <a href="#">VOMS_DefaultData</a>               | 21 |
| 4.1.3.10 | <a href="#">VOMS_Delete</a>                    | 21 |
| 4.1.3.11 | <a href="#">VOMS_DeleteContacts</a>            | 21 |
| 4.1.3.12 | <a href="#">VOMS_Destroy</a>                   | 21 |
| 4.1.3.13 | <a href="#">VOMS_Duplicate</a>                 | 21 |
| 4.1.3.14 | <a href="#">VOMS_ErrorMessage</a>              | 21 |
| 4.1.3.15 | <a href="#">VOMS_Export</a>                    | 22 |
| 4.1.3.16 | <a href="#">VOMS_FindByAlias</a>               | 22 |
| 4.1.3.17 | <a href="#">VOMS_FindByVO</a>                  | 22 |
| 4.1.3.18 | <a href="#">VOMS_FreeTargets</a>               | 23 |
| 4.1.3.19 | <a href="#">VOMS_FreeTargetsList</a>           | 23 |
| 4.1.3.20 | <a href="#">VOMS_GetAC</a>                     | 23 |
| 4.1.3.21 | <a href="#">VOMS_GetAttribute</a>              | 23 |
| 4.1.3.22 | <a href="#">VOMS_GetAttributeGrantor</a>       | 23 |
| 4.1.3.23 | <a href="#">VOMS_GetAttributesNumber</a>       | 23 |
| 4.1.3.24 | <a href="#">VOMS_GetAttributeSourceHandle</a>  | 23 |
| 4.1.3.25 | <a href="#">VOMS_GetAttributeSourcesNumber</a> | 23 |
| 4.1.3.26 | <a href="#">VOMS_GetTargetsList</a>            | 23 |
| 4.1.3.27 | <a href="#">VOMS_Import</a>                    | 23 |
| 4.1.3.28 | <a href="#">VOMS_Init</a>                      | 24 |
| 4.1.3.29 | <a href="#">VOMS_ListTargets</a>               | 24 |
| 4.1.3.30 | <a href="#">VOMS_LoadCredentials</a>           | 24 |
| 4.1.3.31 | <a href="#">VOMS_Ordering</a>                  | 24 |
| 4.1.3.32 | <a href="#">VOMS_ResetOrder</a>                | 24 |
| 4.1.3.33 | <a href="#">VOMS_Retrieve</a>                  | 25 |
| 4.1.3.34 | <a href="#">VOMS_RetrieveEXT</a>               | 25 |
| 4.1.3.35 | <a href="#">VOMS_RetrieveFromAC</a>            | 25 |
| 4.1.3.36 | <a href="#">VOMS_RetrieveFromCred</a>          | 25 |
| 4.1.3.37 | <a href="#">VOMS_RetrieveFromCtx</a>           | 26 |
| 4.1.3.38 | <a href="#">VOMS_RetrieveFromFile</a>          | 26 |
| 4.1.3.39 | <a href="#">VOMS_RetrieveFromProxy</a>         | 26 |
| 4.1.3.40 | <a href="#">VOMS_SetLifetime</a>               | 27 |
| 4.1.3.41 | <a href="#">VOMS_SetTimeout</a>                | 28 |
| 4.1.3.42 | <a href="#">VOMS_SetVerificationTime</a>       | 28 |
| 4.1.3.43 | <a href="#">VOMS_SetVerificationType</a>       | 28 |







# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

|                             |   |    |
|-----------------------------|---|----|
| <a href="#">attribute</a>   | .....   | 5  |
| <a href="#">contactdata</a> |   |    |
|                             | The type of data returned .....               | 5  |
| <a href="#">data</a>        |   |    |
|                             | User's characteristics: can be repeated ..... | 7  |
| <a href="#">voms</a>        | .....   | 8  |
| <a href="#">vomldata</a>    | .....   | 10 |



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

|                             |    |
|-----------------------------|----|
| <a href="#">voms_apic.h</a> | 13 |
|-----------------------------|----|



## 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 62 of file voms\_apic.h.

#### 3.1.2 Field Documentation

##### 3.1.2.1 const char\* attribute::name

Definition at line 63 of file voms\_apic.h.

##### 3.1.2.2 const char\* attribute::qualifier

Definition at line 65 of file voms\_apic.h.

##### 3.1.2.3 const char\* attribute::value

Definition at line 64 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 68 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 73 of file voms\_apic.h.

