Certificate Authority Module

lab 1 issue and revoke of certificate authority

Introduction

Quick introduction to LAMP

LAMP is short for the software bundle of Linux operating system, Apache HTTP Server, MySQL database management system and PHP programming language.

This bundle can realize the role and function of a web server, which can drive Web applications. Although not actually designed to work together, these open source software is comparatively simple and easy to use.

Besides this four software, this software bundle can also be combined with many other free and open-source software packages

Quick introduction to certificate authority

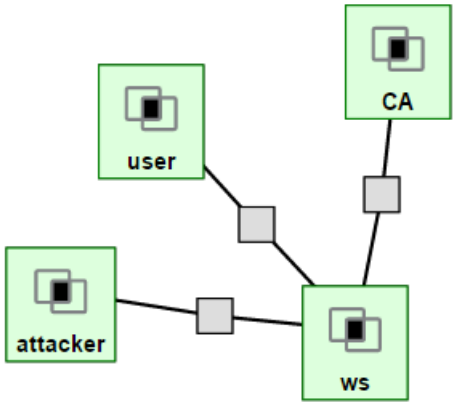
Certificate Authority is a trusted third party that issues electronic documents that verify a digital entity’s identification on the Internet.

In cryptography mean, certificate authority verifies the ownership of the public key of the named subject of the certificate.

Implementation

Network topology on GENI

The network topology of our experiment will be the following one:



The node named CA will be the certificate authority in this experiment. The node named WS will be the web server in this experiment and we will install LAMP on this node to enable it to be a web server.

Notice: we must wait for all the GENI node turn green, which means the remote machines are ready for us to use, then we can continue the following steps. This may take a while.

Setup LAMP on GENI

We already have the GENI node. In other words, we already have a Linux operation system, so we only need to install the remaining Apache, MySQL, and PHP. We should pay attention to the installation order of LAMP, I recommend we install the MySQL and Apache firstly, leave the PHP in the end. The order of installation of MySQL and Apache can be reversed because they are not depending on each other. However, the PHP must be installed after we finish the installation of MySQL and

Apache because PHP server depends on the services of Apache and MySQL.

Using SSH log onto the WS node. The following installation will be on this node.

Before installation, we should download the package lists from the repositories and "updates" them to get information on the newest versions of packages and their dependencies.

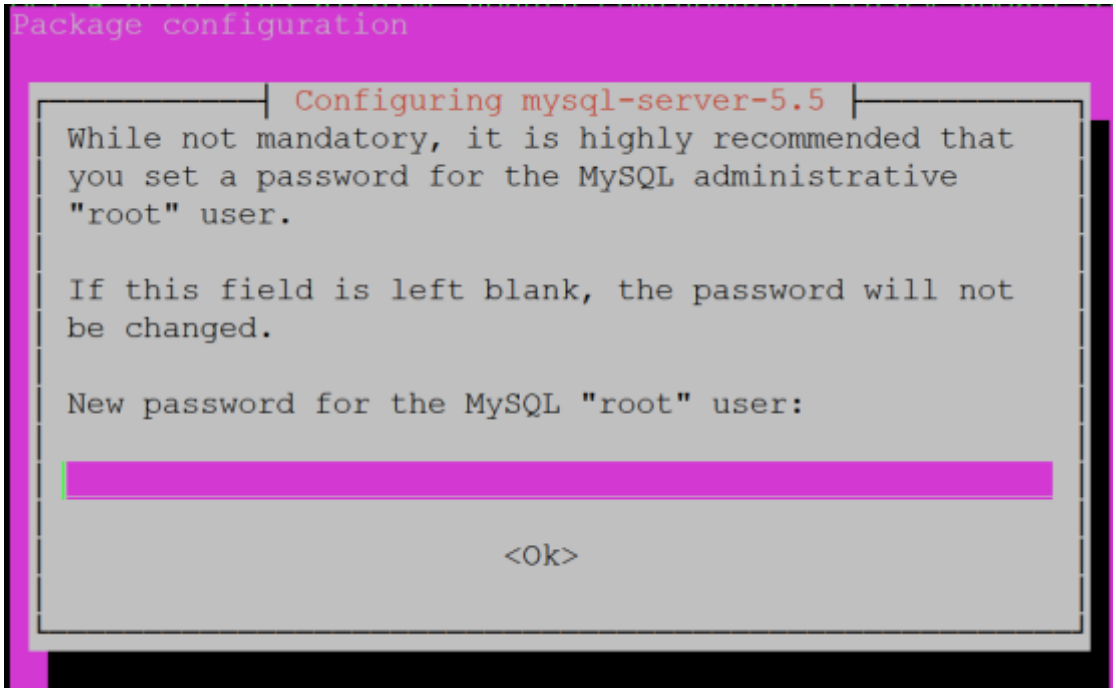
Command: “sudo apt-get update”

Install MySQL:

Command: “sudo apt-get install mysql-server”

```
kqing051@ws:~$ sudo apt-get install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
```

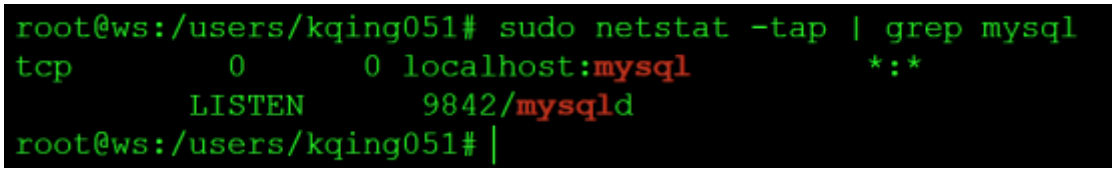
In this process, it will ask you to enter the password for the MySQL administrator, set up the password in this prompt window.



After installation of MySQL, we should check whether this is installed successfully.

Command:" sudo netstat -tap | grep mysql"

If it shows the listening port of MySQL as following, then we prove it is installed successfully.



Install Apache

Command: "sudo apt-get install apache2"

```
root@ws:/users/kqing051# sudo apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-bin apache2-data libaprutil1-dbd-sqlite3 libaprut
il1-ldap
```

We can run the browser to check whether it is installed successfully or not. But don' t hurry; we need to involve third party software to enable the graphics showing on GENI node. We will cover this content in the following.

Install PHP

Command:" sudo apt-get install php5 libapache2-mod-php5"

```
root@ws:/users/kqing051# sudo apt-get install php5 libapach
e2-mod-php5
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  php5-cli php5-common php5-json php5-readline
```

After this installation, it will create a folder named "www" under var. This folder will be reserve the source code of the website.

```
root@ws:/var# ls
backups  emulab  local  log   opt  spool  www
cache    lib     lock   mail  run  tmp
```

Install the extension packets of PHP and MySQL

Command:" sudo apt-get install php5-mysql php5-curl php5-gd php5-intl php-pear php5-imagick php5-imap php5-mcrypt php5-memcache php5-ming php5-ps php5-pspell php5-recode php5-snmp php5-sqlite php5-tidy php5-xmlrpc php5-xsl;"

Notice: the GENI node is initiated with an Ubuntu operation system following its default setting. If you choose other operation systems rather the default operation system, there might be a warning message showing on the screen and it may be failed to initiate the GENI nodes. Therefore there is no need to set up another operation system, but if you do so, either it will not affect much in this experiment.

Enable SSL connection of Apache

Because we need to install the digital certificate later, so we need to enable SSL connection of Apache.

First, just a brief introduction of Apache configuration file.

```
root@ws:/etc/apache2# ls
apache2.conf    magic           sites-available
conf-available  mods-available  sites-enabled
conf-enabled    mods-enabled
envvars         ports.conf
root@ws:/etc/apache2#
```

As we can see, there are several configurations in Apache folder.

In the old versions of Apache, there is only one configuration named "httpd.conf".

And as for the latest version, the main configuration file is "apache2.conf". We can take a quick look of this file.

```
#
# LogLevel: Control the severity of messages logged
error_log.
# Available values: trace8, ..., trace1, debug, inf
e, warn,
# error, crit, alert, emerg.
# It is also possible to configure the log level fo
ular modules, e.g.
# "LogLevel info ssl:warn"
#
LogLevel warn

# Include module configuration:
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf

# Include list of ports to listen on
Include ports.conf
```

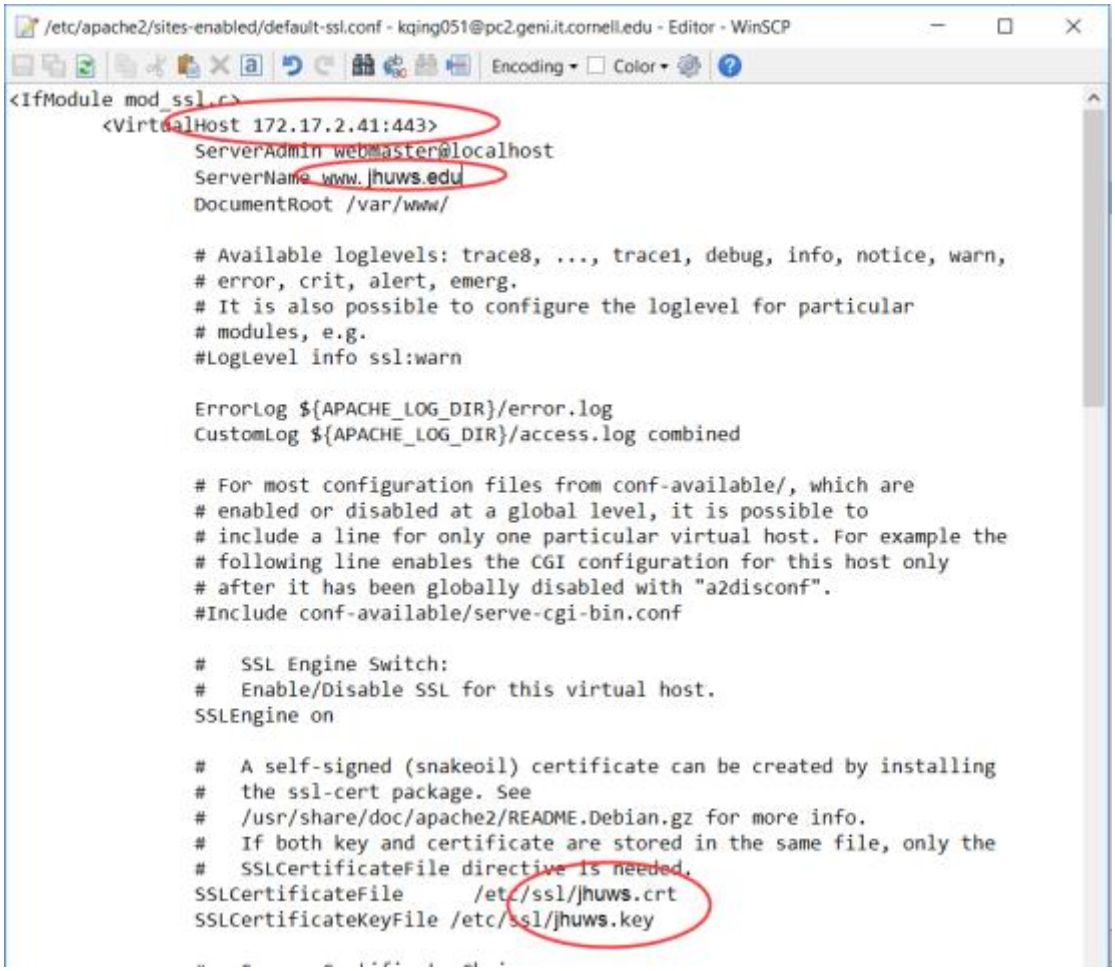
As we can see, there are many "includes" command in this file. It means the apache server will firstly read this file and the other configuration files will be linked using these "include" command.

