Summer Grid Workshop Lab 2: Grid Security

Outline:

- Requesting a DoE Grid certificate
- Using a local grid certificate
- Using MyProxy to store a credential proxy
- Verifying your DoE Grid certificate

In this lab, we'll be dealing with security credentials issued by two different Certificate Authorities. One of the CAs is run by the Department of Energy; the other is a simple CA we have set up just for this workshop.

We'll be storing certificate files in the directory **~/.globus/UTB** and **~/.globus/DOE**. Take a look now in your ~/.globus/UTB directory to find the name of the subdirectory your UTB certificates are in; it should be your name. We'll call that your *username*, and you'll use it several times in this lab.

Create a directory of the same name within ~/.globus/DOE:

- **cd ~/.globus/DOE**
- **mkdir ~/.globus/DOE/***username*

A. Requesting a DoE Grid Certificate

In this part of the lab, we'll request a Grid Certificate from the DoE CA; we'll test the cert we're issued later in the lab. The instructions we give you for this section may be modified between the time this page was printed and the actual lab.

1. REQUESTING A CERT

Open the Firefox browser.
Go to https://pki1.doegrids.org and fill out your info.

Subscriber email is necessary, phone is not.
Affiliation = iVDGL
Sponser Name = Mike Wilde
Sponser Email = wilde@mcs.anl.gov
Sponser Phone = 630-252-7497
Key Length = 1024(Medium Grade)

Click Submit

2. GETTING APPROVAL

Bring Request ID to Rob.

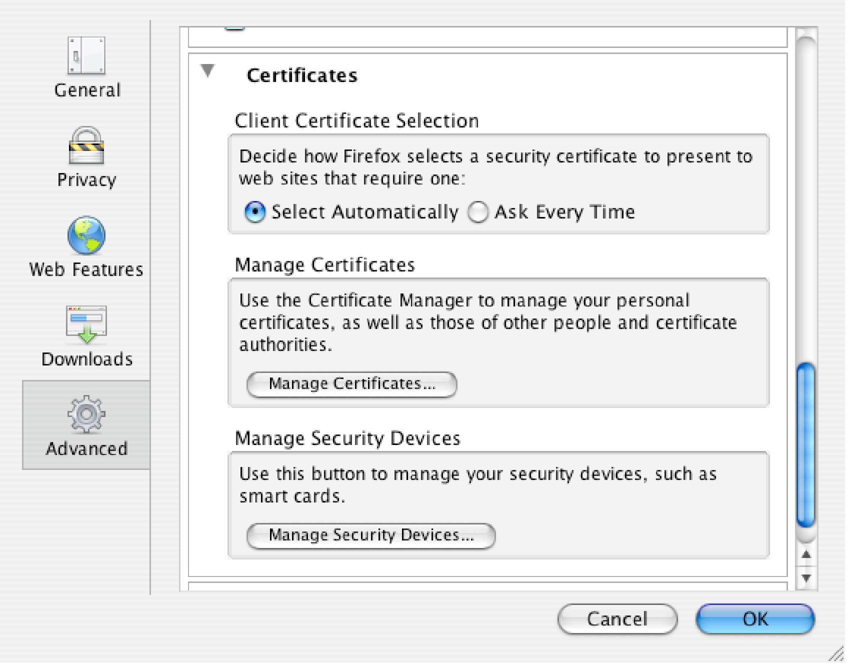
Rob will approve the request and mail will be sent to your account in approximately 2 minutes.

3. IMPORTING AND BACKING UP YOUR CERT

Follow the link from the mail you get.
Click "Import Your Certificate"

Go to the Firefox menu and choose Preferences.
Click on the Advanced button and scroll down to the Certificates box.

Click on the Manage Certificates button:



Certificates

Client Certificate Selection

Decide how Firefox selects a security certificate to present to web sites that require one:

● Select Automatically ○ Ask Every Time

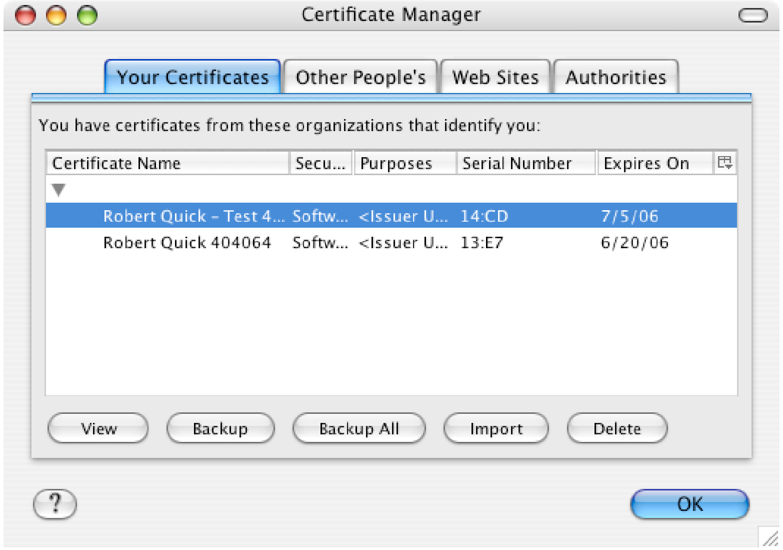
Manage Certificates

Use the Certificate Manager to manage your personal certificates, as well as those of other people and certificate authorities.