#### 3.2.2.2 char\* contactdata::host

The hostname of the server

Definition at line 72 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 71 of file voms\_apic.h.

#### 3.2.2.4 int contactdata::port

The port on which the server is listening

Definition at line 75 of file voms\_apic.h.

#### 3.2.2.5 char\* contactdata::reserved

HANDS OFF!

Definition at line 76 of file voms\_apic.h.

#### 3.2.2.6 int contactdata::version

The version of Globus on which this server runs.

Definition at line 77 of file voms\_apic.h.

### 3.2.2.7 char\* contactdata::vo

The VO served by this server

Definition at line 74 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 56 of file voms\_apic.h.

### 3.3.2 Field Documentation

#### 3.3.2.1 char\* data::cap

user's capability

Definition at line 59 of file voms\_apic.h.

#### 3.3.2.2 char\* data::group

user's group

Definition at line 57 of file voms\_apic.h.

#### 3.3.2.3 char\* data::role

user's role

Definition at line 58 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 88 of file `voms_apic.h`.

### 3.4.2 Field Documentation

#### 3.4.2.1 AC\* voms::ac

Definition at line 108 of file `voms_apic.h`.

#### 3.4.2.2 char\* voms::custom

The data returned by an S command

Definition at line 101 of file `voms_apic.h`.

#### 3.4.2.3 int voms::datalen

Definition at line 102 of file `voms_apic.h`.

#### 3.4.2.4 char\* voms::date1

Beginning of validity of the user info

Definition at line 97 of file `voms_apic.h`.



**3.4.2.5 char\* voms::date2**

End of validity of the user info

Definition at line 98 of file voms\_apic.h.

**3.4.2.6 char\*\* voms::fqan**

User's attributes in compact format

Definition at line 104 of file voms\_apic.h.

**3.4.2.7 X509\* voms::holder**

Definition at line 109 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 105 of file voms\_apic.h.

**3.4.2.9 char\* voms::server**

The VOMS server DN, as from its certificate

Definition at line 93 of file voms\_apic.h.

**3.4.2.10 char\* voms::serverca**

The CA which signed the VOMS certificate

Definition at line 94 of file voms\_apic.h.

**3.4.2.11 int voms::siglen**

The length of the VOMS server signature

Definition at line 89 of file voms\_apic.h.

**3.4.2.12 char\* voms::signature**

The VOMS server signature

Definition at line 90 of file voms\_apic.h.

**3.4.2.13 struct data\*\* voms::std**

User's characteristics

Definition at line 100 of file voms\_apic.h.

#### 3.4.2.14 int voms::type

The type of data returned

Definition at line 99 of file voms\_apic.h.

#### 3.4.2.15 char\* voms::uri

The URI of the VOMS server

Definition at line 96 of file voms\_apic.h.

#### 3.4.2.16 char\* voms::user

The user's DN, as from his certificate

Definition at line 91 of file voms\_apic.h.

#### 3.4.2.17 char\* voms::userca

The CA which signed the user's certificate

Definition at line 92 of file voms\_apic.h.

#### 3.4.2.18 int voms::version

Definition at line 103 of file voms\_apic.h.

#### 3.4.2.19 char\* voms::vname

The name of the VO to which the VOMS belongs

Definition at line 95 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 153 of file voms\_apic.h.

### 3.5.2 Field Documentation

#### 3.5.2.1 `char* vomsdata::cdir`

Definition at line 154 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 156 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 161 of file voms\_apic.h.

#### 3.5.2.4 `int vomsdata::extralen`

Definition at line 171 of file voms\_apic.h.

#### 3.5.2.5 `struct vomsdata* vomsdata::real`

Definition at line 173 of file voms\_apic.h.

#### 3.5.2.6 `char* vomsdata::vdir`

Definition at line 155 of file voms\_apic.h.

#### 3.5.2.7 `int vomsdata::volen`

Definition at line 170 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 159 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](#)

#### Macros

- #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_EXTRINFO 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 Macro Definition 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 113 of file voms\_apic.h.

### 4.1.1.3 #define RECURSE\_NONE 1

Definition at line 114 of file voms\_apic.h.

### 4.1.1.4 #define TYPE\_CUSTOM 2

result of an S command

Definition at line 85 of file voms\_apic.h.

### 4.1.1.5 #define TYPE\_NODATA 0

no data

Definition at line 83 of file voms\_apic.h.

