

Creating Keys Using Java Keytool

Prepared by: Roy Geoghegan
Undergraduate Research Assistant
Department of Computer Science
Jackson State University, Jackson, MS

Faculty Mentor: Dr. Natarajan Meghanathan

Java Keytool

The Java keytool application manages a database of keys (keystore) and certificates.

Users can create their own self-authenticated certificates and public/private key pairs.

Java Keytool

This module will walk you through:

- Creating a pair of keys (public/private)
- Creating a self-signed certificate with the private key
- Importing a public key from a certificate

Java Keytool

In order to complete this module, you must have the Java Runtime Environment (JRE) installed on your computer.

The JRE is available for download at the following address:

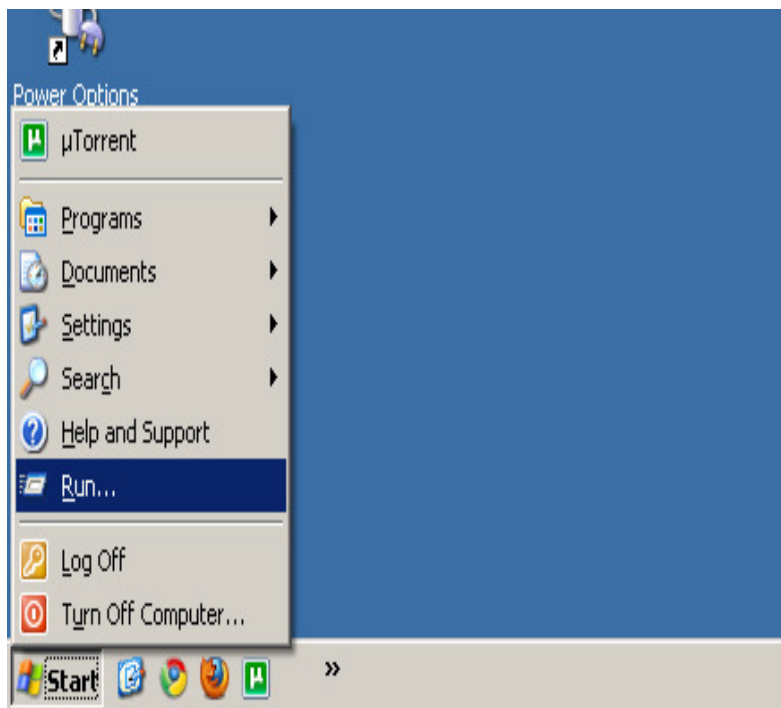
<http://www.java.com/en/download/manual.jsp>

Java Keytool

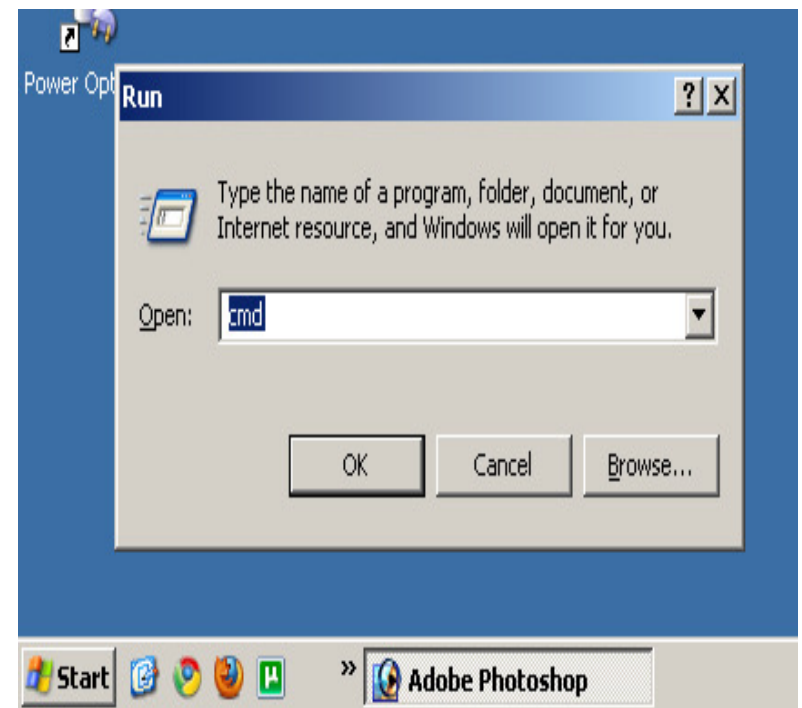
The example will be done in both the Windows environment and the Linux (Ubuntu 10.04) environment.

Windows

Click Start -> Run, type “cmd”, and press Enter.



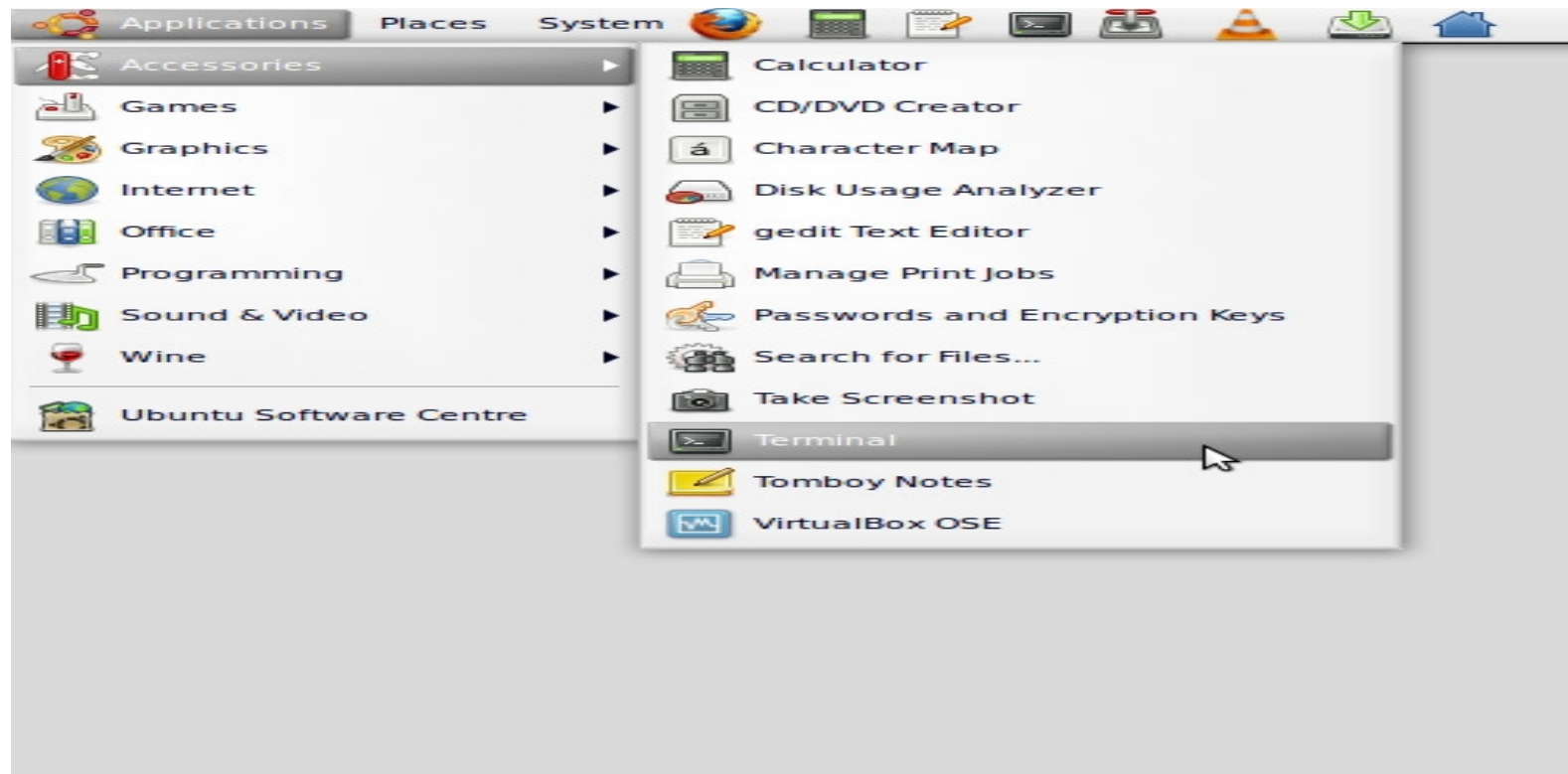
(1)



