

VOMS C API Reference Manual

1.5.0

Generated by Doxygen 1.2.18

Tue Jun 30 05:34:26 2009

Contents

1	VOMS C API Data Structure Index	1
1.1	VOMS C API Data Structures	1
2	VOMS C API File Index	3
2.1	VOMS C API File List	3
3	VOMS C API Data Structure Documentation	5
3.1	attribute Struct Reference.	5
3.2	contactdata Struct Reference.	6
3.3	data Struct Reference	8
3.4	voms Struct Reference	9
3.5	vomsdata Struct Reference.	12
4	VOMS C API File Documentation	15
4.1	vomsapic.h File Reference	15

Chapter 1

VOMS C API Data Structure Index

1.1 VOMS C API Data Structures

Here are the data structures with brief descriptions:

attribute	5
contactdata	6
data (User's characteristics: can be repeated)	8
voms	9
vomsdata	12

Chapter 2

VOMS C API File Index

2.1 VOMS C API File List

Here is a list of all files with brief descriptions:

[voms.apic.h](#) 15

Chapter 3

VOMS C API 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 Field Documentation

3.1.1.1 const char attribute::name

Definition at line 39 of file voms_apic.h.

3.1.1.2 const char attribute::qualifier

Definition at line 41 of file voms_apic.h.

3.1.1.3 const char attribute::value

Definition at line 40 of file voms_apic.h.

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

[voms_apic.h](#)

3.2 contactdata Struct Reference

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

Data Fields

```
char nick  
char host  
char contact  
char vo  
int port  
char reserved  
int version
```

3.2.1 Field Documentation

3.2.1.1 char contactdata::contact

The subject of the server's certificate
Definition at line 49 of file voms_apic.h.

3.2.1.2 char contactdata::host

The hostname of the server
Definition at line 48 of file voms_apic.h.

3.2.1.3 char contactdata::nick

The alias of the server
Definition at line 47 of file voms_apic.h.

3.2.1.4 int contactdata::port

The port on which the server is listening
Definition at line 51 of file voms_apic.h.

3.2.1.5 char contactdata::reserved

HANDS OFF!
Definition at line 52 of file voms_apic.h.

3.2.1.6 int contactdata::version

The version of Globus on which this server runs.
Definition at line 53 of file voms_apic.h.

3.2.1.7 char contactdata::vo

The VO served by this server

Definition at line 50 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 32 of file voms_apic.h.

3.3.2 Field Documentation

3.3.2.1 char data::cap

user's capability

Definition at line 35 of file voms_apic.h.

3.3.2.2 char data::group

user's group

Definition at line 33 of file voms_apic.h.

3.3.2.3 char data::role

user's role

Definition at line 34 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  
data std  
char custom  
int datalen  
int version  
char fqan  
char serial  
AC ac  
X509 holder
```

3.4.1 Field Documentation

3.4.1.1 AC voms::ac

Definition at line 84 of file voms_apic.h.

3.4.1.2 char voms::custom

The data returned by an S command

Definition at line 77 of file voms_apic.h.

3.4.1.3 int voms::datalen

Definition at line 78 of file voms_apic.h.

3.4.1.4 char voms::date1

Beginning of validity of the user info

Definition at line 73 of file voms_apic.h.

3.4.1.5 char voms::date2

End of validity of the user info

Definition at line 74 of file voms.apic.h.

3.4.1.6 char voms::fqan

User's attributes in compact format

Definition at line 80 of file voms.apic.h.

3.4.1.7 X509 voms::holder

Definition at line 85 of file voms.apic.h.

3.4.1.8 char voms::serial

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

Definition at line 81 of file voms.apic.h.

3.4.1.9 char voms::server

The VOMS server DN, as from its certificate

Definition at line 69 of file voms.apic.h.

3.4.1.10 char voms::serverca

The CA which signed the VOMS certificate

Definition at line 70 of file voms.apic.h.

3.4.1.11 int voms::siglen

The length of the VOMS server signature

Definition at line 65 of file voms.apic.h.

3.4.1.12 char voms::signature

The VOMS server signature

Definition at line 66 of file voms.apic.h.

3.4.1.13 struct [data](#) voms::std

User's characteristics

Definition at line 76 of file voms.apic.h.

3.4.1.14 int voms::type

The type of data returned

Definition at line 75 of file voms.apic.h.

3.4.1.15 char voms::uri

The URI of the VOMS server

Definition at line 72 of file voms.apic.h.

3.4.1.16 char voms::user

The user's DN, as from his certificate

Definition at line 67 of file voms.apic.h.

3.4.1.17 char voms::userca

The CA which signed the user's certificate

Definition at line 68 of file voms.apic.h.

3.4.1.18 int voms::version

Definition at line 79 of file voms.apic.h.

3.4.1.19 char voms::voname

The name of the VO to which the VOMS belongs

Definition at line 71 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  
voms data  
char workvo  
char extra.data  
int volen  
int extralen  
vomsdata real
```

3.5.1 Field Documentation

3.5.1.1 char vomsdata::cdir

Definition at line 129 of file voms_apic.h.