### 4.1.1.6 #define TYPE\_STD 1

group, role, capability triplet

Definition at line 84 of file voms\_apic.h.

4.1.1.7 `#define VERIFY_CERTLIST 0x00000040`

Definition at line 124 of file voms\_apic.h.

4.1.1.8 `#define VERIFY_DATE 0x00000001`

Definition at line 118 of file voms\_apic.h.

4.1.1.9 `#define VERIFY_FULL 0xffffffff`

Definition at line 116 of file voms\_apic.h.

4.1.1.10 `#define VERIFY_ID 0x00000020`

Definition at line 123 of file voms\_apic.h.

4.1.1.11 `#define VERIFY_KEY 0x00000004`

Definition at line 120 of file voms\_apic.h.

4.1.1.12 `#define VERIFY_NONE 0x00000000`

Definition at line 117 of file voms\_apic.h.

4.1.1.13 `#define VERIFY_NOTARGET 0x00000002`

Definition at line 119 of file voms\_apic.h.

4.1.1.14 `#define VERIFY_ORDER 0x00000010`

Definition at line 122 of file voms\_apic.h.

4.1.1.15 `#define VERIFY_SIGN 0x00000008`

Definition at line 121 of file voms\_apic.h.

4.1.1.16 `#define VERR_COMM 3`

Server problem

Definition at line 131 of file voms\_apic.h.

4.1.1.17 `#define VERR_DIR 13`

Directory error

Definition at line 143 of file voms\_apic.h.



**4.1.1.18 #define VERR\_EXTRINFO 9**

VO name and URI missing

Definition at line 139 of file voms\_apic.h.

**4.1.1.19 #define VERR\_FORMAT 10**

Wrong data format

Definition at line 140 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 136 of file voms\_apic.h.

**4.1.1.21 #define VERR\_MEM 16**

Memory problems

Definition at line 146 of file voms\_apic.h.

**4.1.1.22 #define VERR\_NODATA 11**

Empty extension

Definition at line 141 of file voms\_apic.h.

**4.1.1.23 #define VERR\_NOEXT 5**

VOMS extension missing

Definition at line 133 of file voms\_apic.h.

**4.1.1.24 #define VERR\_NOIDENT 2**

Cannot identify itself (certificate problem)

Definition at line 130 of file voms\_apic.h.

**4.1.1.25 #define VERR\_NOINIT 6**

Initialization error

Definition at line 134 of file voms\_apic.h.

**4.1.1.26 #define VERR\_NONE 0**

Error codes.

Definition at line 128 of file voms\_apic.h.

**4.1.1.27 #define VERR\_NOSOCKET 1**

Socket problem

Definition at line 129 of file voms\_apic.h.

**4.1.1.28 #define VERR\_NOTAVAIL 21**

Method not available

Definition at line 151 of file voms\_apic.h.

**4.1.1.29 #define VERR\_ORDER 19**

Ordering different than required

Definition at line 149 of file voms\_apic.h.

**4.1.1.30 #define VERR\_PARAM 4**

Wrong parameters

Definition at line 132 of file voms\_apic.h.

**4.1.1.31 #define VERR\_PARSE 12**

Parse error

Definition at line 142 of file voms\_apic.h.

**4.1.1.32 #define VERR\_SERVER 15**

Unidentifiable VOMS server

Definition at line 145 of file voms\_apic.h.

**4.1.1.33 #define VERR\_SERVERCODE 20**

Error from the server

Definition at line 150 of file voms\_apic.h.

**4.1.1.34 #define VERR\_SIGN 14**

Signature error

Definition at line 144 of file voms\_apic.h.

**4.1.1.35 #define VERR\_TIME 7**

Error in time checking

Definition at line 135 of file voms\_apic.h.

#### 4.1.1.36 #define VERR\_TYPE 18

Returned data of unknown type

Definition at line 148 of file voms\_apic.h.

#### 4.1.1.37 #define VERR\_VERIFY 17

Generic verification error

Definition at line 147 of file voms\_apic.h.

### 4.1.2 Typedef Documentation

#### 4.1.2.1 typedef void\* gss\_cred\_id\_t

Definition at line 42 of file voms\_apic.h.

#### 4.1.2.2 typedef void\* gss\_ctx\_id\_t

Definition at line 43 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\_Contact, 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 )`

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 )`

Copies a vomsdata structure.