Manage Security Devices

Use this button to manage your security devices, such as smart cards.

Click on the Certificate you would like to use and click Backup:


Certificate Manager

You have certificates from these organizations that identify you:

Certificate Name	Secu...	Purposes	Serial Number	Expires On
Robert Quick - Test 4...	Softw...	<Issuer U...	14:CD	7/5/06
Robert Quick 404064	Softw...	<Issuer U...	13:E7	6/20/06

Save as "YourNameCert" (ie robcert)

Choose a Certificate backup password and click OK.

4. CREATING YOUR USERCERT AND USERKEY FOR USE

Create a .globus directory in your home directory.
Copy the .p12 file to the .globus directory.
Set up the VDT shell (source $VDT_LOCATION/setup.sh).

Run the following command to extract the usercert.pem:

```
     $GLOBUS_LOCATION/bin/openssl pkcs12 -in YourNameCert.p12
—clcerts -nokeys -out $HOME/.globus/usercert.pem
```

Run the following command to extract the userkey.pem.

```
     $GLOBUS_LOCATION/bin/openssl pkcs12 -in YourNameCert.p12  -
nocerts
-out $HOME/.globus/DOE/username/userkey.pem
```

Change permissions on the userkey.pem

```
     chmod 400 $HOME/.globus/userkey.pem
```

5. GETTING CERT ENTERED TO Grid3-iVDGL VOMS

Send mail to rquick@iupui.edu with your full DN. (ie.
`/DC=org/DC=doegrids/OU=People/CN=Robert Quick 404064`)

Rob will enter this into the iVDGL VOMS and soon -- by the end of this lab, we hope -- you will be able to run Grid3 jobs!

B. Using a Local Grid Certificate

For the next few sections we're going to be using this laptop as a dumb terminal to shell into another server on which you have an account, gk1.phys.utb.edu.
Open the Terminal program and type

- ssh gk1.phys.utb.edu
- If the computer complains that "the authenticity of host gk1 can't be established", answer "yes" to the "continue connecting?" question.
- Your password is the same as your computer's name, training*XX*, for whatever your number is.
- Your prompt should now identify your host as gk1.

We have already created another grid certificate for you, using a temporary Certificate Authority on our local grid rather than the DoE Certificate Authority.

GSI expects your certificates to be at **~/.globus/usercert.pem** and **~/.globus/userkey.pem**. Since we're going to be dealing with more than one kind of certificate in this lab, we won't actually move and rename the keys. Instead we'll use a symbolic link to point to your certificates. This will keep you from overwriting your Simple CA certificates when you receive your DoE certificates.

Putting your certificate in place

- **ls .globus/UTB**

You should see two subdirectories listed, one for each user. Pick one, for whoever is going to go first. We'll refer to these directory names as username in this lab, because you'll use that specific name for several other purposes. Now we'll remove the links to the certificates left over from the previous lab and show how to link the certificates in.

- **cd ~/.globus/**
- **rm usercert.pem userkey.pem**
- **ln -s UTB/***username***/usercert.pem usercert.pem**
- **ln -s UTB/***username***/userkey.pem userkey.pem**

Looking at your certificate

Does this certificate "know" who you are? We can investigate what's in the certificate using **grid-cert-info**.

- **grid-cert-info -subject**

Do you see your distinguished name?

When is the end date according to **grid-cert-info –enddate**?

- **grid-cert-info -help**

Creating a Grid proxy

GSI doesn't check your private key directly; instead you generate a grid proxy certificate, and it's this grid proxy that acts as your key to grid services. Remember, the certificates were generated using your username as a password; you'll need it for grid-proxy-init.

- Make a proxy with the command: **grid-proxy-init**
- What can you learn about your proxy with **grid-proxy-info**?
- Make a proxy that is good for something other than the default of twelve hours using the **-hours** switch

Submitting jobs

Now that you've got a grid proxy, you can check to see whether you really do have access to grid services. In other words, you can check to see whether GSI accepts your grid proxy as valid identification.