3.5.1.2 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 131 of file voms_apic.h.

3.5.1.3 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 136 of file voms_apic.h.

3.5.1.4 int vomsdata::extralen

Definition at line 146 of file voms_apic.h.

3.5.1.5 struct vomsdata vomsdata::real

Definition at line 148 of file voms_apic.h.

3.5.1.6 char vomsdata::vdir

Definition at line 130 of file voms_apic.h.

3.5.1.7 int vomsdata::volen

Definition at line 145 of file voms.apic.h.

3.5.1.8 char vomsdata::workvo

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

Definition at line 134 of file voms.apic.h.

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

[voms.apic.h](#)

Chapter 4

VOMS C API File Documentation

4.1 vomsapic.h File Reference

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

Data Structures

```
struct attribute
struct contactdata
struct data
    User's characteristics: can be repeated.

struct voms
struct vomsdata
```

Defines

```
#define TYPE\_NODATA 0
#define TYPE\_STD 1
#define TYPE\_CUSTOM 2
#define RECURSECHAIN 0
#define RECURSENONE 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

```
contactdata VOMS_FindByAlias(struct vomsdata vd, char alias, char system, char user, int
error)
contactdata VOMS_FindByVO (struct vomsdata vd, char vo, char system, char user, int
error)
void VOMS_DeleteContact(struct contactdata list)
vomsdata VOMS_Init (char voms, char cert)
voms VOMS_Copy(struct voms v, int error)
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)
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_RetrieveFromCredential(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_RetrieveFromContext(gss_ctx_id_t ctx, int how, struct vomsdata vd, int error)
int VOMS_RetrieveFromProxy(int how, struct vomsdata vd, int error)
int VOMS_GetAttributeSourcesNumbers(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_GetAttributesNumbers(struct voms v, int handle, struct vomsdata vd, int error)
int VOMS_GetAttribute(struct voms v, int handle, int num, struct attribute attr, struct vomsdata
vd, int error)
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 *)

```

4.1.1 De ne Documentation

4.1.1.1 #de ne RECURSECHAIN 0

De nition at line 89 of le voms.apic.h.

4.1.1.2 #de ne RECURSENONE 1

De nition at line 90 of le voms.apic.h.

4.1.1.3 #de ne TYPECUSTOM 2

result of an S command

De nition at line 61 of le voms.apic.h.

4.1.1.4 #de ne TYPENODATA 0

no data

De nition at line 59 of le voms.apic.h.

4.1.1.5 #define TYPESTD 1

group, role, capability triplet

Definition at line 60 of `le_voms.apic.h`.

4.1.1.6 #define VERIFY_CERTLIST 0x00000040

Definition at line 100 of `le_voms.apic.h`.

4.1.1.7 #define VERIFY_DATE 0x00000001

Definition at line 94 of `le_voms.apic.h`.

4.1.1.8 #define VERIFY_FULL 0xffffffff

Definition at line 92 of `le_voms.apic.h`.

4.1.1.9 #define VERIFY_ID 0x00000020

Definition at line 99 of `le_voms.apic.h`.

4.1.1.10 #define VERIFY_KEY 0x00000004

Definition at line 96 of `le_voms.apic.h`.

4.1.1.11 #define VERIFY_NONE 0x00000000

Definition at line 93 of `le_voms.apic.h`.

4.1.1.12 #define VERIFY_NOTARGET 0x00000002

Definition at line 95 of `le_voms.apic.h`.

4.1.1.13 #define VERIFY_ORDER 0x00000010

Definition at line 98 of `le_voms.apic.h`.

4.1.1.14 #define VERIFY_SIGN 0x00000008

Definition at line 97 of `le_voms.apic.h`.

4.1.1.15 #define VERRCOMM 3

Server problem

Definition at line 107 of `le_voms.apic.h`.

4.1.1.16 #define VERRDIR 13

Directory error

Definition at line 118 of file voms.apic.h.

4.1.1.17 #define VERREXTRAINFO 9

VO name and URI missing

Definition at line 114 of file voms.apic.h.

4.1.1.18 #define VERRFORMAT 10

Wrong data format