(2)

Linux

Open a new Terminal window.



Create A Pair Of Keys

In the open terminal/command window, type:

```
keytool -genkey -alias %alias% -keystore %keystore%
```

replacing *%alias%* with a name for your pair of keys and *%keystore%* with a name for your keystore, then press Enter.

In this example, *%alias%* is `KeyPair_1`, and *%keystore%* is `MyPrivateKey.store`

Create A Pair Of Keys

First you will be asked to enter a password for your keystore. Enter an easy password you will remember for the rest of this module. The password must be at least 6 characters long.

You will then be asked a number of questions regarding your identity, which will be included in the certificate.

Enter any information you like during this part.

Create A Pair Of Keys

You will be asked to provide:

- Name
- Organizational Unit
- Organization
- City
- State
- Country

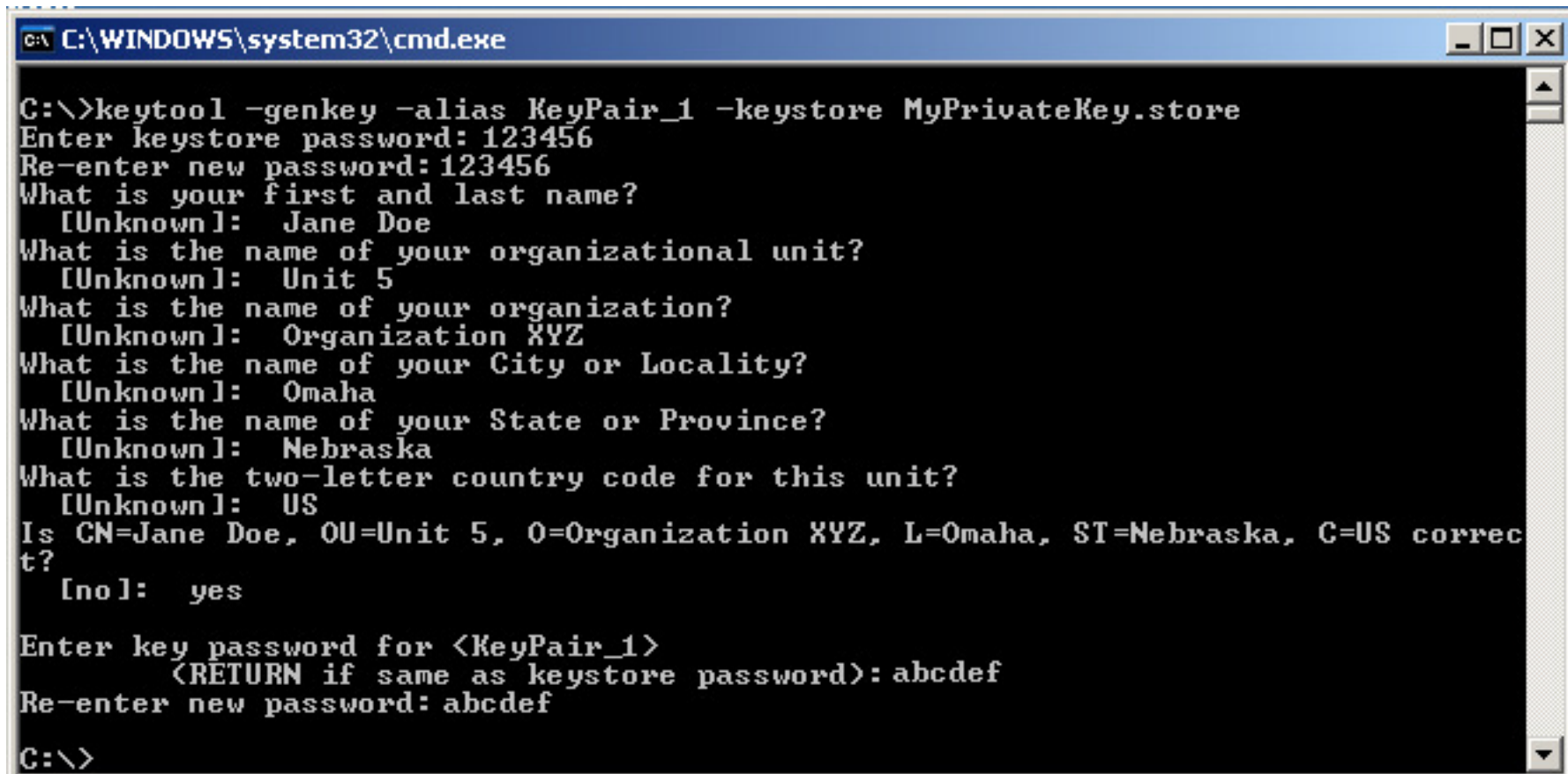
You will then be asked to verify that the information you entered is correct. Type “yes” and press enter.