- Check that the gatekeeper "knows" you:
  - **globusrun –a –r gk1/jobmanager-fork**
  - Look for "successful" message.
  - This means you are "authorized" to use the resource
  - If this fails, try in a few minutes – let your partner try the above steps, or just wait. If this still fails, we were unable to authorize you, so contact an instructor.
- Submit a simple job such as /bin/hostname to the "fork jobmanager" – the one that runs simple jobs by simply using "fork" on the gatekeeper (like ssh) – using globus-job-run
  - **globus-job-run gk1 /bin/hostname**
  - What else can you learn with this? Eg, how can you find your uid, etc?
  - /usr/bin/env (real important!), uptime, pwd, ls?
    (Remember to use full paths!)
  - Also: please PRETEND your home directory is NOT available here!
  - Try running a shell command:

```
/bin/sh –c 'cmd here'          (use this, e.g. to find your $PATH!)
```

- try this on gk2 as well

Extra credit section: Resource Specification Language

- Extra credit: look into “RSL”: try the above with the option “–dumprsl”, then try globusrun with your own rsl. This is the Globus GRAM “Resource Specification Language” – Google for more info.
  - **globusrun – r gk1 –o '&(executable=/bin/pwd)'**
  - Write a small shell script to dump more about your environment in one command. Try it locally.

```
#!/bin/bash
date
hostname -f
uptime
env
cat /etc/issue
```

  Then: globus-job-run gk1 –s *my-scriptname*

  Extra extra credit: Try “staging” a binary – copying it from your job submission host to the execution host (such as the local /bin/date)

- What RSL does globus-job-run create?
- Submit a simple job such as /bin/hostname to the Condor jobmanager using globus-job-run and globusrun. The “*contact string*” is gk1/jobmanager-condor rather than just “gk1” (which is a default name for **gk1/jobmanager-fork**)
- Create a new program in your favorite language. C, Bourne shell, Perl, and Python are reasonable choices. This program should take one argument, and integer, and return the square of that argument. Run this program on our local grid for value. You'll need to stage the executable--see the -s command for globus-job-run. (What RSL do you get now?) Run this program for n in [10, 30], and collect the results in distinct files. **Extra Credit:** Make a script to do all of this for you.

C. Using MyProxy to store your certificate in a credential repository

Right now your grid credentials are associated with one particular host, the one you’re on right now -- that is, gk1.phys.utb.edu. Now we’re going go through some steps that will allow you to use that certificate on another host, gk2.phys.utb.edu.

We’ve put a MyProxy repository on gk1.phys.utb.edu. In this part of the lab, we’ll check our grid proxy certificate into the repository, shell into gk2, and from there retrieve and use a delegated proxy certificate. (Usually, the MyProxy repository won’t be on the same host as the one you’re logged into, but in this lab, it is.)

First, we need to check our grid proxy certificate into the trusted MyProxy repository. The repository is on gk1, so do this:

- **myproxy-init -s gk1.phys.utb.edu**

This command will prompt you for your grid credential's passphrase. That's the same one you used to get the grid credential. It will also prompt you for *another* passphrase, a new one to be associated with your MyProxy proxy. (Why two different passwords? This way, if your MyProxy credential gets compromised, it won't permanently compromise your private key.) What does the response tell you about the proxy you've deposited in the MyProxy repository?

- **myproxy-info -s gk1.phys.utb.edu**

Now let's go to another host.

- **ssh gk2.phys.utb.edu**

Do we have a grid proxy here already?

- **grid-proxy-info**

Nope. Let's try a grid job anyway.

- **globusrun -o -r gk2 '&(executable=/usr/bin/cal)'**

Crash & burn? But this host trusts the MyProxy repository we've set up on gk1, so we can check out a delegated grid proxy.

- **myproxy-get-delegation -s gk1.phys.utb.edu**
- **grid-proxy-info**

What is the default time limit on the delegated proxy? Let's test it out to make sure we're an authenticated user:

- **globusrun -o -r gk2 '&(executable=/usr/bin/cal)'**

We have created two MyProxy-related items with time limits: the delegated proxy certificate on gk2, and the registered proxy certificate in the MyProxy repository. Rather than letting them time out, let's destroy both of them.

- **grid-proxy-destroy**

That should spell doom for the delegated proxy certificate, and our grid example should no longer run.

- **globusrun -o -r gk2 '&(executable=/usr/bin/cal)'**

A delegated proxy credential can still be checked out of the MyProxy credential repository until that registration runs out. If we want to remove the proxy credential from MyProxy directly before that, we can use myproxy-destroy.

- **myproxy-destroy -s gk1.phys.utb.edu**
- **myproxy-info**

D. Using your DoE Certificate

By now (we hope) your DoE Certificate should be ready to use. Earlier, you put your DOE keys into ~/.globus/DOE/username. We'll now link to those instead of the SimpleCA certificate we used earlier.

- **cd ~/.globus/**
- **rm usercert.pem userkey.pem**
- **ln -s DOE/***username***/usercert.pem usercert.pem**
- **ln -s DOE/***username***/userkey.pem userkey.pem**

If all went well, you should be able to use your DOE certificate now:

- **globusrun -o -r gk2 '&(executable=/usr/bin/cal)'**

This is the certificate you'll be using through the rest of the week.