As for the SSL configuration file, it's in the "sites-enabled" folder.

We can use this command to create a configuration file for SSL connection.

“sudo cp /etc/apache2/sites-available/default-ssl.conf /etc/apache2/sites-enabled/default-ssl.conf ”

Then we modify this default-ssl.conf like follows:



We temporary named our server name as jhuws.edu, and the digital certificate name is jhuws.crt, the private key of digital certificate name is jhuws.key. Of course you can make the name as you like, but make sure to use the same name in the later.

And the 172.17.2.41 is the IP address of our web server, you can use

command “ifconfig” to check it out. The “443” is the port number for SSL connection.

You should also use this command” sudo a2enmod ssl” to enable the SSL module of Apache2 if there prompt the problem that you try to connect 443 port to our web server while it refuse.

Then restart the apache2 service using this command” service apache2 restart”.

Install and enable browser on GENI

We choose Firefox browser in this experiment. Of course, you can choose other browsers if you like. This part should be done on user node.

Command:” sudo apt-get install firefox”

```
kqing051@user:~$ sudo apt-get install firefox
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libcanberra0 libdbusmenu-glib4 libdbusmenu-gtk4 libgtk2.0-0 libgtk2.0-bin
  libgtk2.0-common libogg0 libstartup-notification0 libtdb1 libvorbis0a
  libvorbisfile3 libxcb-util0 sound-theme-freedesktop xul-ext-ubufox
```

The operation of next step will be different for windows, Mac and Linux operation systems. For windows and MacOS operation systems, we need to depend on third party software to enable the graphics display on GENI node.

For Windows operation system:

Install the Xming software on your local operation system. Xming is an X11 display server for Microsoft Windows operating systems. Then run it to start the X server. You should see the Xming icon in the taskbar if it is running.

locale	2017/2/12 18:14
xkb	2017/2/12 18:14
COPYING	2007/4/15 20:22
example_Xmingrc	2006/10/29 15:01
font-dirs	2007/1/31 8:38
libfreetype-6.dll	2007/5/4 9:31
plink.exe	2007/5/2 8:16
pthreadGC2.dll	2007/3/27 13:42
rgb.txt	2004/11/11 0:14
run.exe	2007/3/27 14:15
SecurityPolicy	2006/2/13 9:02
unins000.dat	2017/2/12 18:14
unins000.exe	2017/2/12 18:13
X0.hosts	2006/3/8 19:40
XErrorDB	2005/11/24 10:52
xkbcomp.exe	2007/5/3 18:43
XKeysymDB	2004/11/11 15:16
XLaunch.chm	2007/4/13 12:17
XLaunch.exe	2007/5/3 18:43
XLaunch.xsd	2007/1/22 15:56
Xming.exe	2007/5/3 18:43
Xming	2017/2/12 18:14

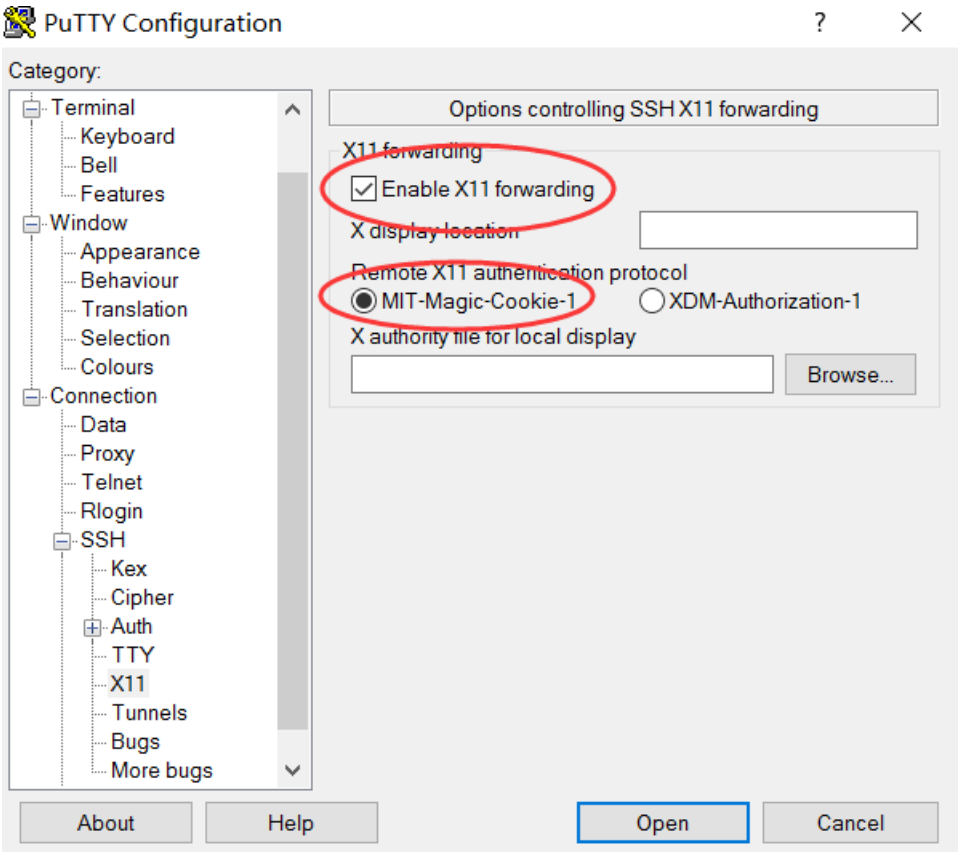
Then use PuTTY to log onto the GENI node.

You can see the instruction here about how to log onto GENI node using PuTTY.

http://groups.geni.net/geni/wiki/HowTo/LoginToNodes

Remember to click the option on X11 option besides the other steps of

logging onto GENI node using PuTTY.

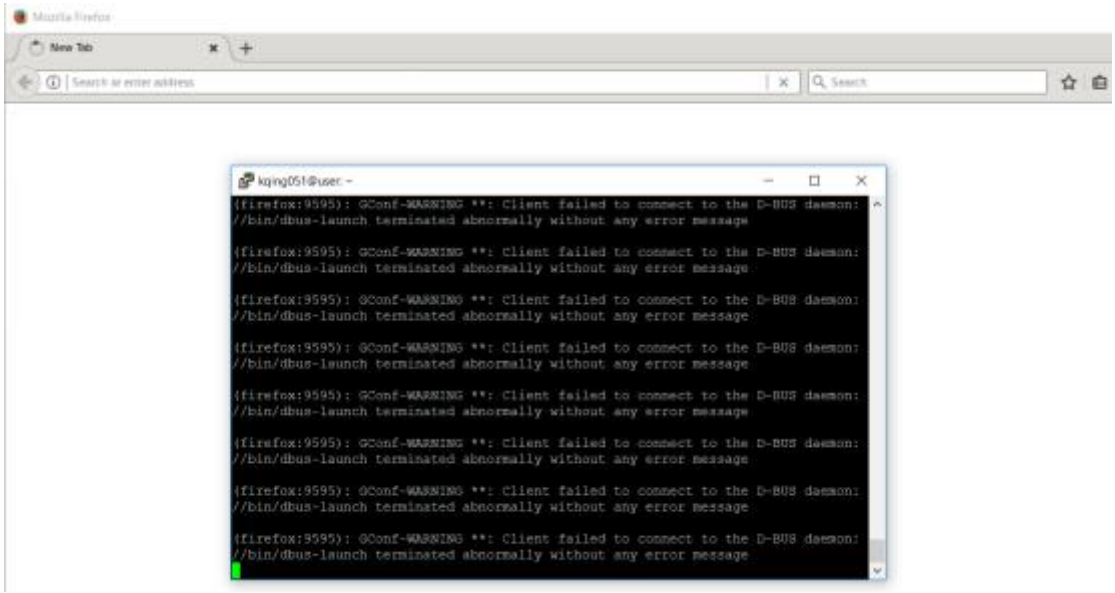


Then we can run the graphics display on GENI node on Windows operation system.

Command:" firefox"

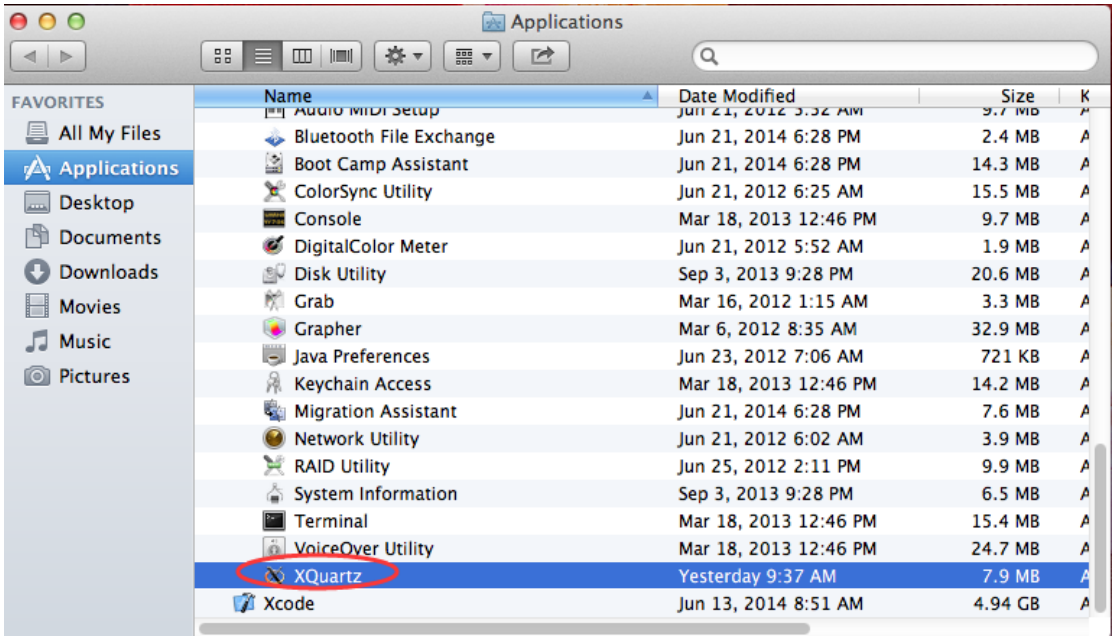
```
kqing051@user: ~
.
Setting up libgtk2.0-bin (2.24.23-0ubuntu1.4) ...
Setting up xul-ext-ubufox (3.2-0ubuntu0.14.04.1) ...
Processing triggers for libc-bin (2.19-0ubuntu6.6) ...
kqing051@user:~$ firefox

(firefox:9595): GLib-GObject-CRITICAL **: g_object_ref: assertion 'object->ref_c
ount > 0' failed

(firefox:9595): GLib-GObject-CRITICAL **: g_object_unref: assertion 'object->ref
_count > 0' failed

(firefox:9595): GLib-GObject-CRITICAL **: g_object_ref: assertion 'object->ref_c
ount > 0' failed

(firefox:9595): GLib-GObject-CRITICAL **: g_object_unref: assertion 'object->ref
_count > 0' failed

(firefox:9595): GConf-WARNING **: Client failed to connect to the D-BUS daemon:
//bin/dbus-launch terminated abnormally without any error message

(firefox:9595): GConf-WARNING **: Client failed to connect to the D-BUS daemon:
//bin/dbus-launch terminated abnormally without any error message
```

Wait for a second, and then we can see the browser GUI display is shown with the help of Xming