Create A Pair Of Keys

Once you have verified your information, you will be asked for a password for your alias. Again, enter an easy password you will remember (at least 6 characters).

You will need to verify this password by entering it a second time.

Windows

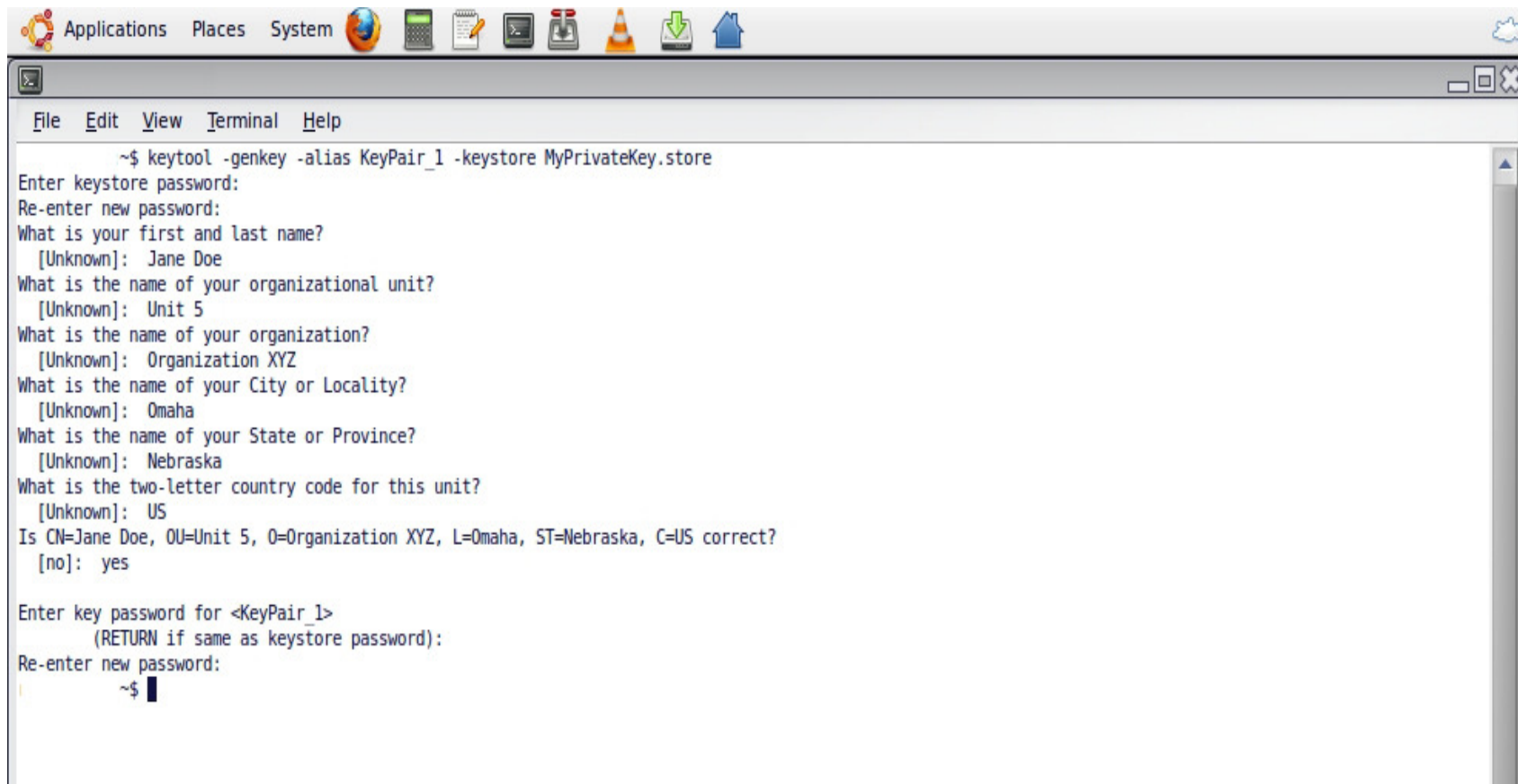


```
C:\>keytool -genkey -alias KeyPair_1 -keystore MyPrivateKey.store
Enter keystore password: 123456
Re-enter new password: 123456
What is your first and last name?
  [Unknown]:  Jane Doe
What is the name of your organizational unit?
  [Unknown]:  Unit 5
What is the name of your organization?
  [Unknown]:  Organization XYZ
What is the name of your City or Locality?
  [Unknown]:  Omaha
What is the name of your State or Province?
  [Unknown]:  Nebraska
What is the two-letter country code for this unit?
  [Unknown]:  US
Is CN=Jane Doe, OU=Unit 5, O=Organization XYZ, L=Omaha, ST=Nebraska, C=US correct?
  [no]:  yes

Enter key password for <KeyPair_1>
  (RETURN if same as keystore password): abcdef
Re-enter new password: abcdef

C:\>
```

Linux



The screenshot shows a Linux desktop environment with a top panel containing icons for Applications, Places, System, and various utilities. A terminal window is open, displaying the execution of the `keytool -genkey` command. The terminal prompts the user for a keystore password, a new password, and organizational details. The user provides the following information: Jane Doe, Unit 5, Organization XYZ, Omaha, Nebraska, and US. The terminal confirms the details and prompts for a key password, which is entered as the same as the keystore password.