Definition at line 115 of file voms.apic.h.

4.1.1.19 #define VERRIDCHECK 8

User data in extension different from the real ones

Definition at line 112 of file voms.apic.h.

4.1.1.20 #define VERRMEM 16

Memory problems

Definition at line 121 of file voms.apic.h.

4.1.1.21 #define VERRNODATA 11

Empty extension

Definition at line 116 of file voms.apic.h.

4.1.1.22 #define VERRNOEXT 5

VOMS extension missing

Definition at line 109 of file voms.apic.h.

4.1.1.23 #define VERRNOIDENT 2

Cannot identify itself (certificate problem)

Definition at line 106 of file voms.apic.h.

4.1.1.24 #define VERRNOINIT 6

Initialization error

Definition at line 110 of file voms.apic.h.

4.1.1.25 #define VERRNONE 0

Error codes.

Definition at line 104 of file voms.apic.h.

4.1.1.26 #define VERRNOSOCKET 1

Socket problem

Definition at line 105 of file voms.apic.h.

4.1.1.27 #define VERRNOTAVAIL 21

Method not available

Definition at line 126 of file voms.apic.h.

4.1.1.28 #define VERRORDER 19

Ordering different than required

Definition at line 124 of file voms.apic.h.

4.1.1.29 #define VERRPARAM 4

Wrong parameters

Definition at line 108 of file voms.apic.h.

4.1.1.30 #define VERRPARSE 12

Parse error

Definition at line 117 of file voms.apic.h.

4.1.1.31 #define VERRSERVER 15

Unidentifiable VOMS server

Definition at line 120 of file voms.apic.h.

4.1.1.32 #define VERRSERVERCODE 20

Error from the server

Definition at line 125 of file voms.apic.h.

4.1.1.33 #define VERRSIGN 14

Signature error

Definition at line 119 of file voms.apic.h.

4.1.1.34 #define VERRTIME 7

Error in time checking

Definition at line 111 of file vomsapic.h.

4.1.1.35 #define VERRTYPE 18

Returned data of unknown type

Definition at line 123 of file vomsapic.h.

4.1.1.36 #define VERRVERIFY 17

Generic verification error

Definition at line 122 of file vomsapic.h.

4.1.2 Typedef Documentation

4.1.2.1 typedef void gsscred_id_t

Definition at line 19 of file vomsapic.h.

4.1.2.2 typedef void gssctx_id_t

Definition at line 20 of file vomsapic.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:

vd The vomsdata structure.

target The target to add. It should be a FQDN.

error 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 `vomsproxy_init` command, but without the `--include` functionality.

Parameters:

hostname FQDN of the VOMS server

port the port on which the VOMS server is listening

servsubject the subject of the server's certificate

command Command

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (\neq 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:

data and

datalen fields.

hostname FQDN of the VOMS server

port the port on which the VOMS server is listening

servsubject the subject of the server's certificate

command the command sent to the server

version is the version number of the data.

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (\neq 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:

v The structure to copy.

error 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 as a copy of another vomsdata structure.

Parameters:

vd The structure to copy.

error 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:

vd the vomsdata structure to analyze

error RETURN PARAMETER: Qualifies the error message

Returns:

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

4.1.3.10 void VOMSDelete (struct `voms` v)

Deletes a voms structure

Parameters:

v Pointer to the structure to delete.

4.1.3.11 void VOMSDeleteContacts (struct `contactdata` list)

Frees a contactdata vector.

Parameters:

list The vector to free.

Returns:

NONE

4.1.3.12 void VOMSDestroy (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:

vd The vomsdata structure to analyze

error The error returned by the last function

buffer 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.

len 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:

buffer OUTPUT PARAMETER contains the converted data

bu en OUTPUT PARAMETER contains the length of buffer

vd contains the data to convert

error 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:

vd The correctly initialized vomsdata structured.

alias The alias to look for.

system The directory in which to look for the system configuration files. If NULL, defaults to /opt/edc/etc/vomses

user The directory in which to look for the user configuration files. Defaults to \$VOMSUSERCONF if NULL. Again defaults to \$HOME/.edg/vomses if the latter is NULL, or to .edg/vomses as a last resort.

error 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:

vd The correctly initialized vomsdata structure.

vo The VO to look for.

system The directory in which to look for the system configuration files. If NULL, defaults to /opt/edc/etc/vomses

user The directory in which to look for the user configuration files. Defaults to \$VOMSUSERCONF if NULL. Again defaults to \$HOME/.edc/vomses if the latter is NULL, or to /etc/vomses as a last resort.

error 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 `VOMSDeleteContacts` function. Note also that the order in which the servers are returned is unspecified.

4.1.3.18 void VOMSFreeTargets (struct `vomsdata` vd, int error)

Delete the targets from the AC.

Parameters:

vd The vomsdata structure.

error RETURN PARAMETER: qualifies the error message.

- 4.1.3.19 void VOMSFreeTargetsList (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:

- buffer contains the data to be converted
- bu en contains the length of buffer
- vd RETURN PARAMETER: contains the data returned by the connection
- error 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:

- voms The directory which contains the certificates of the VOMS servers
- cert The directory which contains the CA certificates

If voms_dir is empty, the value of the environment variable X509VOMS_DIR is taken

If cert_dir is empty, the value of the environment variable X509CERT_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.Ordering (char order, struct [vomsdata](#) vd, int error)

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

Parameters:

order the group:role attribute.

vd RETURN PARAMETER: contains the modified data.

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.31 int VOMS.ResetOrder (struct [vomsdata](#) vd, int error)

Unsets the return order of the attributes.

Parameters:

vd RETURN PARAMETER: contains the modified data.

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.32 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:

cert The certificate with the VOMS extensions

chain The chain of the validation certificates (only the intermediate ones)

how Recursion type

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.33 int VOMS.RetrieveEXT (X509_EXTENSION ext, struct [vomsdata](#) vd, int error)

Gets VOMS information from the given extension

Parameters:

ext The extension to parse.

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.34 int VOMS.RetrieveFromCred (gsscred_id_t cred, int how, struct vomsdata vd, int error)

Gets VOMS information from the given globus credential

Parameters:

cred The credential from which to retrieve the certificate.

how Recursion type

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.35 int VOMS.RetrieveFromCtx (gssctx_id_t ctx, int how, struct vomsdata vd, int error)

Gets VOMS information from the given globus context