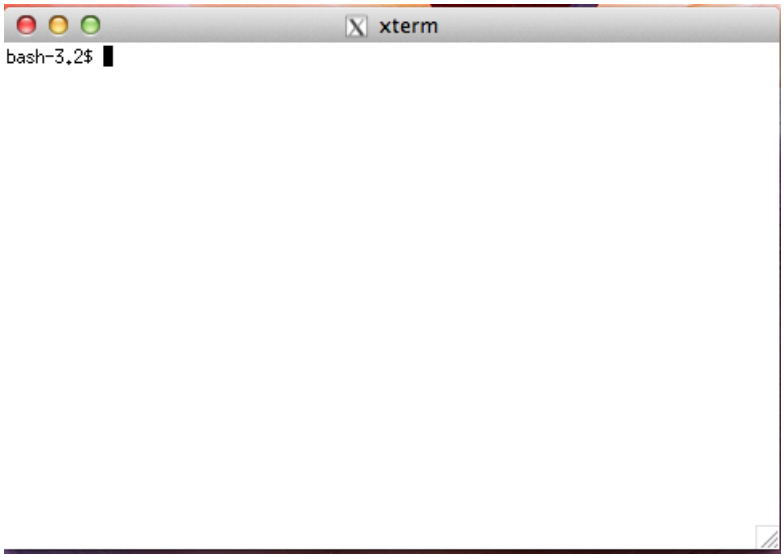


For MacOS operation system

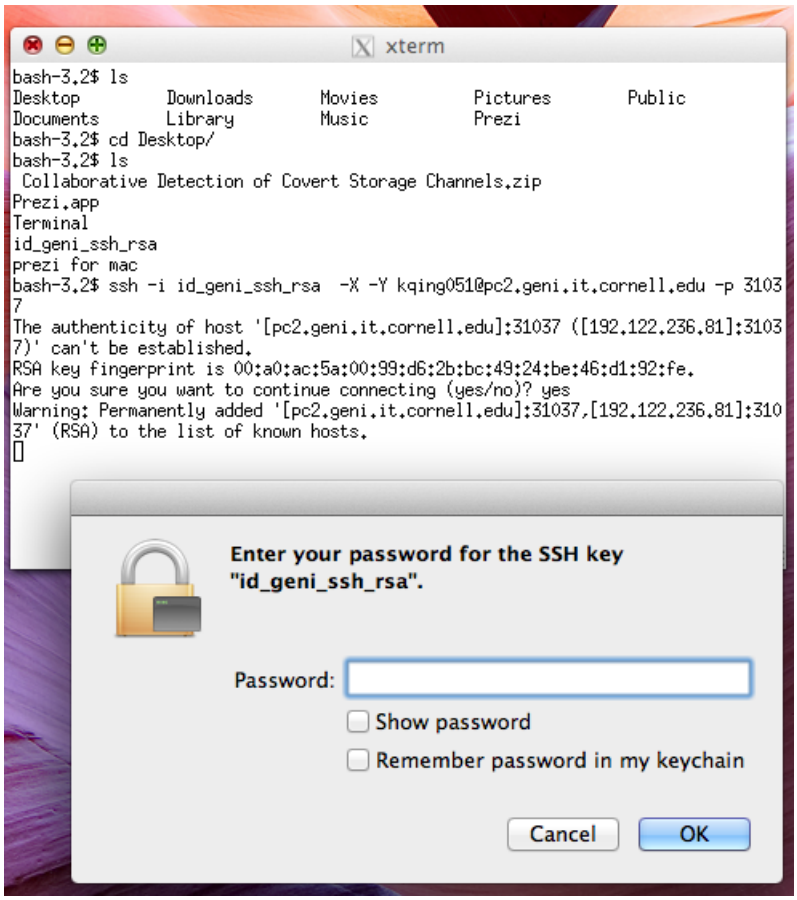
Install XQuartz on your Mac. XQuartz is an X server designed for MacOS.



Right click on the XQuartz icon in the dock and select Applications > Terminal. This should bring up a new xterm terminal windows.

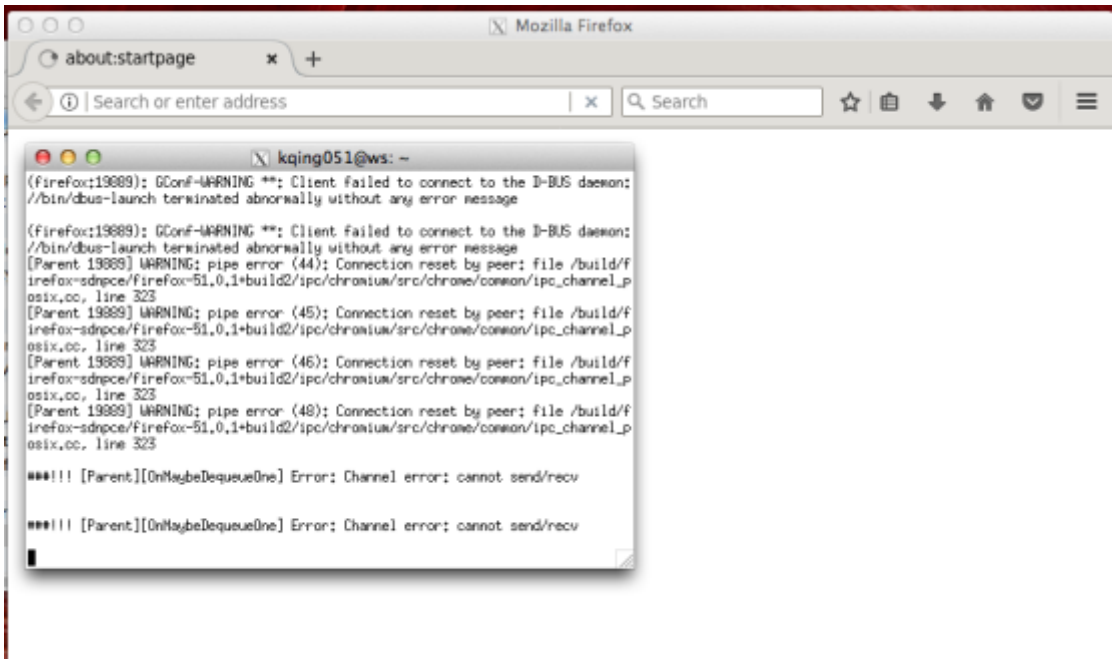


Then make an ssh connection to the GENI node on this terminal windows.



We can enable the graphics display of browser on GENI node with the help of XQuartz software.

Command:" firefox"



For other Linux operation system

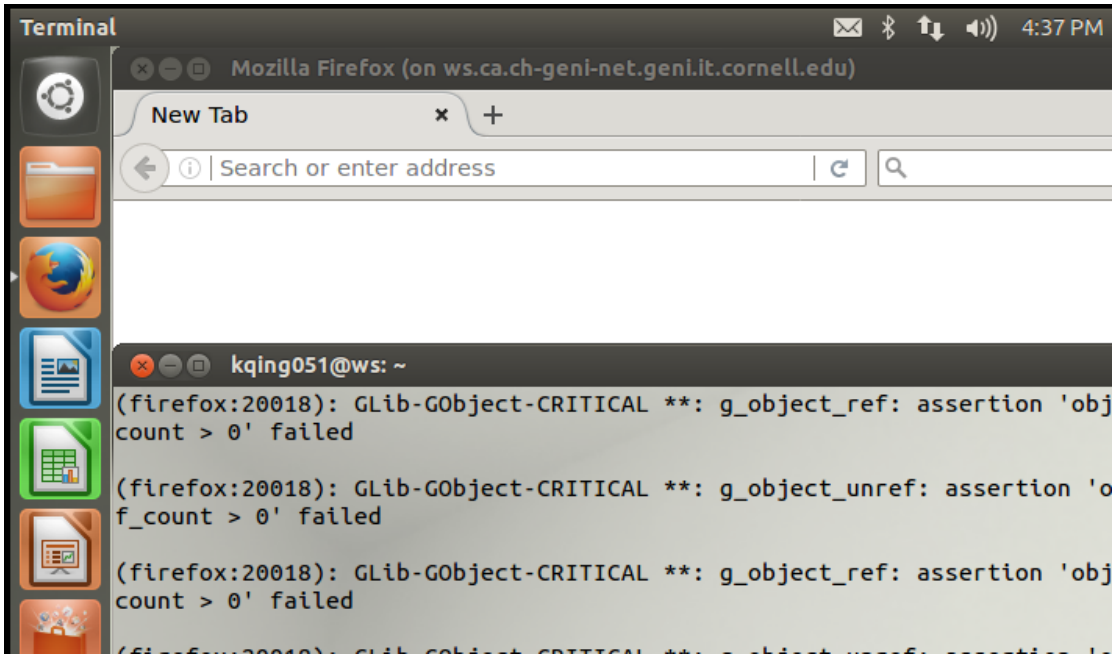
It’ s much simple when you are using Linux operation system.

Just ssh into the Linux system of your choice using the -Y argument.

```
[02/20/2017 16:35] root@ubuntu:/home/seed# ssh -i id_geni_ssh_rsa -Y kqing051@pc
2.geni.it.cornell.edu -p 31037
The authenticity of host '[pc2.geni.it.cornell.edu]:31037 ([192.122.236.81]:3103
7)' can't be established.
RSA key fingerprint is 00:a0:ac:5a:00:99:d6:2b:bc:49:24:be:46:d1:92:fe.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[pc2.geni.it.cornell.edu]:31037,[192.122.236.81]:310
37' (RSA) to the list of known hosts.
Enter passphrase for key 'id_geni_ssh_rsa':
```

Then run Firefox browser.

Command:” firefox”



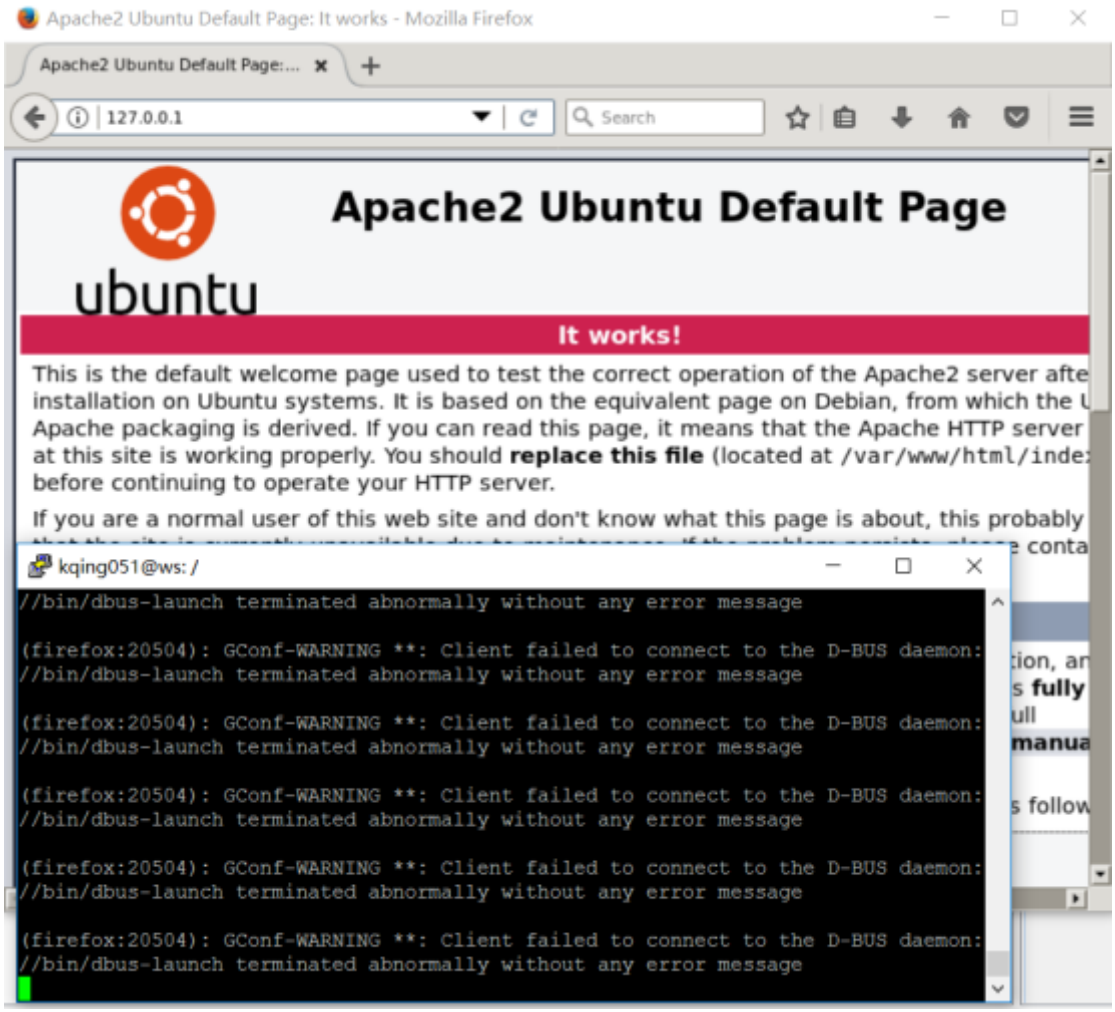
Test Apache service and PHP service

Now because we already installed and enable the browser. We can test the Apache service and PHP service installed before.