N.B: This is the ONLY way to correctly initialize a vomsdata structure

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* )**

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* )****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* )

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* )

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 )**

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<br><br>If voms_dir is empty, the value of the environment variable X509_VOMS_DIR is taken<br><br>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, [8](#)
- attribute, [5](#)
  - name, [5](#)
  - qualifier, [5](#)
  - value, [5](#)
- cap
  - data, [7](#)
- cdir
  - vomsdata, [11](#)
- contact
  - contactdata, [6](#)
- contactdata, [5](#)
  - contact, [6](#)
  - host, [6](#)
  - nick, [6](#)
  - port, [6](#)
  - reserved, [6](#)
  - version, [6](#)
  - vo, [6](#)
- custom
  - voms, [8](#)
- data, [7](#)
  - cap, [7](#)
  - group, [7](#)
  - role, [7](#)
  - vomsdata, [11](#)
- datalen
  - voms, [8](#)
- date1
  - voms, [8](#)
- date2
  - voms, [8](#)
- extra\_data
  - vomsdata, [11](#)
- extralen
  - vomsdata, [11](#)
- fqn
  - voms, [9](#)
- getMajorVersionNumber
  - voms\_apic.h, [19](#)
- getMinorVersionNumber
  - voms\_apic.h, [19](#)
- getPatchVersionNumber
  - voms\_apic.h, [19](#)
- group
  - data, [7](#)
- gss\_cred\_id\_t
  - voms\_apic.h, [19](#)
- gss\_ctx\_id\_t
  - voms\_apic.h, [19](#)
- holder
  - voms, [9](#)
- host
  - contactdata, [6](#)
- NOGLOBUS
  - voms\_apic.h, [15](#)
- name
  - attribute, [5](#)
- nick
  - contactdata, [6](#)
- port
  - contactdata, [6](#)
- qualifier
  - attribute, [5](#)
- RECURSE\_CHAIN
  - voms\_apic.h, [15](#)
- RECURSE\_NONE
  - voms\_apic.h, [15](#)
- real
  - vomsdata, [11](#)
- reserved
  - contactdata, [6](#)
- role
  - data, [7](#)
- serial
  - voms, [9](#)
- server
  - voms, [9](#)
- serverca
  - voms, [9](#)
- siglen
  - voms, [9](#)
- signature
  - voms, [9](#)
- std
  - voms, [9](#)
- TYPE\_CUSTOM
  - voms\_apic.h, [15](#)
- TYPE\_NODATA

- voms\_apic.h, [15](#)
- TYPE\_STD
  - voms\_apic.h, [15](#)
- type
  - voms, [9](#)
- uri
  - voms, [10](#)
- user
  - voms, [10](#)
- userca
  - voms, [10](#)
- VERIFY\_CERTLIST
  - voms\_apic.h, [15](#)
- VERIFY\_DATE
  - voms\_apic.h, [16](#)
- VERIFY\_FULL
  - voms\_apic.h, [16](#)
- VERIFY\_ID
  - voms\_apic.h, [16](#)
- VERIFY\_KEY
  - voms\_apic.h, [16](#)
- VERIFY\_NONE
  - voms\_apic.h, [16](#)
- VERIFY\_NOTARGET
  - voms\_apic.h, [16](#)
- VERIFY\_ORDER
  - voms\_apic.h, [16](#)
- VERIFY\_SIGN
  - voms\_apic.h, [16](#)
- VERR\_COMM
  - voms\_apic.h, [16](#)
- VERR\_DIR
  - voms\_apic.h, [16](#)
- VERR\_EXTRINFO
  - voms\_apic.h, [16](#)
- VERR\_FORMAT
  - voms\_apic.h, [17](#)
- VERR\_IDCHECK
  - voms\_apic.h, [17](#)
- VERR\_MEM
  - voms\_apic.h, [17](#)
- VERR\_NODATA
  - voms\_apic.h, [17](#)
- VERR\_NOEXT
  - voms\_apic.h, [17](#)
- VERR\_NOIDENT
  - voms\_apic.h, [17](#)
- VERR\_NOINIT
  - voms\_apic.h, [17](#)
- VERR\_NONE
  - voms\_apic.h, [17](#)
- VERR\_NOSOCKET
  - voms\_apic.h, [17](#)
- VERR\_NOTAVAIL
  - voms\_apic.h, [18](#)
- VERR\_ORDER
  - voms\_apic.h, [18](#)
- VERR\_PARAM
  - voms\_apic.h, [18](#)
- VERR\_PARSE
  - voms\_apic.h, [18](#)
- VERR\_SERVER
  - voms\_apic.h, [18](#)
- VERR\_SERVERCODE
  - voms\_apic.h, [18](#)
- VERR\_SIGN
  - voms\_apic.h, [18](#)
- VERR\_TIME
  - voms\_apic.h, [18](#)
- VERR\_TYPE
  - voms\_apic.h, [18](#)
- VERR\_VERIFY
  - voms\_apic.h, [19](#)
- VOMS\_AddTarget
  - voms\_apic.h, [19](#)
- VOMS\_Contact
  - voms\_apic.h, [19](#)
- VOMS\_ContactRaw
  - voms\_apic.h, [20](#)
- VOMS\_Copy
  - voms\_apic.h, [20](#)
- VOMS\_CopyAll
  - voms\_apic.h, [20](#)
- VOMS\_DefaultData
  - voms\_apic.h, [21](#)
- VOMS\_Delete
  - voms\_apic.h, [21](#)
- VOMS\_DeleteContacts
  - voms\_apic.h, [21](#)
- VOMS\_Destroy
  - voms\_apic.h, [21](#)
- VOMS\_Duplicate
  - voms\_apic.h, [21](#)
- VOMS\_ErrorMessage
  - voms\_apic.h, [21](#)
- VOMS\_Export
  - voms\_apic.h, [22](#)
- VOMS\_FindByAlias
  - voms\_apic.h, [22](#)
- VOMS\_FindByVO
  - voms\_apic.h, [22](#)
- VOMS\_FreeTargets
  - voms\_apic.h, [23](#)
- VOMS\_FreeTargetsList
  - voms\_apic.h, [23](#)
- VOMS\_GetAC
  - voms\_apic.h, [23](#)
- VOMS\_GetAttribute
  - voms\_apic.h, [23](#)
- VOMS\_GetAttributeGrantor
  - voms\_apic.h, [23](#)
- VOMS\_GetAttributeSourceHandle
  - voms\_apic.h, [23](#)
- VOMS\_GetAttributeSourcesNumber
  - voms\_apic.h, [23](#)