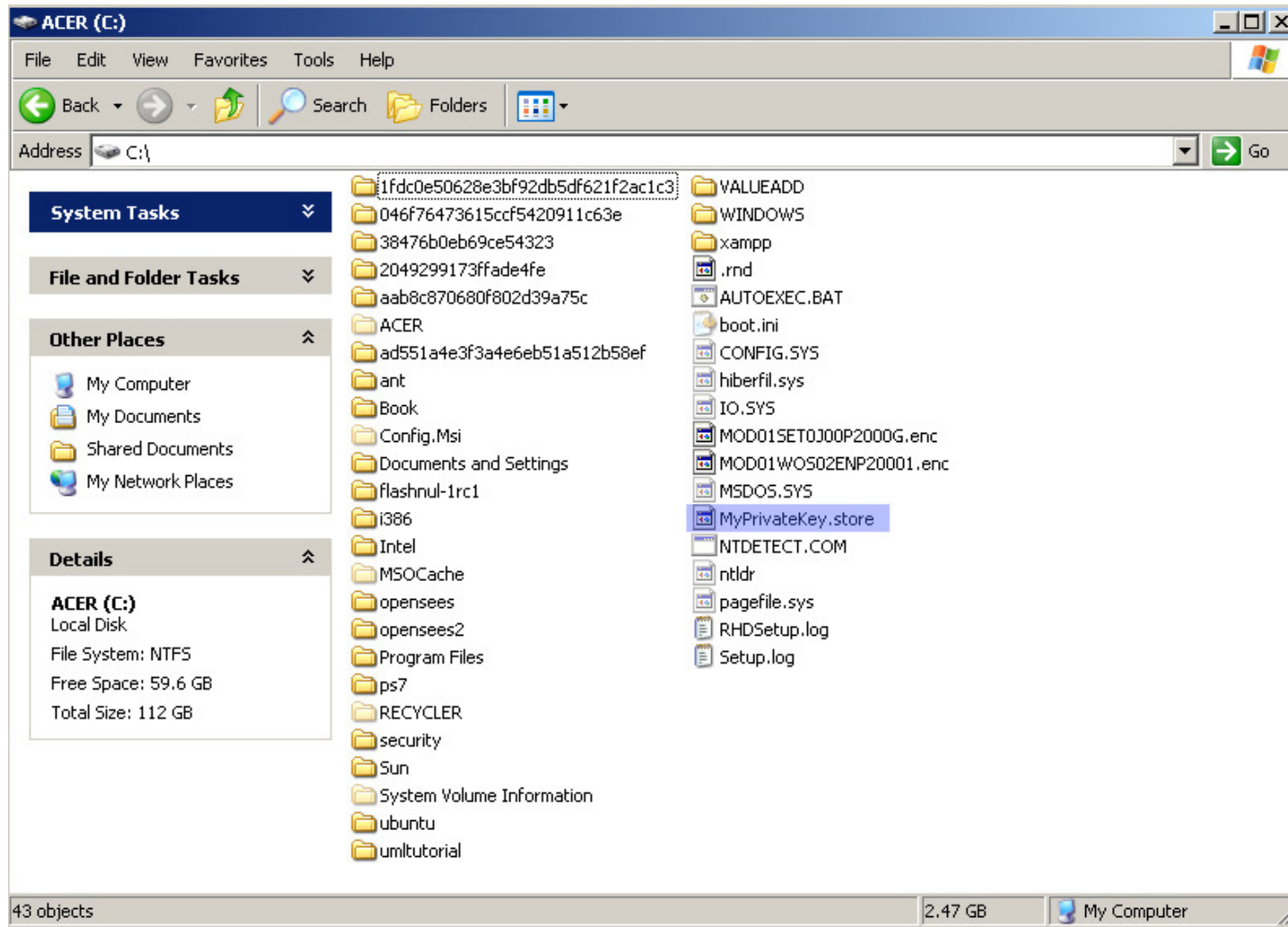
```
~$ keytool -genkey -alias KeyPair_1 -keystore MyPrivateKey.store
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Jane Doe
What is the name of your organizational unit?
[Unknown]: Unit 5
What is the name of your organization?
[Unknown]: Organization XYZ
What is the name of your City or Locality?
[Unknown]: Omaha
What is the name of your State or Province?
[Unknown]: Nebraska
What is the two-letter country code for this unit?
[Unknown]: US
Is CN=Jane Doe, OU=Unit 5, O=Organization XYZ, L=Omaha, ST=Nebraska, C=US correct?
[no]: yes

Enter key password for <KeyPair_1>
(RETURN if same as keystore password):
Re-enter new password:
~$
```

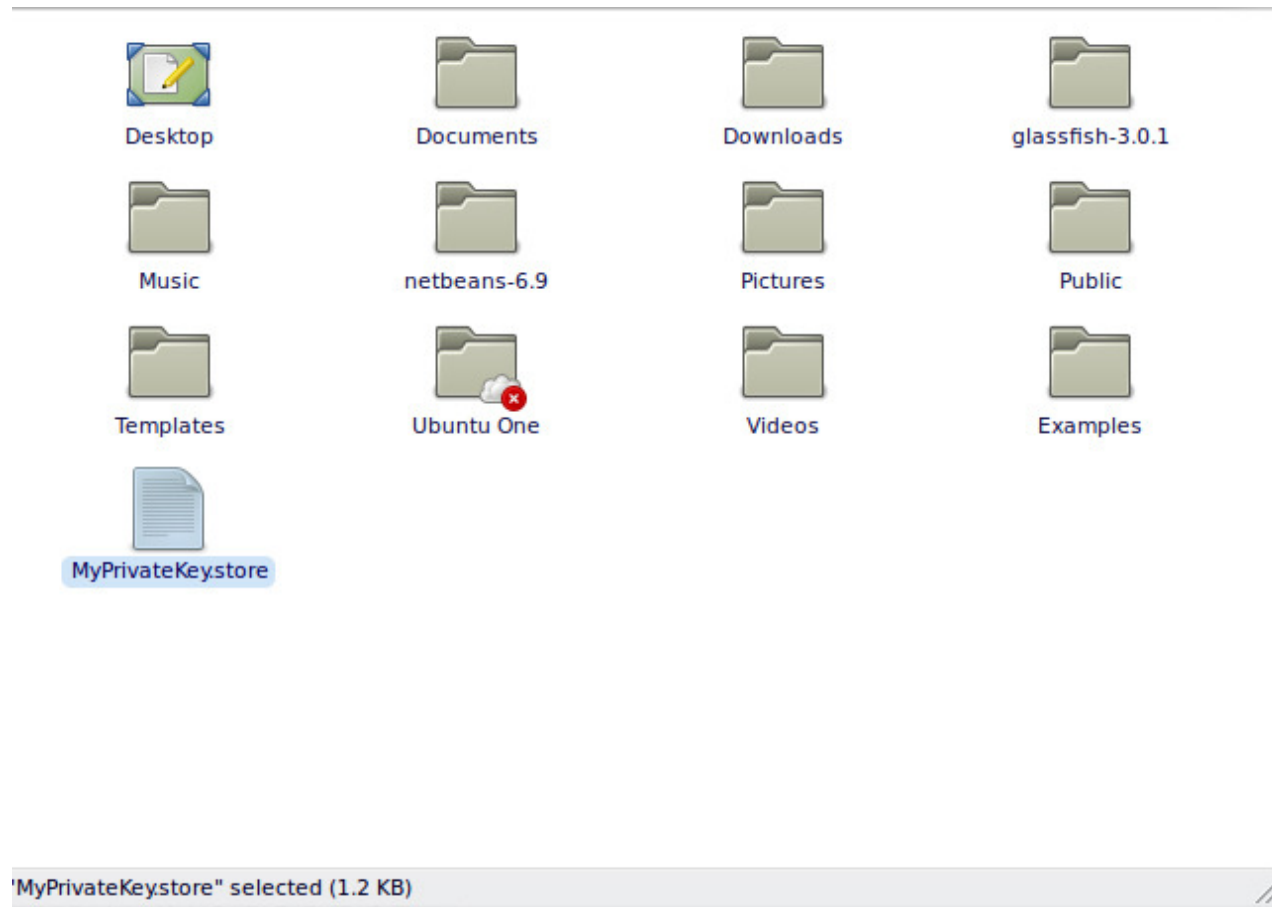
Create A Pair Of Keys

Once you have completed these steps, your keystore will be saved on the computer. It may be saved in your Documents folder, /home folder, or on the root drive.