Test Apache service

Open the Firefox browser using command "firefox"

Enter "127.0.0.1" on the browser. If it shows the following image then it means the Apache service is installed successfully.

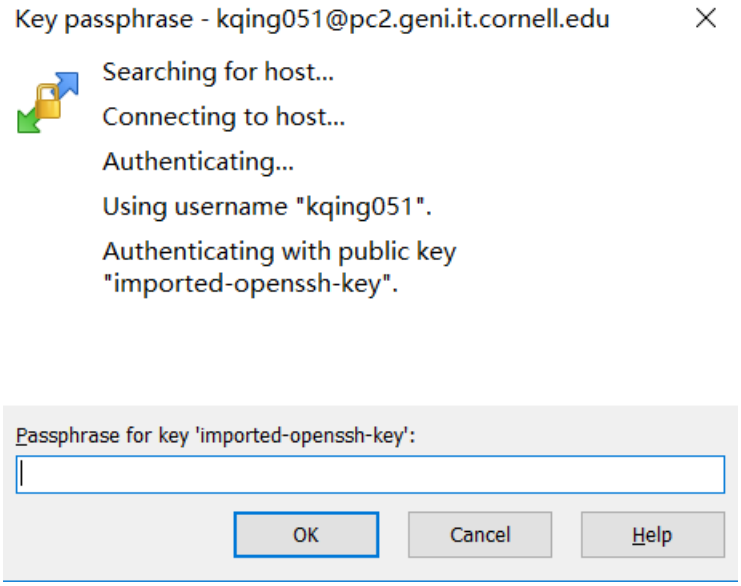


Test PHP service

We can write a simple PHP website and then run it to test whether the PHP service is installed successfully or not.

For the convenience, we can use the WinSCP software to write PHP source code. The instruction of how to log onto GENI node using WinSCP can be found in this link:

http://mountrouidoux.people.cofc.edu/CyberPaths/winscp.html



We need to give the privilege to edit the "www" folder under the /var. As I mentioned before, "www" folder contains the website source code.

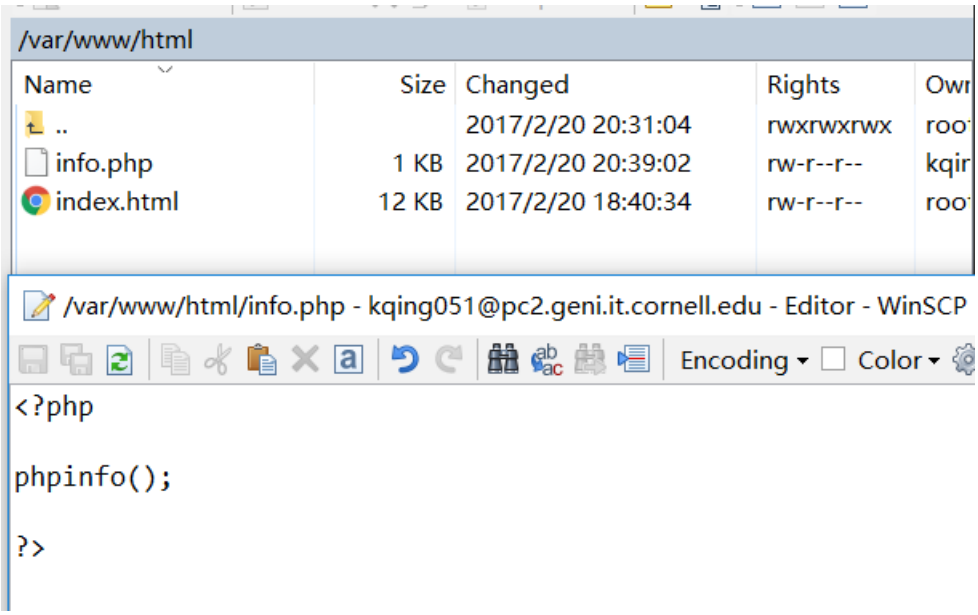
Command:" sudo chmod 777 /var/www"

```
kqing051@ws:/var/www$ sudo su
root@ws:/var/www# chmod 777 html
root@ws:/var/www#
```

Under the "www/html" folder, create a file named "info.php" and then write the following code into it:

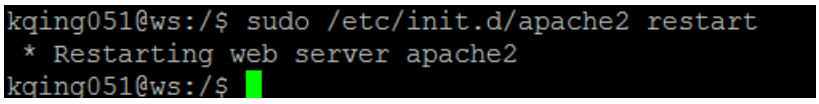
```
<?php
phpinfo();
```

?>

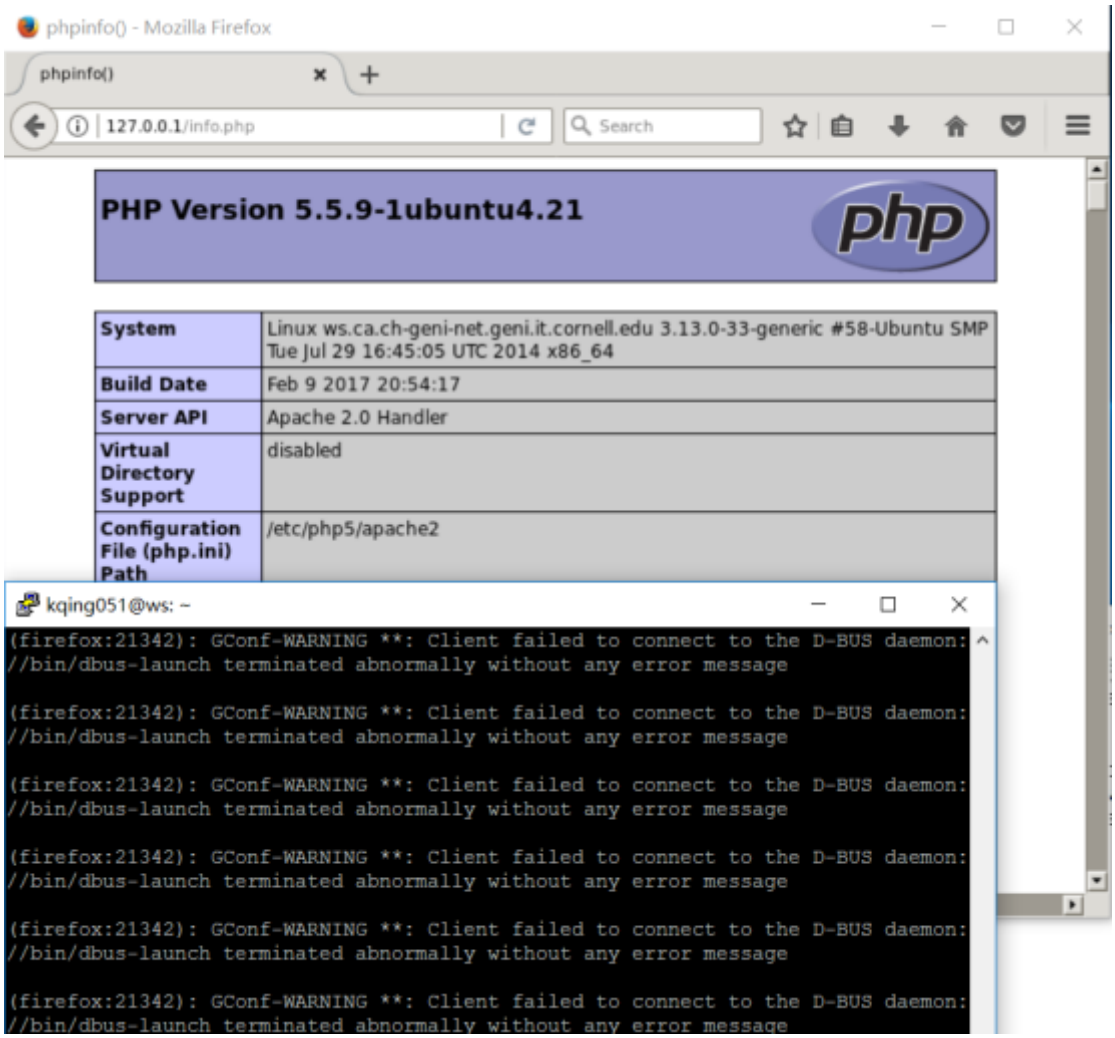


And then we need to restart the Apache service.

Command:"sudo /etc/init.d/apache2 restart"



Run the browser and this time enters "127.0.0.1/info.php" into the browser. If we can see the relative configuration information of PHP then it means that PHP service is installed successfully.



Set up certificate authority on GENI

GENI node is pre-installed OpenSSL. OpenSSL is a general purpose cryptography library that provides an open-source implementation of the SSL and TLS.

We can take a quick view of OpenSSL on GENI node. The folder is in the location of "/etc/ssl" We can see three documents already in this folder. The "certs" is a folder to store the digital certificate of this machine. "Private" is the folder to store the private key of the digital certificate. "openssl.cnf" is the main configuration document of OpenSSL.

```
kqing051@ws:/etc/ssl$ ls
certs  openssl.cnf  private
kqing051@ws:/etc/ssl$
```

Right now, the documents of enabling the whole function of OpenSSL is not enough. We need to do some configurations.

Build Certificate Authority

Firstly, we need to do something on certificate authority node, which is CA node in our topology to let it be able to play a role as a certificate authority.

Create some new documents in this folder.

```
root@ca:/etc/ssl# ls
cacert.pem  index.txt.attr  private     ws.csr
certs       index.txt.old   serial
crl         newcerts        serial.old
index.txt   openssl.cnf     ws.crt
root@ca:/etc/ssl#
```

We must create newcerts folder and index.txt. If we don’t then we will get stuck with the issue step.