- VOMS\_GetAttributesNumber
  - voms\_apic.h, [23](#)
- VOMS\_GetTargetsList
  - voms\_apic.h, [23](#)
- VOMS\_Import
  - voms\_apic.h, [23](#)
- VOMS\_Init
  - voms\_apic.h, [24](#)
- VOMS\_ListTargets
  - voms\_apic.h, [24](#)
- VOMS\_LoadCredentials
  - voms\_apic.h, [24](#)
- VOMS\_Ordering
  - voms\_apic.h, [24](#)
- VOMS\_ResetOrder
  - voms\_apic.h, [24](#)
- VOMS\_Retrieve
  - voms\_apic.h, [25](#)
- VOMS\_RetrieveEXT
  - voms\_apic.h, [25](#)
- VOMS\_RetrieveFromAC
  - voms\_apic.h, [25](#)
- VOMS\_RetrieveFromCred
  - voms\_apic.h, [25](#)
- VOMS\_RetrieveFromCtx
  - voms\_apic.h, [26](#)
- VOMS\_RetrieveFromFile
  - voms\_apic.h, [26](#)
- VOMS\_RetrieveFromProxy
  - voms\_apic.h, [26](#)
- VOMS\_SetLifetime
  - voms\_apic.h, [26](#)
- VOMS\_SetTimeout
  - voms\_apic.h, [28](#)
- VOMS\_SetVerificationTime
  - voms\_apic.h, [28](#)
- VOMS\_SetVerificationType
  - voms\_apic.h, [28](#)
- value
  - attribute, [5](#)
- vdir
  - vomsdata, [11](#)
- version
  - contactdata, [6](#)
  - voms, [10](#)
- vo
  - contactdata, [6](#)
- volen
  - vomsdata, [11](#)
- voms, [8](#)
  - ac, [8](#)
  - custom, [8](#)
  - datalen, [8](#)
  - date1, [8](#)
  - date2, [8](#)
  - fqan, [9](#)
  - holder, [9](#)
  - serial, [9](#)
  - server, [9](#)
  - serverca, [9](#)
  - siglen, [9](#)
  - signature, [9](#)
  - std, [9](#)
  - type, [9](#)
  - uri, [10](#)
  - user, [10](#)
  - userca, [10](#)
  - version, [10](#)
  - voname, [10](#)
- voms\_apic.h, [13](#)
  - getMajorVersionNumber, [19](#)
  - getMinorVersionNumber, [19](#)
  - getPatchVersionNumber, [19](#)
  - gss\_cred\_id\_t, [19](#)
  - gss\_ctx\_id\_t, [19](#)
  - NOGLOBUS, [15](#)
  - RECURSE\_CHAIN, [15](#)
  - RECURSE\_NONE, [15](#)
  - TYPE\_CUSTOM, [15](#)
  - TYPE\_NODATA, [15](#)
  - TYPE\_STD, [15](#)
  - VERIFY\_CERTLIST, [15](#)
  - VERIFY\_DATE, [16](#)
  - VERIFY\_FULL, [16](#)
  - VERIFY\_ID, [16](#)
  - VERIFY\_KEY, [16](#)
  - VERIFY\_NONE, [16](#)
  - VERIFY\_NOTARGET, [16](#)
  - VERIFY\_ORDER, [16](#)
  - VERIFY\_SIGN, [16](#)
  - VERR\_COMM, [16](#)
  - VERR\_DIR, [16](#)
  - VERR\_EXTRINFO, [16](#)
  - VERR\_FORMAT, [17](#)
  - VERR\_IDCHECK, [17](#)
  - VERR\_MEM, [17](#)
  - VERR\_NODATA, [17](#)
  - VERR\_NOEXT, [17](#)
  - VERR\_NOIDENT, [17](#)
  - VERR\_NOINIT, [17](#)
  - VERR\_NONE, [17](#)
  - VERR\_NOSOCKET, [17](#)
  - VERR\_NOTAVAIL, [18](#)
  - VERR\_ORDER, [18](#)
  - VERR\_PARAM, [18](#)
  - VERR\_PARSE, [18](#)
  - VERR\_SERVER, [18](#)
  - VERR\_SERVERCODE, [18](#)
  - VERR\_SIGN, [18](#)
  - VERR\_TIME, [18](#)
  - VERR\_TYPE, [18](#)
  - VERR\_VERIFY, [19](#)
  - VOMS\_AddTarget, [19](#)
  - VOMS\_Contact, [19](#)
  - VOMS\_ContactRaw, [20](#)
  - VOMS\_Copy, [20](#)