Windows



Linux



Export A Certificate

In this step, we will export a certificate to a file.

In the terminal/command window, type:

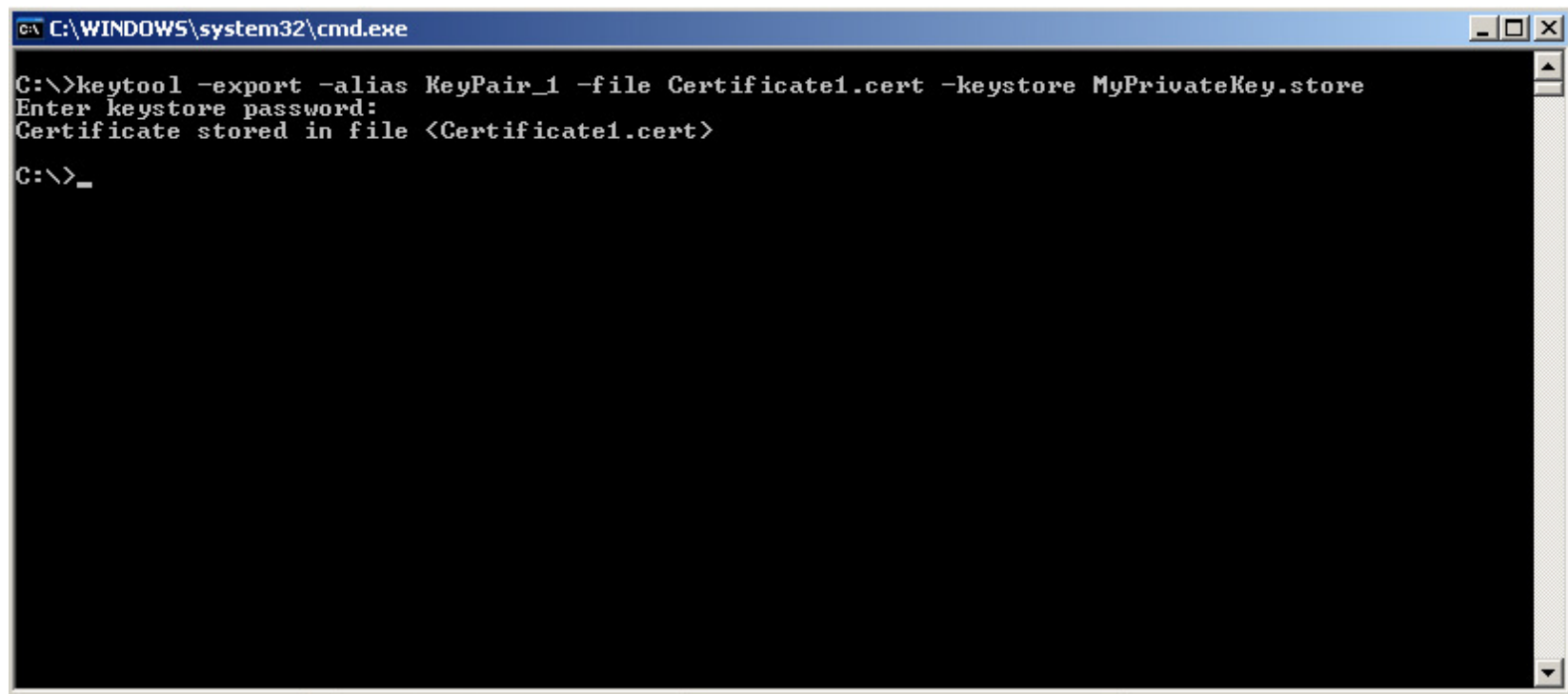
```
keytool -export -alias %alias% -file %file% -keystore %keystore%
```

replacing *%alias%* and *%keystore%* with the values you used previously, and *%file%* with the name you would like to save your certificate as.

In this example, *%file%* is Certificate1.cert, but you should name yours something different

You will also be asked for the keystore password.

Windows



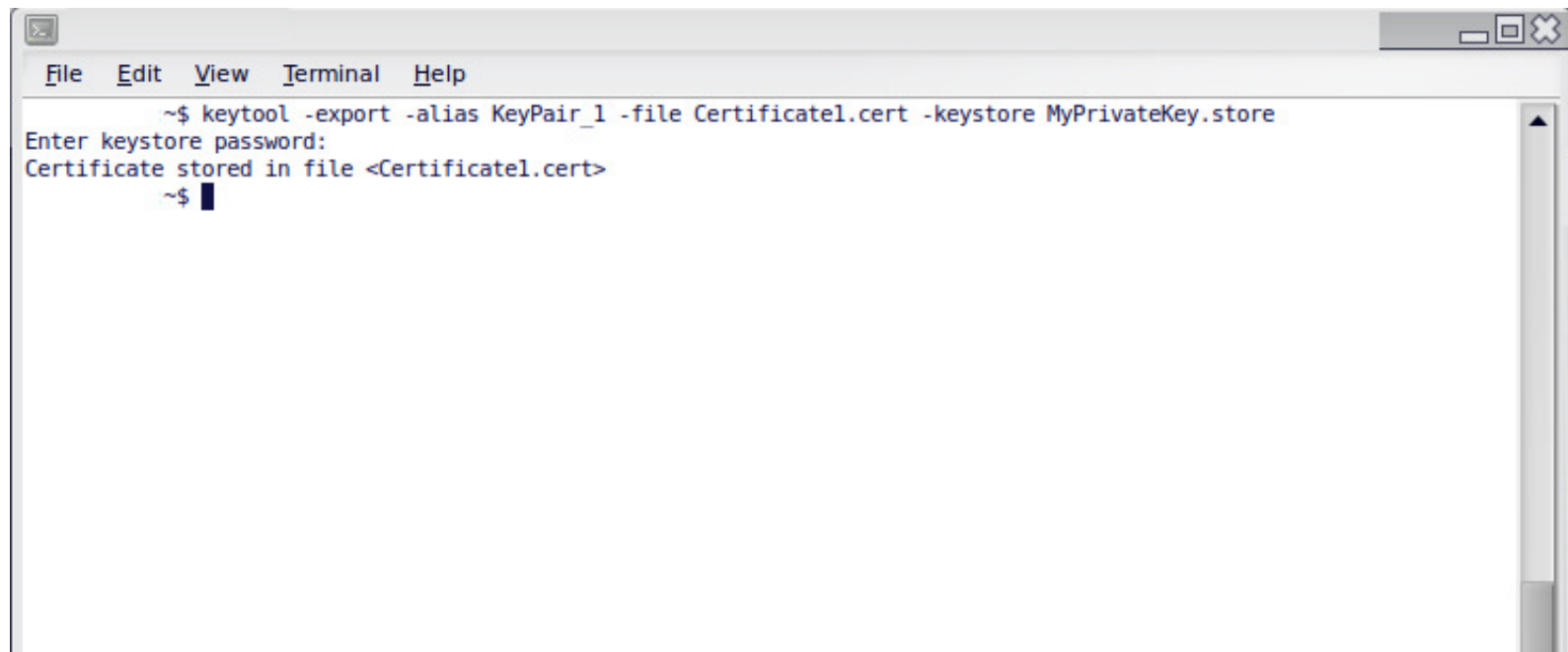
```
C:\WINDOWS\system32\cmd.exe

C:\>keytool -export -alias KeyPair_1 -file Certificate1.cert -keystore MyPrivateKey.store
Enter keystore password:
Certificate stored in file <Certificate1.cert>

C:\>_
```