Create a serial document and set the serial number.

```
kqing051@ca:/etc/ssl$ sudo touch serial
kqing051@ca:/etc/ssl$ sudo echo 01 > serial
-bash: serial: Permission denied
kqing051@ca:/etc/ssl$ sudo su
root@ca:/etc/ssl# echo 01 > serial
root@ca:/etc/ssl#
```

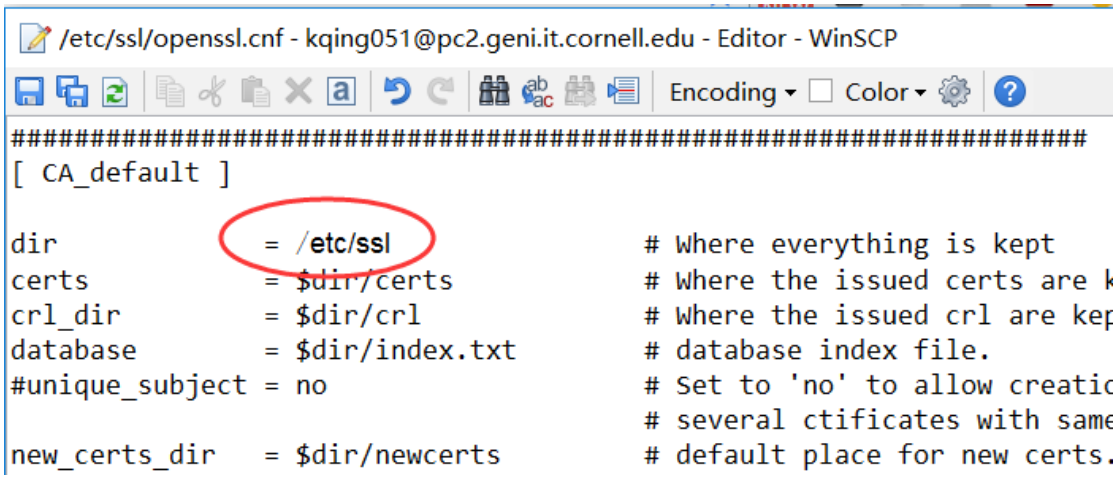
And then make changes in openssl.cnf.

To succeed it, we should give the privilege to write openssl.cnf.

Command:" sudo chmod 777 openssl.cnf"

```
kqing051@ca:/etc/ssl$ sudo chmod 777 openssl.cnf
kqing051@ca:/etc/ssl$
```

And then using WinSCP open it and set the following values as followings:



/etc/ssl/openssl.cnf - kqing051@pc2.geni.it.cornell.edu - Editor - WinSCP

```
# req_extensions = v3_req # The extensions to add to a certificate

[ req_distinguished_name ]
countryName                     = Country Name (2 letter code)
countryName_default             = US
countryName_min                 = 2
countryName_max                 = 2

stateOrProvinceName             = State or Province Name (full nam
stateOrProvinceName_default     = MD

localityName                    = Locality Name (eg, city)

0.organizationName              = Organization Name (eg, company)
0.organizationName_default      = JHU

# we can do this but it is not needed normally :-)
#1.organizationName             = Second Organization Name (eg, co
#1.organizationName_default     = World Wide Web Pty Ltd

organizationalUnitName          = Organizational Unit Name (eg, se
#organizationalUnitName_default =ISI
```

Then we need to generate root private key for the root digital certificate for the certificate authority.

Command:" openssl genrsa -out private/cakey.pem 2048"

Create root digital certificate for the certificate authority.

Command:" openssl req -new -x509 -key private/cakey.pem -out ca cert.pem"

```
You are about to be asked to enter information that will be
 incorporated
into your certificate request.
What you are about to enter is what is called a Distinguish
ed Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:
State or Province Name (full name) [MD]:
Locality Name (eg, city) []:
Organization Name (eg, company) [JHU]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:jhuca.edu
Email Address []:kqing0515@jhu.edu
root@ca:/etc/ssl#
```

This time we set the server name as" jhuca.edu" . You can set what ever as you like, we will do some configuration later so we can visit this server name as you set here.

On web server node

ın this part, we need to log onto the WS node, which we have set it to be a web server before.

Generate private key for web server

Command:" sudo openssl genrsa -out jhuws.key 2048"

```
root@ws:/etc/ssl# sudo openssl genrsa -out jhuws.key 2048
Generating RSA private key, 2048 bit long modulus
.................+++
......+++
e is 65537 (0x10001)
```

Generate certificate sign request for the web server.

Command:" sudo openssl req -new -key jhuws.key -out jhuws.csr"

```
root@ws:/etc/ssl# sudo openssl req -new -key jhuws.key -out
 jhuws.csr
You are about to be asked to enter information that will be
 incorporated
into your certificate request.
What you are about to enter is what is called a Distinguish
ed Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:
State or Province Name (full name) [MD]:
Locality Name (eg, city) [Baltimore]:
Organization Name (eg, company) [JHU]:
Organizational Unit Name (eg, section) [ISI]:
Common Name (e.g. server FQDN or YOUR name) []:jhuws.edu
Email Address []:kqing0515@jhu.edu

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:admin
An optional company name []:
root@ws:/etc/ssl#
```

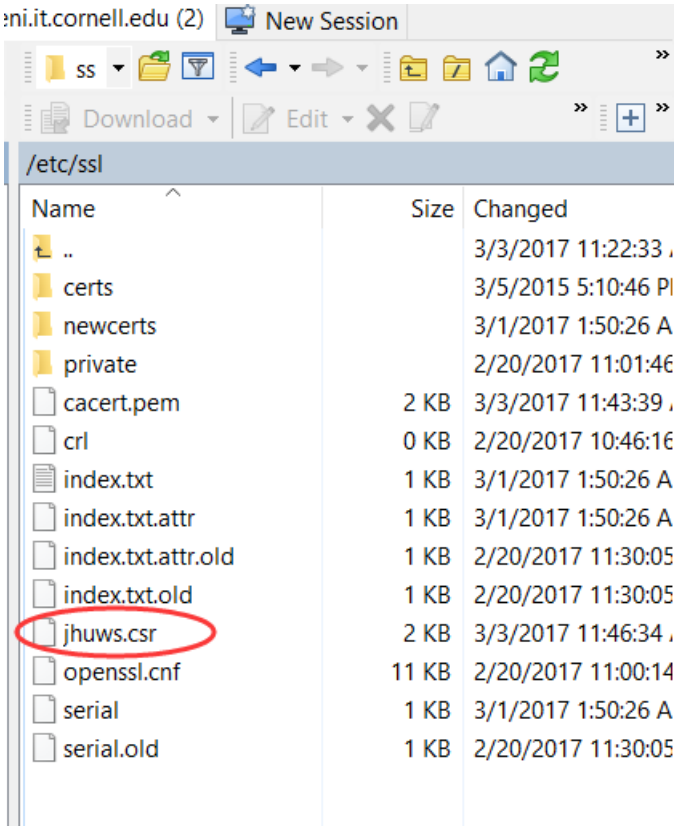
Issue a digital certificate

In this step, CA node needs to get the request sign document .csr from the web server. We can use WinSCP or SCP command to make the transportation. If we want to transport it using WinSCP, we can firstly use command chmod to give out the privilege and then transport it. It' s the

same in the later using of the WinSCP.

Command:" chmod 777 file_name"

In this case, it should be:" chmod 777 jhuws.csr"




```
root@ca:/etc/ssl# ls
cacert.pem  index.txt.attr      newcerts     serial.old
certs       index.txt.attr.old  openssl.cnf
crl         index.txt.old       private
index.txt   jhuws.csr           serial
root@ca:/etc/ssl#
```

Then we can sign this digital certificate on CA node.

Command:" openssl ca -in /etc/ssl/jhuws.csr -out /etc/ssl/jhuws.crt -days 3650"

```
root@ca:/etc/ssl# openssl  ca  -in  /etc/ssl/jhuws.csr  -ou
t  /etc/ssl/jhuws.crt  -days  3650
Using configuration from /usr/lib/ssl/openssl.cnf
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 3 (0x3)
        Validity
            Not Before: Mar  3 17:11:40 2017 GMT
            Not After : Mar  1 17:11:40 2027 GMT
            stateOrProvinceName       = MD
            organizationName          = JHU
            organizationalUnitName    = ISI
            commonName                = jhuws.edu
            emailAddress              = kqing0515@jhu.edu
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                07:04:F4:DB:62:9D:64:5B:C4:F5:7A:78:68:BD:E
E:BC:96:ED:CD:1C
            X509v3 Authority Key Identifier:
                keyid:BE:23:89:56:ED:A6:B2:9D:D3:70:C1:EA:E
6:92:77:32:82:C5:06:A9
```

```
            Not After : Mar  1 17:11:40 2027 GMT

            stateOrProvinceName       = MD
            organizationName          = JHU
            organizationalUnitName    = ISI
            commonName                = jhuws.edu
            emailAddress              = kqing0515@jhu.edu
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                07:04:F4:DB:62:9D:64:5B:C4:F5:7A:78:68:BD:E
E:BC:96:ED:CD:1C
            X509v3 Authority Key Identifier:
                keyid:BE:23:89:56:ED:A6:B2:9D:D3:70:C1:EA:E
6:92:77:32:82:C5:06:A9

Certificate is to be certified until Mar  1 17:11:40 2027 G
MT (3650 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
root@ca:/etc/ssl#
```

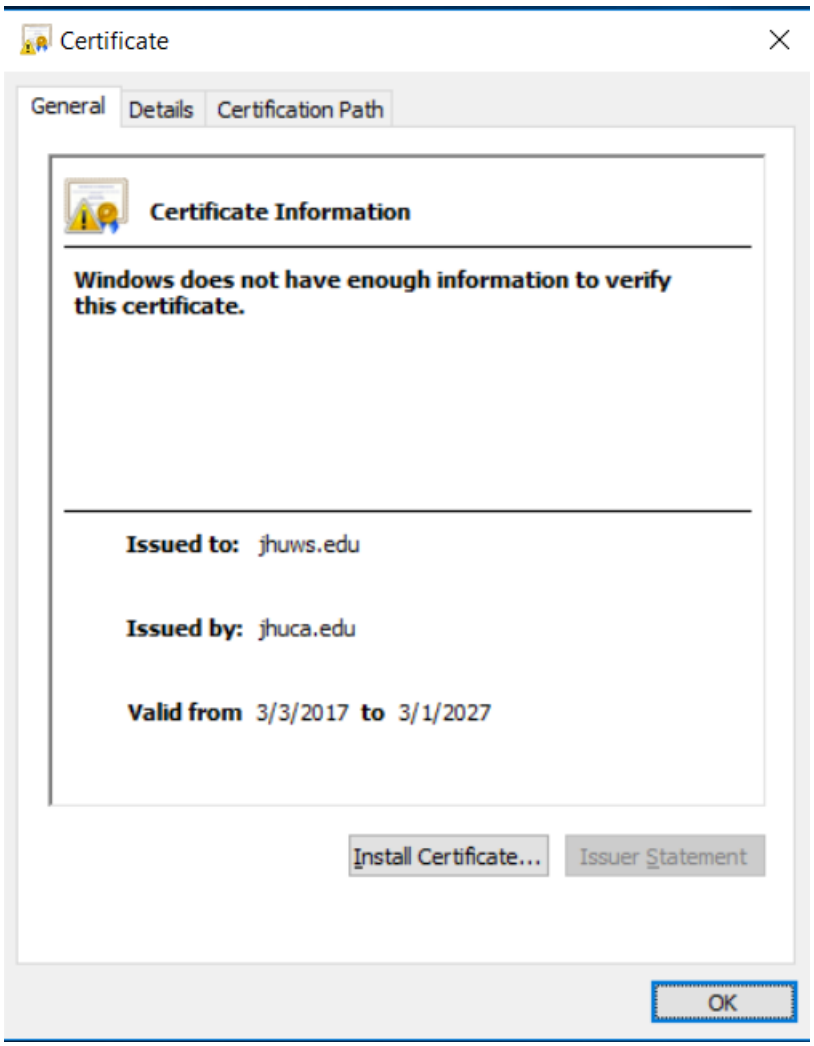
Then we send this digital certificate back to the web server, which is the ws node. This still can be done by WinSCP.

```
root@ws:/etc/ssl# ls
certs      jhuws.crt  jhuws.key    private
httpd.key  jhuws.csr  openssl.cnf
root@ws:/etc/ssl#
```

We now can see this digital certificate.