- VOMS\_CopyAll, [20](#)
- VOMS\_DefaultData, [21](#)
- VOMS\_Delete, [21](#)
- VOMS\_DeleteContacts, [21](#)
- VOMS\_Destroy, [21](#)
- VOMS\_Duplicate, [21](#)
- VOMS\_ErrorMessage, [21](#)
- VOMS\_Export, [22](#)
- VOMS\_FindByAlias, [22](#)
- VOMS\_FindByVO, [22](#)
- VOMS\_FreeTargets, [23](#)
- VOMS\_FreeTargetsList, [23](#)
- VOMS\_GetAC, [23](#)
- VOMS\_GetAttribute, [23](#)
- VOMS\_GetAttributeGrantor, [23](#)
- VOMS\_GetAttributeSourceHandle, [23](#)
- VOMS\_GetAttributeSourcesNumber, [23](#)
- VOMS\_GetAttributesNumber, [23](#)
- VOMS\_GetTargetsList, [23](#)
- VOMS\_Import, [23](#)
- VOMS\_Init, [24](#)
- VOMS\_ListTargets, [24](#)
- VOMS\_LoadCredentials, [24](#)
- VOMS\_Ordering, [24](#)
- VOMS\_ResetOrder, [24](#)
- VOMS\_Retrieve, [25](#)
- VOMS\_RetrieveEXT, [25](#)
- VOMS\_RetrieveFromAC, [25](#)
- VOMS\_RetrieveFromCred, [25](#)
- VOMS\_RetrieveFromCtx, [26](#)
- VOMS\_RetrieveFromFile, [26](#)
- VOMS\_RetrieveFromProxy, [26](#)
- VOMS\_SetLifetime, [26](#)
- VOMS\_SetTimeout, [28](#)
- VOMS\_SetVerificationTime, [28](#)
- VOMS\_SetVerificationType, [28](#)
- vomsdata, [10](#)
  - cdir, [11](#)
  - data, [11](#)
  - extra\_data, [11](#)
  - extralen, [11](#)
  - real, [11](#)
  - vdir, [11](#)
  - volen, [11](#)
  - workvo, [11](#)
- voname
  - voms, [10](#)
- workvo
  - vomsdata, [11](#)