The image shows a Windows command prompt window with a blue title bar. The title bar text is "C:\WINDOWS\system32\cmd.exe". The command prompt shows the execution of the command "keytool -export -alias KeyPair_1 -file Certificate1.cert -keystore MyPrivateKey.store". It prompts for the keystore password, which is entered (indicated by asterisks), and then displays the message "Certificate stored in file <Certificate1.cert>". The prompt "C:\>_" is shown at the bottom.

Linux



```
~$ keytool -export -alias KeyPair_1 -file Certificat1.cert -keystore MyPrivateKey.store
Enter keystore password:
Certificate stored in file <Certificat1.cert>
~$
```

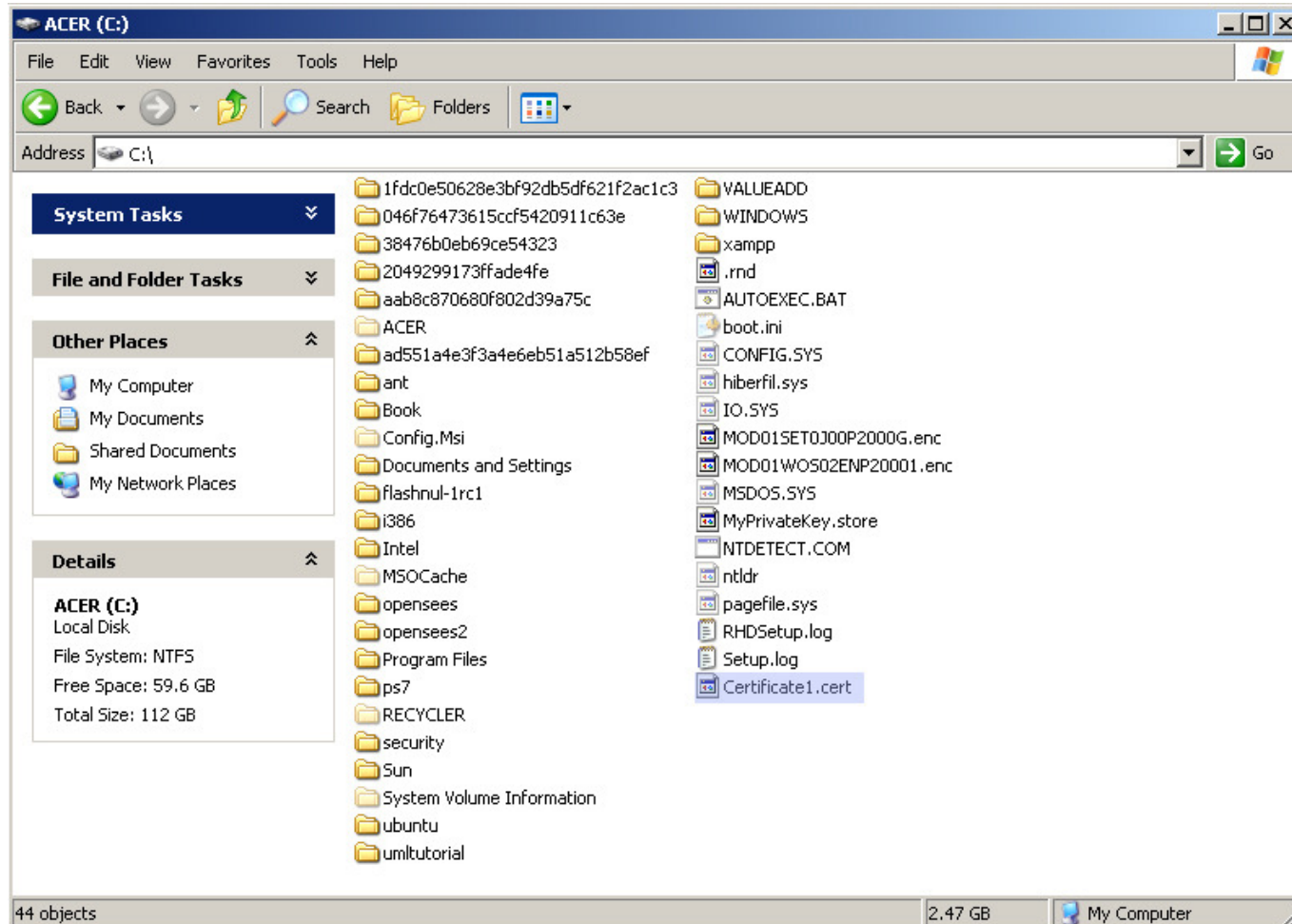
The image shows a terminal window with a menu bar (File, Edit, View, Terminal, Help) and standard window controls. The terminal displays a command to export a certificate, followed by a password prompt and a confirmation message. The prompt is followed by a cursor.