Parameters:

ctx The context from which to retrieve the certificate.

how Recursion type

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.36 int VOMS.RetrieveFromFile (FILE file, int how, struct vomsdata vd, int error)

Gets VOMS information from the given globus credential

Parameters:

file The file from which to retrieve the certificate.

how Recursion type

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.37 int VOMS.RetrieveFromProxy (int how, struct [vomsdata](#) vd, int error)

Gets VOMS information from an existing globus proxy

Parameters:

how Recursion type

vd RETURN PARAMETER: contains the data returned by the connection

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.38 int VOMS.SetLifetime (int length, struct [vomsdata](#) vd, int error)

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

Parameters:

length Lifetime requested.

vd RETURN PARAMETER: contains the modified data.

error RETURN PARAMETER: Qualifies the error message

Returns:

failure (0) or success (≠ 0)

4.1.3.39 int VOMS.SetVerificationTime (time_t verificationtime, struct [vomsdata](#) vd, int error)

4.1.3.40 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
 vomsdata,12
contact
 contactdata,6
contactdata,6
 contact,6
 host,6
 nick, 6
 port,6
 reserved,6
 version,6
 vo, 6
custom
 voms,9

data,8
 cap,8
 group,8
 role, 8
 vomsdata,12
datalen
 voms,9
date1
 voms,9
date2
 voms,9

extra.data
 vomsdata,12
extralen
 vomsdata,12

fqan
 voms,10

getMajorVersionNumber
 voms.apic.h,21

getMinorVersionNumber
 voms.apic.h,21
getPatchVersionNumber
 voms.apic.h,21
group
 data,8
gsscred_id_t
 voms.apic.h,21
gssctx_id_t
 voms.apic.h,21

holder
 voms,10
host
 contactdata,6

name
 attribute,5
nick
 contactdata,6

port
 contactdata,6

qualifier
 attribute,5

real
 vomsdata,12
RECURSECHAIN
 voms.apic.h,17
RECURSENONE
 voms.apic.h,17
reserved
 contactdata,6
role
 data,8

serial
 voms,10
server
 voms,10
serverca
 voms,10
siglen
 voms,10

- signature
 - voms,10
- std
 - voms,10
- type
 - voms,10
- TYPE_CUSTOM
 - voms.apic.h,17
- TYPE_NODATA
 - voms.apic.h,17
- TYPE_STD
 - voms.apic.h,17
- uri
 - voms,11
- user
 - voms,11
- 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,18
- VERIFY_NONE
 - voms.apic.h,18
- VERIFY_NOTARGET
 - voms.apic.h,18
- VERIFY_ORDER
 - voms.apic.h,18
- VERIFY_SIGN
 - voms.apic.h,18
- VERR_COMM
 - voms.apic.h,18
- VERR_DIR
 - voms.apic.h,18
- VERR_EXTRINFO
 - voms.apic.h,19
- VERR_FORMAT
 - voms.apic.h,19
- VERR_IDCHECK
 - voms.apic.h,19
- VERR_MEM
 - voms.apic.h,19
- VERR_NODATA
 - voms.apic.h,19
- VERR_NOEXT
 - voms.apic.h,19
- VERR_NOIDENT
 - voms.apic.h,19
- VERR_NOINIT
 - voms.apic.h,19
- VERR_NONE
 - voms.apic.h,19
- VERR_NOSOCKET
 - voms.apic.h,20
- VERR_NOTAVAIL
 - voms.apic.h,20
- VERR_ORDER
 - voms.apic.h,20
- VERR_PARAM
 - voms.apic.h,20
- VERR_PARSE
 - voms.apic.h,20
- VERR_SERVER
 - voms.apic.h,20
- VERR_SERVERCODE
 - voms.apic.h,20
- VERR_SIGN
 - voms.apic.h,20
- VERR_TIME
 - voms.apic.h,20
- VERR_TYPE
 - voms.apic.h,21
- VERR_VERIFY
 - voms.apic.h,21
- version
 - contactdata,6
 - voms,11
- vo
 - contactdata,6
- volen
 - vomsdata,12
- voms,9
 - ac,9
 - custom,9
 - datalen,9
 - date1,9
 - date2,9
 - fqan,10
 - holder,10
 - serial,10
 - server,10
 - serverca,10
 - siglen,10
 - signature,10
 - std,10

- type, [10](#)
- uri, [11](#)
- user, [11](#)
- userca, [11](#)
- version, [11](#)
- voname, [11](#)
- VOMS.AddTarget
 - voms.apic.h, [21](#)
- voms.apic.h, [15](#)
 - getMajorVersionNumber, [21](#)
 - getMinorVersionNumber, [21](#)
 - getPatchVersionNumber, [21](#)
 - gsscred_id_t, [21](#)
 - gssctx_id_t, [21](#)
 - RECURSECHAIN, [17](#)
 - RECURSENONE, [17](#)
 - TYPE.CUSTOM, [17](#)
 - TYPE.NODATA, [17](#)