We can use the cat command to view it on Linux, or we can just double click and open it on Windows.

```
root@ws:/etc/ssl# cat jhuws.crt
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 3 (0x3)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, ST=MD, O=JHU, CN=jhuca.edu/emailAddre
ss=kqing0515@jhu.edu
        Validity
            Not Before: Mar  3 17:11:40 2017 GMT
            Not After : Mar  1 17:11:40 2027 GMT
        Subject: C=US, ST=MD, O=JHU, OU=ISI, CN=jhuws.edu/e
mailAddress=kqing0515@jhu.edu
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:d0:b9:79:57:20:4e:55:cf:e9:0f:31:b9:
b5:b7:
                    fb:a4:68:52:a3:82:26:d0:e8:e6:b9:dd:54:
d7:38:
                    57:03:12:ff:b5:45:f0:33:02:70:d8:2e:2b:
```



And we can see clearly that this digital certificate is issued by jhuca.edu, which is the certificate authority node and is issued to jhuws.edu, which is the web server in our topology, the ws node.

Use these two commands to install the web server' s digital certificate. In other words, we should put the digital certificate file and key fileinto the correct folder of the web server.

Command:" sudo cp jhuws.crt /etc/ssl/certs"

Command:" sudo cp jhuws.key /etc/ssl/private"

```
root@ws:/etc/ssl# sudo cp jhuws.crt /etc/ssl/certs
root@ws:/etc/ssl# sudo cp jhuws.key /etc/ssl/private
root@ws:/etc/ssl#
```

Now before we open the browser to see the result, we should copy the cacert.pem from ca node to user node, we need to let the ca to verify the digital certificate issued by ca, so we need this file on the web server.

And we should change the extension ".pem" to ".crt" .

If you meet with the alarm of privilege when transport the file using WinSCP, just use command "chmod 777 file_name" to give out the privilege.

```
root@user:/etc# chmod 777 ssl
root@user:/etc# cd ssl
root@user:/etc/ssl# ls
cacert.crt  certs  openssl.cnf  private
root@user:/etc/ssl#
```

Result of issued digital certificate

Now we can view the result of the former work we have done.

The following operations are all on user node.

Firstly log onto the user node.

Because the display of browser on GENI is kind of slow. TO quick test whether we are right by far, we can install the “curl” to perform a quick test.

Curl is a tool to transfer data from or to a server, using one of the supported protocols.

Use “sudo su” to enter the root account

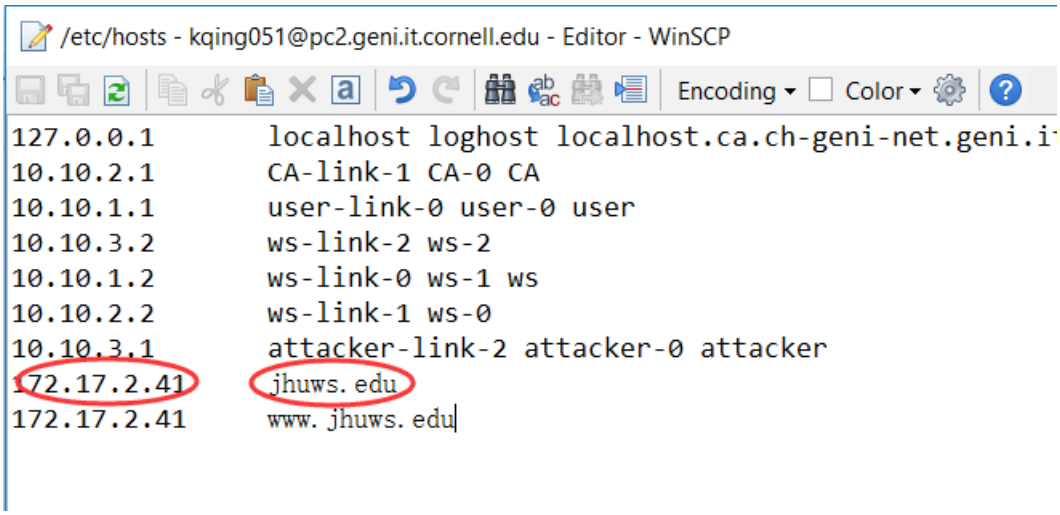
Use “apt-get update” to update the existed packets.

Use “apt-get install curl” to install the curl.

Then move to the hosts file to do a little modification.

```
root@user:/etc# cat hosts
127.0.0.1       localhost loghost localhost.ca.ch-geni-net.
geni.it.cornell.edu
10.10.2.1       CA-link-1 CA-0 CA
10.10.1.1       user-link-0 user-0 user
10.10.3.2       ws-link-2 ws-2
10.10.1.2       ws-link-0 ws-1 ws
10.10.2.2       ws-link-1 ws-0
10.10.3.1       attacker-link-2 attacker-0 attacker
root@user:/etc#
```

Use command ” chmod 777 hosts” to give out the privilege.



We do these modifications on it. While the “172.17.2.41” is the IP address of ws node, this is the web server in out topology. And jhuws.edu is the server name we set up by ourselves before.

Use command “curl jhuws.edu” to see if it works right.

```
root@user:/etc# curl jhuws.edu

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//
EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dt
d">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2014-03-19
    See: https://launchpad.net/bugs/1288690
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; cha
rset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
```

It returns the HTML response of our web server. It seems it works well.

Then we test whether SSL configuration we modified before is work right.

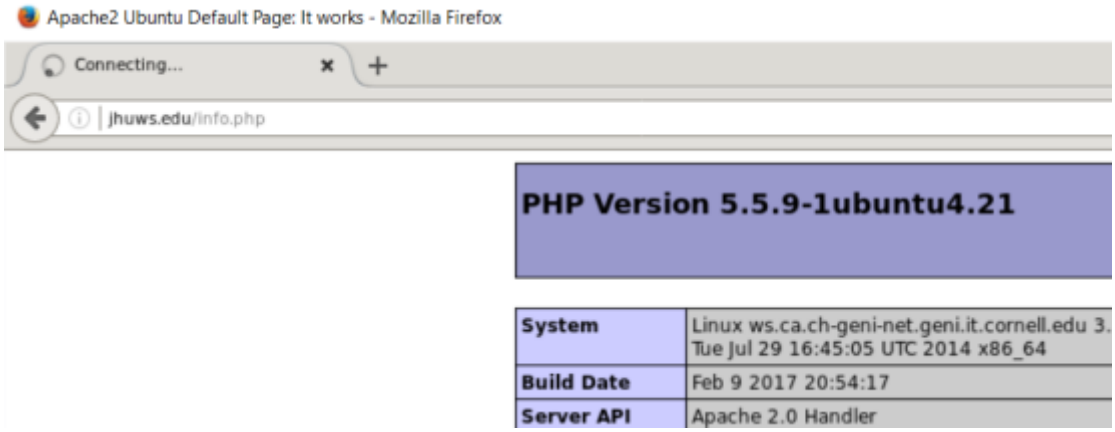
We use this command “curl https://jhuws.edu --cacert cacert.crt” to test.

```
root@user:/etc/ssl# curl https://jhuws.edu --cacert cacert.
crt
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
 <head>
  <title>Index of /</title>
 </head>
 <body>
<h1>Index of /</h1>
  <table>
   <tr><th valign="top"><img src="/icons/blank.gif" alt="[I
CO]"></th><th><a href="?C=N;O=D">Name</a></th><th><a href="
?C=M;O=A">Last modified</a></th><th><a href="?C=S;O=A">Size
</a></th><th><a href="?C=D;O=A">Description</a></th></tr>
   <tr><th colspan="5"><hr></th></tr>
<tr><td valign="top"><img src="/icons/folder.gif" alt="[DIR
```

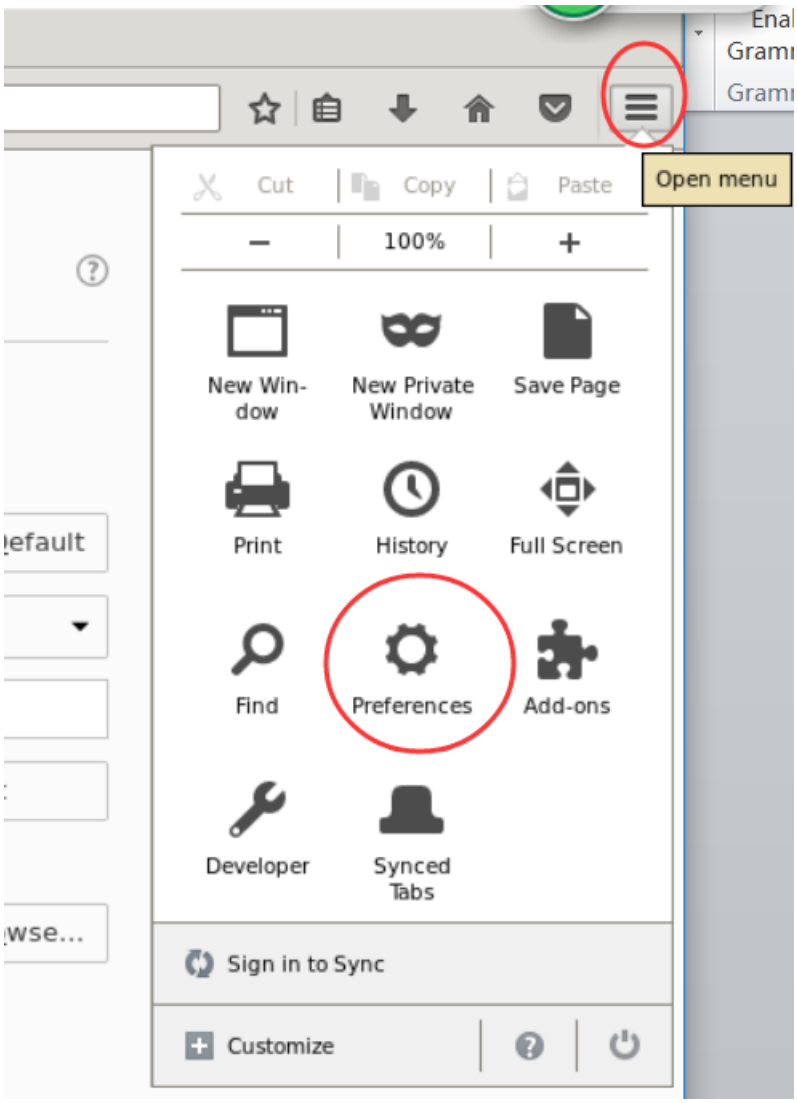
It also works well. So we can turn to the browser.

Open the browser on user node. You can refer the instruction before about how to run and open a browser on GENI node.