Export A Certificate

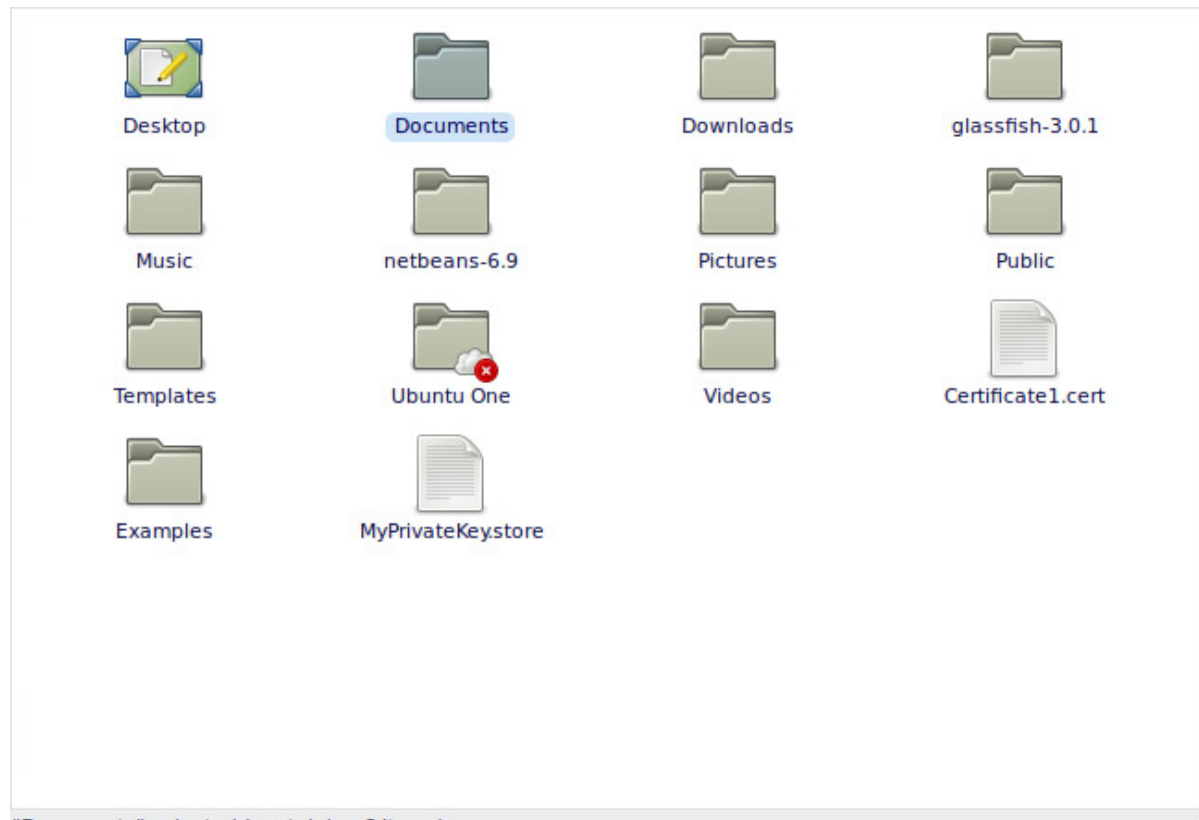
The certificate should not be saved to your computer.

It will be saved in the same location as the keystore (usually your Documents or /home folder or the root drive).

Export A Certificate



Linux



Import A Certificate

Suppose now that you have been sent a certificate via some secure method.

We will discover how to import a certificate into a keystore.

Import A Certificate

In the terminal/command window, type:

```
keytool -import -alias %alias% -file %file% -keystore %keystore%
```

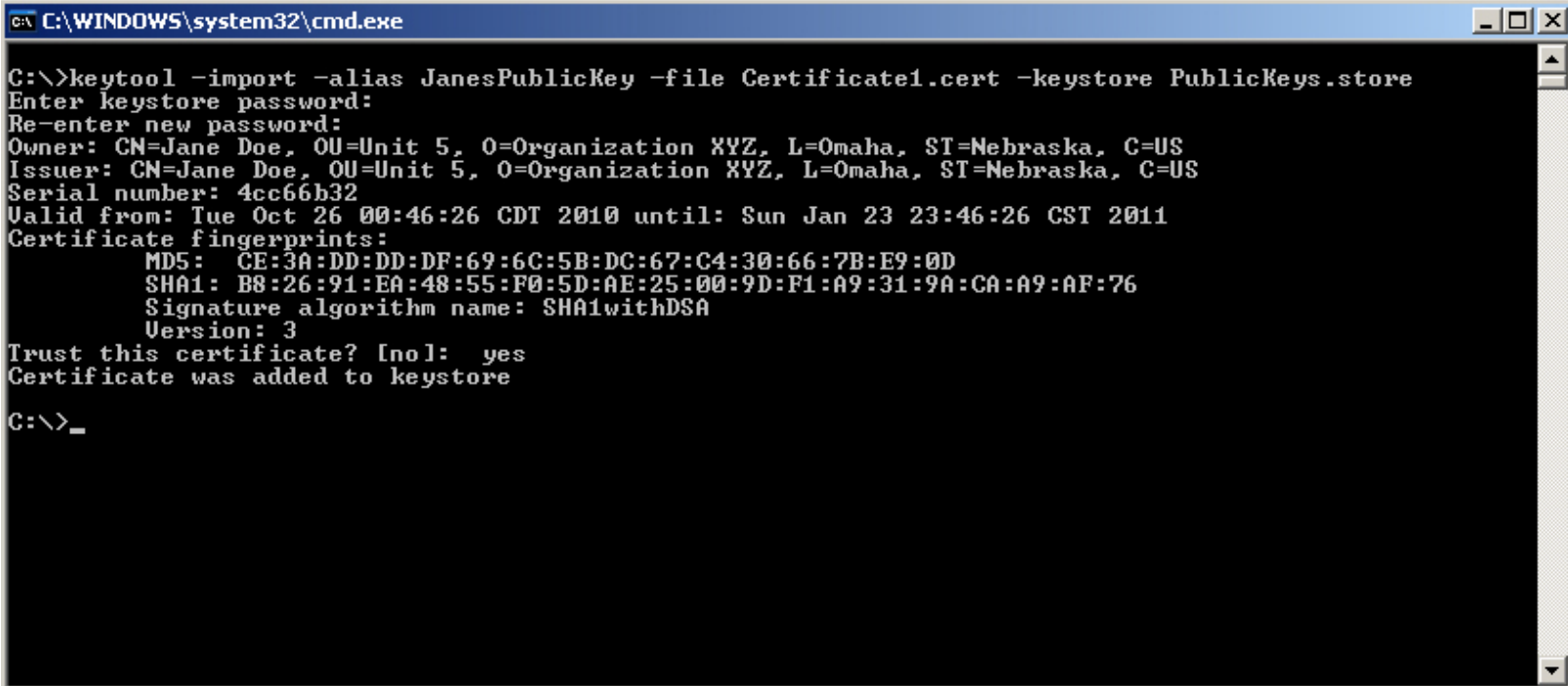
where *%alias%* and *%keystore%* are different values than you used previously, and *%file%* is the name of the file you saved your certificate to.