 - TYPE.STD, [17](#)
 - VERIFY_CERTLIST, [18](#)
 - VERIFY_DATE, [18](#)
 - VERIFY_FULL, [18](#)
 - VERIFY_ID, [18](#)
 - VERIFY_KEY, [18](#)
 - VERIFY_NONE, [18](#)
 - VERIFY_NOTARGET, [18](#)
 - VERIFY_ORDER, [18](#)
 - VERIFY_SIGN, [18](#)
 - VERR.COMM, [18](#)
 - VERR.DIR, [18](#)
 - VERR.EXTRAINFO, [19](#)
 - VERR.FORMAT, [19](#)
 - VERR.IDCHECK, [19](#)
 - VERR.MEM, [19](#)
 - VERR.NODATA, [19](#)
 - VERR.NOEXT, [19](#)
 - VERR.NOIDENT, [19](#)
 - VERR.NOINIT, [19](#)
 - VERR.NONE, [19](#)
 - VERR.NOSOCKET, [20](#)
 - VERR.NOTAVAIL, [20](#)
 - VERR.ORDER, [20](#)
 - VERR.PARAM, [20](#)
 - VERR.PARSE, [20](#)
 - VERR.SERVER, [20](#)
 - VERR.SERVERCODE, [20](#)
 - VERR.SIGN, [20](#)
 - VERR.TIME, [20](#)
 - VERR.TYPE, [21](#)
 - VERR.VERIFY, [21](#)
 - VOMS.AddTarget, [21](#)
 - VOMS.Contact, [21](#)
 - VOMS.ContactRaw, [22](#)
 - VOMS.Copy, [22](#)
 - VOMS.CopyAll, [22](#)
 - VOMS.DefaultData, [23](#)
 - VOMS.Delete, [23](#)
 - VOMS.DeleteContacts, [23](#)
 - VOMS.Destroy, [23](#)
 - VOMS.Duplicate, [23](#)
 - VOMS.ErrorMessage, [24](#)
 - VOMS.Export, [24](#)
 - VOMS.FindByAlias, [24](#)
 - VOMS.FindByVO, [24](#)
 - VOMS.FreeTargets, [25](#)
 - VOMS.FreeTargetsList, [25](#)
 - VOMS.GetAC, [26](#)
 - VOMS.GetAttribute, [26](#)
 - VOMS.GetAttributeGrantor, [26](#)
 - VOMS.GetAttributesNumber, [26](#)
 - VOMS.GetAttributeSourceHandle, [26](#)
 - VOMS.GetAttributeSourcesNumber, [26](#)
 - VOMS.GetTargetsList, [26](#)
 - VOMS.Import, [26](#)
 - VOMS.Init, [26](#)
 - VOMS.ListTargets, [26](#)
 - VOMS.Ordering, [27](#)
 - VOMS.ResetOrder, [27](#)
 - VOMS.Retrieve, [27](#)
 - VOMS.RetrieveEXT, [27](#)
 - VOMS.RetrieveFromCred, [28](#)
 - VOMS.RetrieveFromCtx, [28](#)
 - VOMS.RetrieveFromFile, [28](#)
 - VOMS.RetrieveFromProxy, [28](#)
 - VOMS.SetLifetime, [29](#)
 - VOMS.SetVerificationTime, [29](#)
 - VOMS.SetVerificationType, [29](#)
 - VOMS.Contact
 - voms.apic.h, [21](#)
 - VOMS.ContactRaw
 - voms.apic.h, [22](#)
 - VOMS.Copy
 - voms.apic.h, [22](#)
 - VOMS.CopyAll
 - voms.apic.h, [22](#)
 - VOMS.DefaultData
 - voms.apic.h, [23](#)
 - VOMS.Delete
 - voms.apic.h, [23](#)
 - VOMS.DeleteContacts
 - voms.apic.h, [23](#)
 - VOMS.Destroy
 - voms.apic.h, [23](#)
 - VOMS.Duplicate
 - voms.apic.h, [23](#)
 - VOMS.ErrorMessage
 - voms.apic.h, [24](#)
 - VOMS.Export

- voms.apic.h, [24](#)
- VOMS.FindByAlias
 - voms.apic.h, [24](#)
- VOMS.FindByVO
 - voms.apic.h, [24](#)
- VOMS.FreeTargets
 - voms.apic.h, [25](#)
- VOMS.FreeTargetsList
 - voms.apic.h, [25](#)
- VOMS.GetAC
 - voms.apic.h, [26](#)
- VOMS.GetAttribute
 - voms.apic.h, [26](#)
- VOMS.GetAttributeGrantor
 - voms.apic.h, [26](#)
- VOMS.GetAttributesNumber
 - voms.apic.h, [26](#)
- VOMS.GetAttributeSourceHandle
 - voms.apic.h, [26](#)
- VOMS.GetAttributeSourcesNumber
 - voms.apic.h, [26](#)
- VOMS.GetTargetsList
 - voms.apic.h, [26](#)
- VOMS.Import
 - voms.apic.h, [26](#)
- VOMS.Init
 - voms.apic.h, [26](#)
- VOMS.ListTargets
 - voms.apic.h, [26](#)
- VOMS.Ordering
 - voms.apic.h, [27](#)
- VOMS.ResetOrder
 - voms.apic.h, [27](#)
- VOMS.Retrieve
 - voms.apic.h, [27](#)
- VOMS.RetrieveEXT
 - voms.apic.h, [27](#)
- VOMS.RetrieveFromCred
 - voms.apic.h, [28](#)
- VOMS.RetrieveFromCtx
 - voms.apic.h, [28](#)
- VOMS.RetrieveFromFile
 - voms.apic.h, [28](#)
- VOMS.RetrieveFromProxy
 - voms.apic.h, [28](#)
- VOMS.SetLifetime
 - voms.apic.h, [29](#)
- VOMS.SetVerificationTime
 - voms.apic.h, [29](#)
- VOMS.SetVerificationType
 - voms.apic.h, [29](#)
- vomsdata, [12](#)
 - cdir, [12](#)
 - data, [12](#)
 - extra.data, [12](#)
 - extralen, [12](#)
 - real, [12](#)
 - vdir, [12](#)
 - volen, [12](#)
 - workvo, [13](#)
- voname
 - voms, [11](#)
- workvo
 - vomsdata, [13](#)