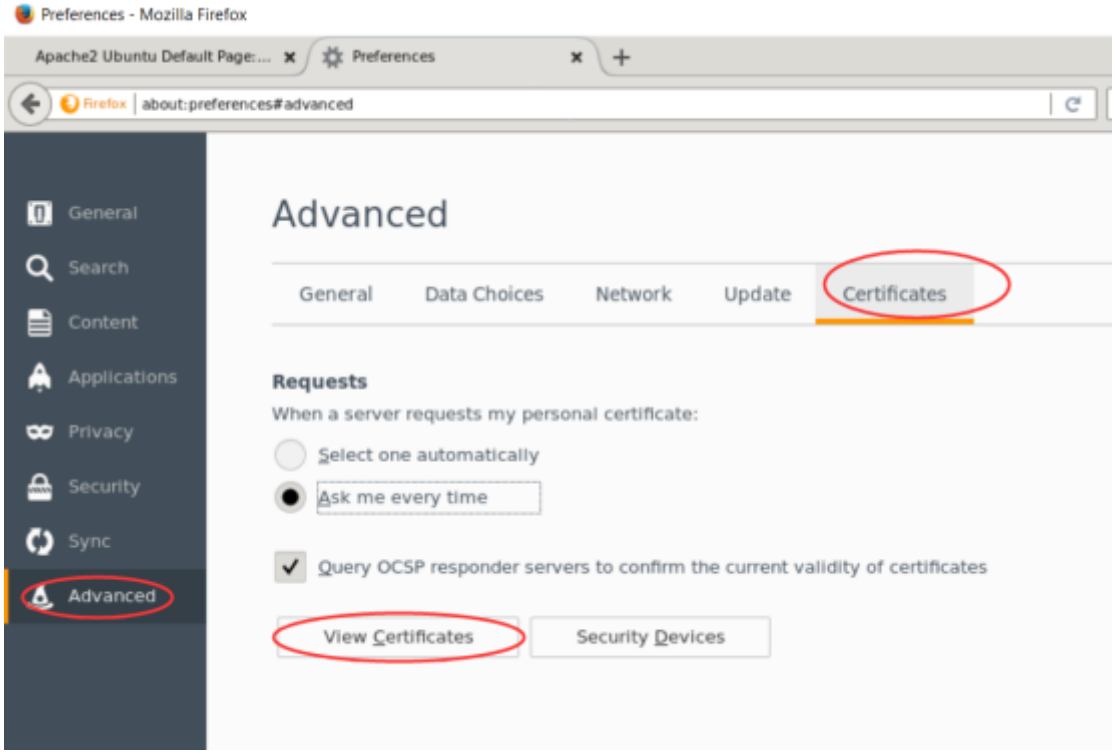
We visit the jhuws.edu/info.php first.



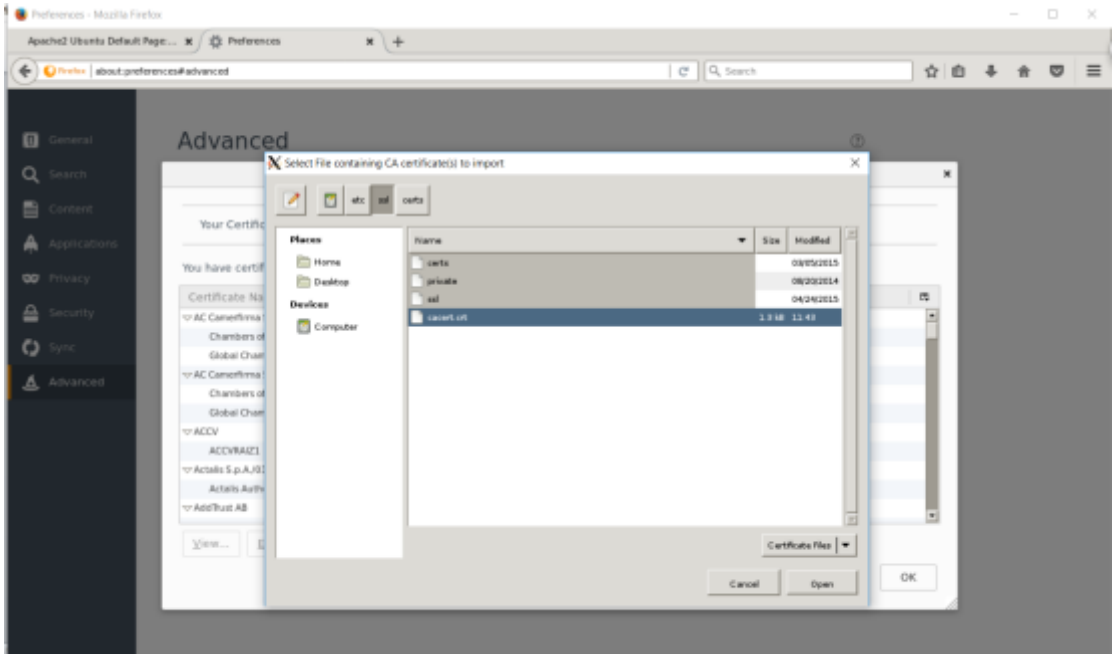
It works right, and then we install the digital certificate. We go to the preference option of the browser.



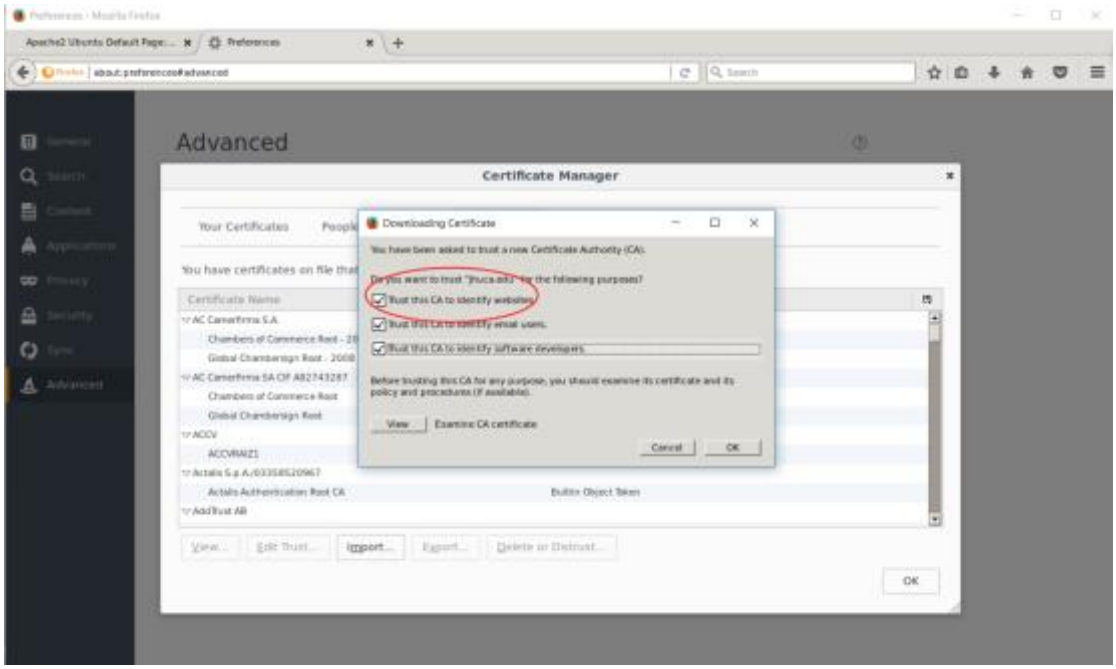
Then go to the Advanced, Certificates options. Click on "View Certificates"



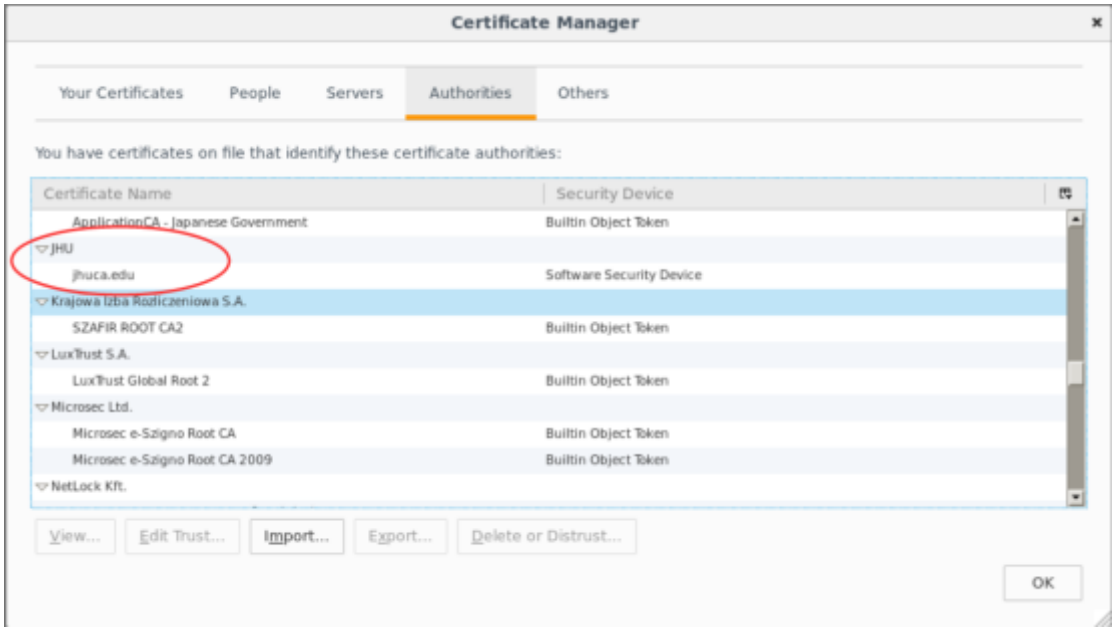
Then import the cacert.crt file we put on user node before.



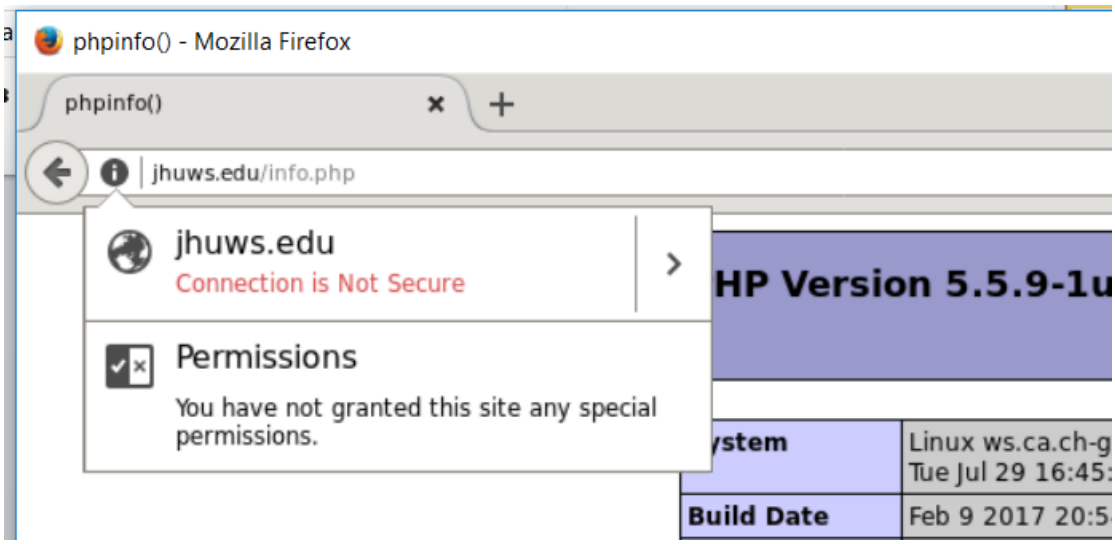
Trust this certificate authority we build by our own.



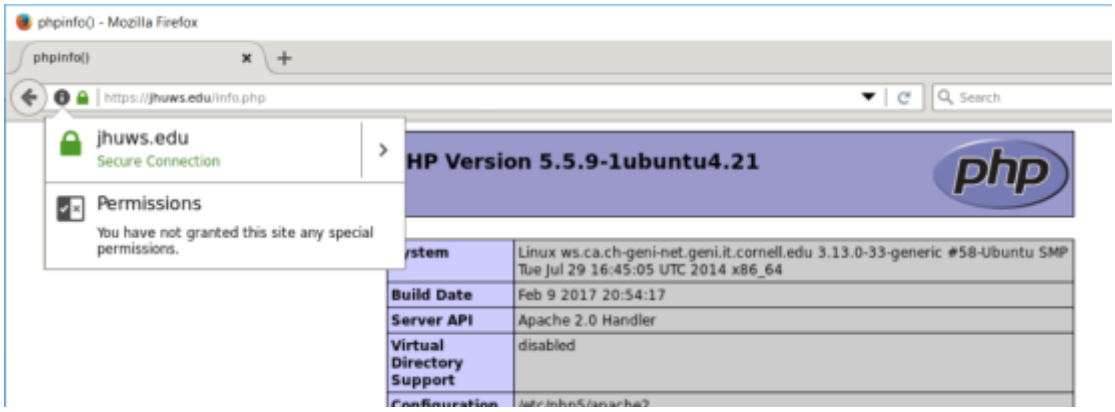
Now we can see our certificate authority is in the list of certificate authorities.



