

Getting a Certificate

Last review date	Reviewer
2009-09-15	Marco Bencivenni Enrico Fattibene

Table of Contents

[Getting a Certificate](#)

[Digital Certificates](#)

[Requesting the Certificate](#)

[Getting the Certificate](#)

[Renewing the Certificate](#)

[Taking Care of Private Keys](#)

Getting a Certificate

How to apply for and install a certificate to gain access to the Grid.

Digital Certificates

The first requirement the user must fulfill is to be in possession of a valid digital certificate issued by a recognized *Certification Authority* (CA). The type of digital certificate used is called X.509. The role of a CA is to guarantee that users are who they claim to be and are entitled to own their certificate. It is up to the users to discover which CA they should contact. In general, CAs are organised geographically and by research institute. Each CA has its own procedure to release certificates.

The following URL maintains an updated list of recognised CAs, as well as detailed information on how to request certificates from a particular CA:

<http://www.eugridpma.org/members/worldmap/>

For many purposes it may be useful to install the root certificates of Grid CAs in a web browser and/ or email client, as this will enable the validation of Grid certificates used in web servers and to sign email. Installing the root certificate is browser- specific and is not covered here. In particular, users should install the root certificate for their own CA. The root certificates can be obtained from this URL:

<https://www.tacar.org/repos/>

Requesting the Certificate

In order to obtain a certificate, users must create a request to a CA. The request is normally generated using either a web- based interface or console commands. Details of which type of request a particular CA accepts can be found on each CA's website.

For a web- based certificate request, a form must usually be filled in with information such as the name of the user, home institute, etc. After submission, a pair of private and public keys are generated, together with a request for the certificate containing the public key and the user data. The request is then sent to the CA, while the private key stays in the browser, hence the same browser must be used to retrieve the certificate once it is issued.

Users must usually install the CA root certificate in their browser first. This is because the CA has to sign the user certificate using its private key, and the user's browser must be able to validate the signature.

For some CAs, the certificate requests are generated using a command line interface. The following shows how to make a command- line certificate application using the `grid- cert- request` command. Again, details of the exact command and the requirements of each CA will vary and can be found on the CA's website.

The `grid- cert- request` creates, for example, the following 3 files:

userkey.pem	contains the private key associated with the certificate. This should be set with permissions so that only the owner can read it, i.e. <code>chmod 400 userkey.pem</code> ;
userreq.pem	contains the request for the user certificate (essentially the public key);
usercert.pem	a placeholder, to be replaced by the actual certificate when received from the CA (this can be readable by anyone), i.e. <code>chmod 444 usercert.pem</code>

Then the `userreq.pem` file has to be sent (usually by e-mail) to the desired CA. This is the standard way to complete a certificate request and it is recommended that this procedure is followed.

Getting the Certificate

After a request is generated and sent to a CA, that CA will have to confirm the users' authenticity through their certificate.. This usually involves a physical meeting or a phone call with a Registration Authority (RA). A RA is delegated by the CA to verify the legitimacy of a request, and approve it if it is valid. The RA is usually someone at the user's home institute, and will generally need some kind of ID to prove the user's identity.

After approval, the certificate is generated and delivered to the user. This can be done via e-mail, or by giving instructions to the user to download it from a web page. If the certificate was directly installed in the user's browser then it must be exported (saved) to disk for Grid use. Details of how to do this will depend on the browser, and are described on the CA web site.

The received certificate will usually be in one of two formats: *Privacy Enhanced Mail Security Certificate (PEM)* with extension `.pem` or *Personal Information Exchange File (PKCS12)* with extensions `.p12` or `.pfx`. The latter is the most common for certificates exported from a browser (e.g. Internet Explorer, Mozilla and Firefox), but the `PEM` format is currently needed on a *World Wide LCG (WLCG)/ EGEE* user interface. The certificates can be converted from one format to the other using the `openssl` command.

If the certificate is in `PKCS12` format, then it can be converted to `PEM` using:

```
$ openssl pkcs12 -nocerts -in my_cert.p12 -out userkey.pem
```

If only the certificate, without the private key, is to be exported the command is:

```
$ openssl pkcs12 -clcerts -nokeys -in my_cert.p12 -out usercert.pem
```

where:

my_cert.p12	is the input <code>PKCS12</code> format file;
userkey.pem	is the output private key file;
usercert.pem	is the output <code>PEM</code> certificate file.

The first command creates only the private key (due to the `-nocerts` (no certificate in output) option), and the second one creates the user certificate (`-clcerts` (create only client certificate) - `nokeys` (do not output keys) option). For further information on the options of the `pkcs12` command, consult

```
$ man pkcs12
```

It is strongly recommended that the names of all these files are kept as shown.

The command of the form

```
grid-change-pass-phrase -file <private_key_file>
```

changes the pass phrase that protects the private key. This command will work even if the original key is not password protected. If the user loses the passphrase, the certificate will become unusable and a new certificate will have to be requested.

Once in `PEM` format, the two files, `userkey.pem` and `usercert.pem`, should be copied to a *User Interface (UI)*.

Renewing the Certificate

CAs issue certificates with a limited duration (usually one year); this implies the need to renew them periodically. The renewal procedure usually requires that the certificate holder sends a request for renewal signed with the old certificate and/ or that the request is confirmed by a phone call; the details depend on the policy of the CA. The certificate usually needs to be renewed before the old certificate expires; CAs may send an email to remind users that renewal is necessary, but users should try to be aware of the renewal date, and take appropriate action if they are away for extended periods of time.

Taking Care of Private Keys

A private key is the essence of a Grid identity. Anyone who steals it can impersonate the owner and if it is lost, it is no longer possible to do anything in the Grid. Certificates are issued personally to individuals, and must never be shared with other users. To use the Grid, users must agree to an Acceptable Use Policy, which among other things requires them to keep their private key secure.

On a UNIX UI, the certificate and private key are stored in two files. Typically they are in a directory called `$HOME/.globus` and are named `usercert.pem` and `userkey.pem`, and it is strongly recommended that they are not changed. The certificate is public and world-readable, but the key must only be readable by the owner. The key should be stored on a disk local to the user's UI rather than, for example, an NFS-mounted disk. If a certificate has been exported from a browser, a PKCS12-format file (`.p12` or `.pfx`), which contains the private key, will have been locally stored and this file must be either encrypted, hidden or have its access rights restricted to only the owner.

If a private key is stored under the *Andrew File System (AFS)*, access is controlled by the AFS Access Control Lists (ACL) rather than the normal file permissions, so users must ensure that the key is not in a publicly-readable area.

Web browsers also store private keys internally, and these also need to be protected. The details vary depending on the browser, but password protection should be used if available; this may not be the default (it is not with Internet Explorer). The most secure mode is one in which every use of the private key needs the password to be entered, but this can cause problems as some web sites ask for the certificate many times. Reaching a compromise between security and convenience is vital here, so that neither come too short.

It is important not to lose the private key, as this implies loss of all access to the Grid, and registration will have to be started again from scratch. Having several securely protected copies in different places is strongly advised, so the certificate can be used from a web browser and several UI machines.

A private key stored on a UI must be encrypted, meaning that a passphrase must be typed whenever it is used. A key must never be stored without a passphrase. The passphrase should follow similar rules to any computer password. Users should be aware of the usual risks, like people watching them type or transmitting the passphrase over an insecure link.