We visit jhuws.edu/info.php firstly, we can see the connection is unsecure.

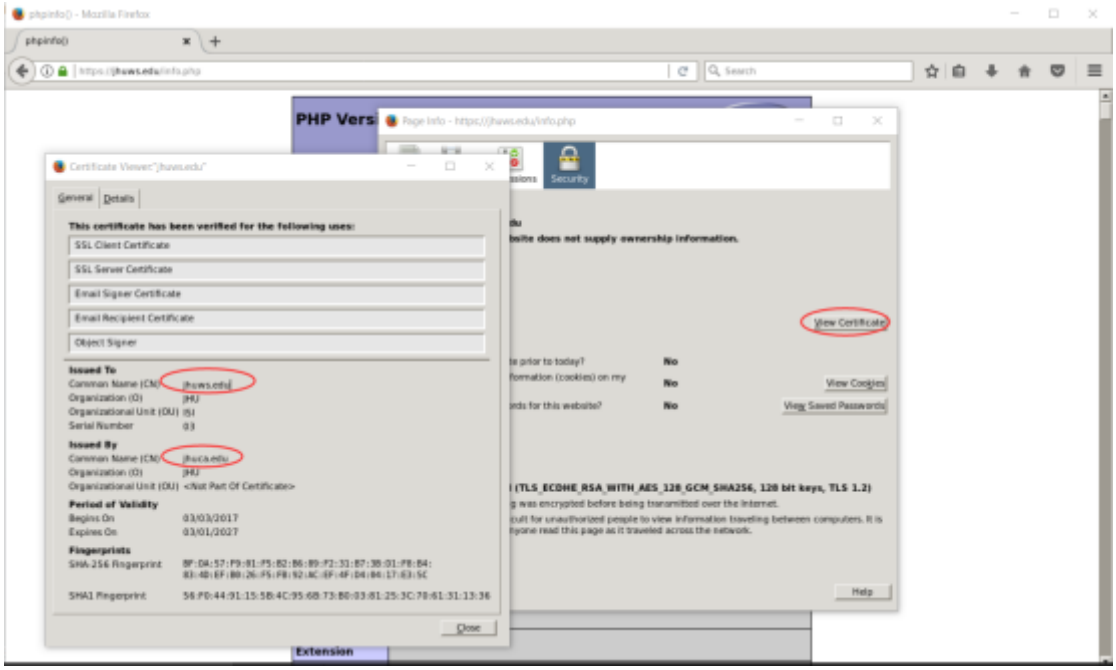


Then we visit https://jhuws.edu/info.php



And this time we can see there appears a green lock and remind us now it’s a secure connection.

We can view the details of this connection by click the “>”.



The information we put in when making the digital certificate is shown in this display.

By now, we have already finished the experiment of building a certificate authority and issue a digital certificate.

Revoke a digital certificate

Then we can try to revoke the digital certificate to see how it works and what is the result it will turn out.

To revoke the digital certificate we issue before, we should log onto the ca node, which is the certificate authority.

Use this command to revoke the digital certificate we issued before. The name of digital certificate we issued before is “jhuws.crt”

Command : “openssl ca -revoke jhuws.crt”

```
root@ca:/etc/ssl# openssl ca -revoke jhuws.crt
Using configuration from /usr/lib/ssl/openssl.cnf
Revoking Certificate 03.
Data Base Updated
root@ca:/etc/ssl#
```

We can use this command" cat index.txt" to check the index of our digital certificate. It' s 03, so everything is going right.

```
root@ca:/etc/ssl# cat index.txt
V       270219042923Z           01      unknown /C=US/ST=MD
/O=JHU/OU=ISI/CN=sis.jhu.edu/emailAddress=kqing0515@jhu.edu
V       270227065017Z           02      unknown /C=US/ST=MD
/O=JHU/OU=ISI/CN=www.a.org/emailAddress=www@a.org
R       270301171140Z   170303185717Z   03      unknown /C=
US/ST=MD/O=JHU/OU=ISI/CN=jhuws.edu/emailAddress=kqing0515@j
hu.edu
root@ca:/etc/ssl#
```

And we should renew the crl, which is short for Certificate Revocation List. The digital certificate we revoked will be given the index and record in this list.

Use this command:" openssl ca -gencrl -out thisca.crl" to generate the index of the digital certificate we just revoke.

We can have a look at the thisca.crl file.

```
root@ca:/etc/ssl# cat thisca.crl
-----BEGIN X509 CRL-----
MIIBzTCBtgIBATANBgkqhkiG9w0BAQsFADBeMQswCQYDVQQGEwJVUzELMAk
GA1UE
CAwCTUQxDDAKBgNVBAoMA0pIVTESMBAGA1UEAwwJamh1Y2EuZWR1MSAwHgY
JKoZI
hvcNAQkBFhFrcWluZzA1MTVAamh1LmVkdRcNMTcwMzAzMjAyMDU2WhcNMTc
wNDAy
MjAyMDU2WjAUMBICAQMXDTE3MDMwMzE4NTcxN1qgDjAMMAoGA1UdFAQDAgE
BMA0G
CSqGSIb3DQEBCwUAA4IBAQBn9NtW3cCcNJwZ9d8Gc0BThE0FajYHwz62WZm
+0oPM
```

It stores the Certificate Revocation List.

Then we need add content ” 00” into ” crlnumber” , to let the crl start to record.

Each time we revoke a digital certificate, it will add one to itself.

```
root@ca:/etc/ssl# cat crlnumber
02
root@ca:/etc/ssl#
```

I have revoked twice the digital certificate, so the number shown is 02.

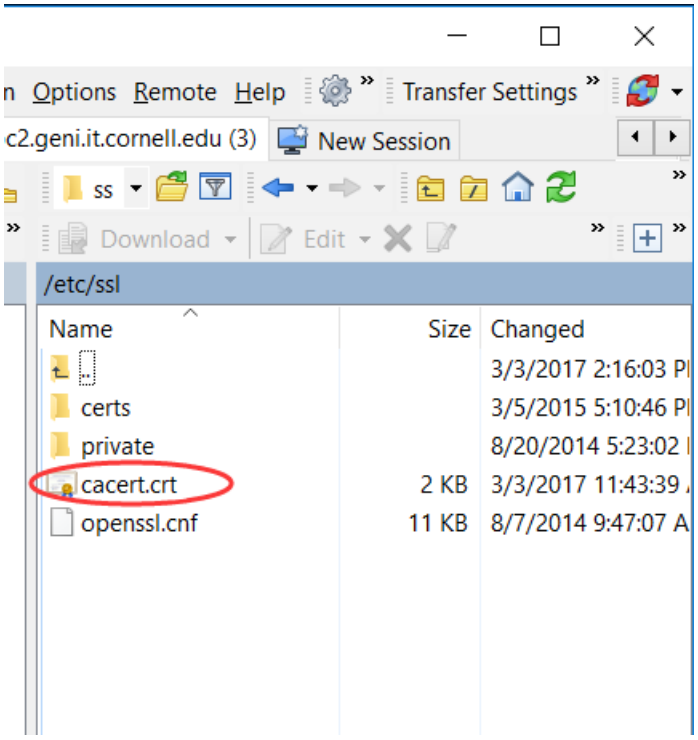
Then we have successfully revoked a digital certificate we issued before.

Result of revoke a digital certificate

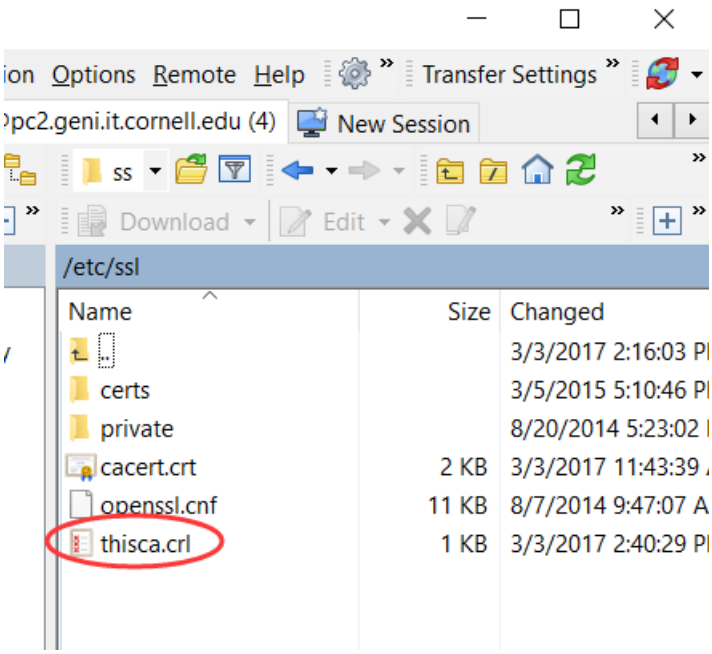
To see the result after we revoke an issued digital certificate. Because we are actually not using network to verify. Therefore we need to import the new cacert.crt onto the user node.

Remove the old cacert.crt and import the new cacert.crt. As the same we have done before, we actually move the file “cacert.pem” , but we need to change the extension” .pem” to “.crt” .

```
root@user:/etc/ssl# ls
cacert.crt  certs  openssl.cnf  private
root@user:/etc/ssl# rm cacert.crt
root@user:/etc/ssl#
```



Also we should move the thisca.crl which stores the Certificate Revocation List to the user node.



Because our certificate authority is offline, therefore we need to move these files manually.

Then we can see the result after we revoke a digital certificate. We still use curl tool to find out.

We use this command:" curl https://jhuws.edu --cacert cacert.crt --crlfile thisca.crl" to visit our site we build before. We need the options" —cacert" and "—crlfile" to include the ca document and crl document manually.

```
root@user:/etc/ssl# curl https://jhuws.edu --cacert cacert.
crt --crlfile thisca.crl
curl: (60) SSL certificate problem: certificate revoked
More details here: http://curl.haxx.se/docs/sslcerts.html

curl performs SSL certificate verification by default, usin
g a "bundle"
 of Certificate Authority (CA) public keys (CA certs). If t
he default
 bundle file isn't adequate, you can specify an alternate f
ile
 using the --cacert option.
If this HTTPS server uses a certificate signed by a CA repr
esented in
 the bundle, the certificate verification probably failed d
ue to a
 problem with the certificate (it might be expired, or the
name might
```

Now we can see the digital certificate has already been revoked.

As for the Firefox browser we installed before, we also need to update the ca document and crl document. But Firefox browser has already remove the user interface on Firefox to import the crl document, so we can' t see the result on Firefox since we are the offline certificate authority and need us to import the ca document and crl document manually.

https://wiki.mozilla.org/CA:ImprovingRevocation#Preload_Revocations_of_Intermediate_CA_Certificates

Remove CRL User-Interface

As of Firefox 24, the user-interface for importing CRLs via Firefox has been removed. Auto-importing/updating of CRLs through Firefox has also been removed. NSS still supports CRLs, but Firefox is moving away from checking CRLs, and moving towards using a revocation list push mechanism.

- Release: Mozilla 24