You will also be asked to create a password for this new keystore.

Import A Certificate

When you have imported your certificate to your new keystore, you should see the contents of the certificate displayed in the terminal/command window.

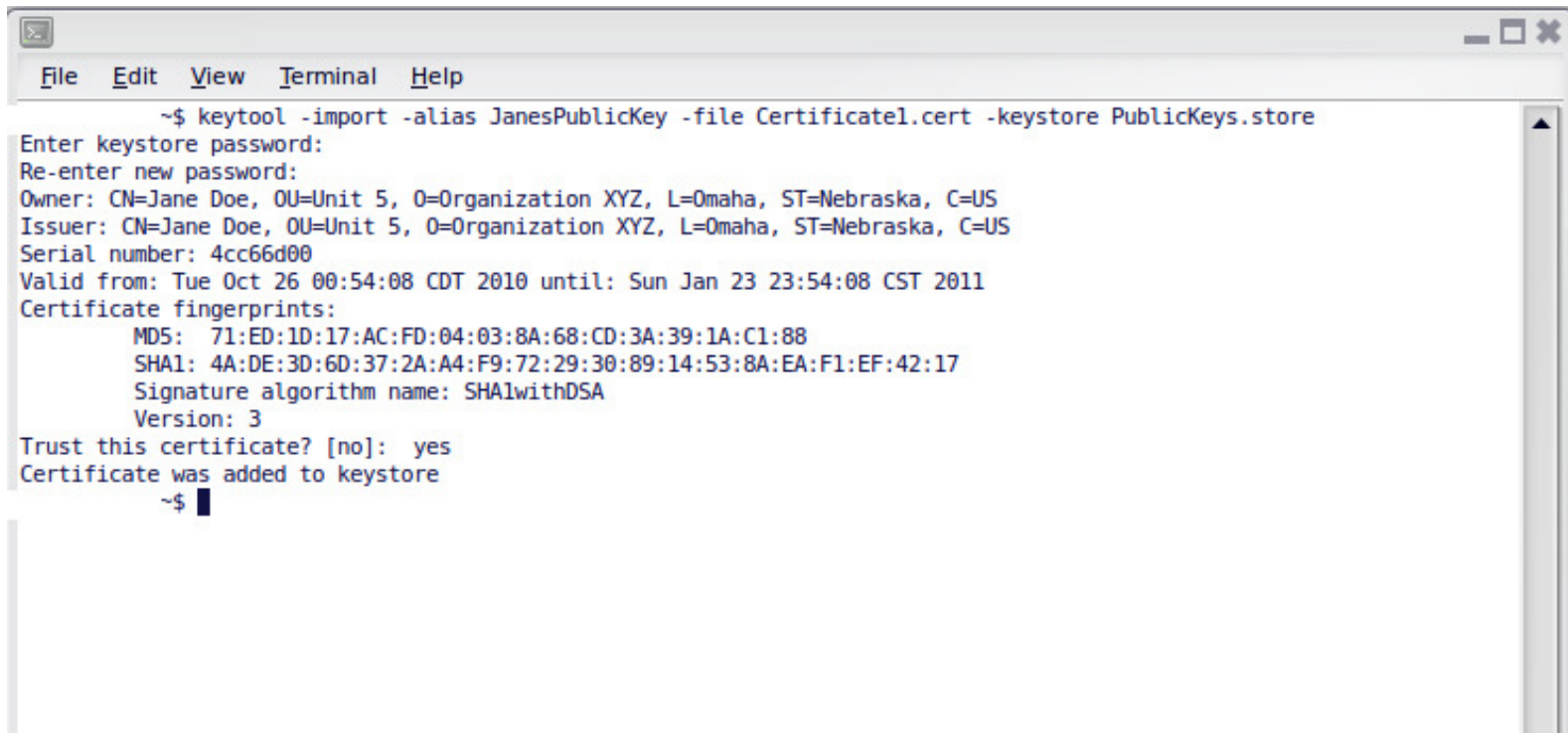
Windows



```
C:\>keytool -import -alias JanesPublicKey -file Certificate1.cert -keystore PublicKeys.store
Enter keystore password:
Re-enter new password:
Owner: CN=Jane Doe, OU=Unit 5, O=Organization XYZ, L=Omaha, ST=Nebraska, C=US
Issuer: CN=Jane Doe, OU=Unit 5, O=Organization XYZ, L=Omaha, ST=Nebraska, C=US
Serial number: 4cc66b32
Valid from: Tue Oct 26 00:46:26 CDT 2010 until: Sun Jan 23 23:46:26 CST 2011
Certificate fingerprints:
    MD5: CE:3A:DD:DD:DF:69:6C:5B:DC:67:C4:30:66:7B:E9:0D
    SHA1: B8:26:91:EA:48:55:F0:5D:AE:25:00:9D:F1:A9:31:9A:CA:A9:AF:76
    Signature algorithm name: SHA1withDSA
    Version: 3
Trust this certificate? [no]: yes
Certificate was added to keystore

C:\>_
```

Linux



```
~$ keytool -import -alias JanesPublicKey -file Certificat1.cert -keystore PublicKeys.store
Enter keystore password:
Re-enter new password:
Owner: CN=Jane Doe, OU=Unit 5, O=Organization XYZ, L=Omaha, ST=Nebraska, C=US
Issuer: CN=Jane Doe, OU=Unit 5, O=Organization XYZ, L=Omaha, ST=Nebraska, C=US
Serial number: 4cc66d00
Valid from: Tue Oct 26 00:54:08 CDT 2010 until: Sun Jan 23 23:54:08 CST 2011
Certificate fingerprints:
    MD5: 71:ED:1D:17:AC:FD:04:03:8A:68:CD:3A:39:1A:C1:88
    SHA1: 4A:DE:3D:6D:37:2A:A4:F9:72:29:30:89:14:53:8A:EA:F1:EF:42:17
    Signature algorithm name: SHA1withDSA
    Version: 3
Trust this certificate? [no]: yes
Certificate was added to keystore
~$
```

Import A Certificate

Locate the file “Certificate1.cert” that was included with this module and place it in the same location that your keystores and certificate were saved to.

Now import this certificate (Certificate1.cert) into the keystore to which you imported your certificate.

Import A Certificate

When you have imported Certificate1.cert, answer the following questions:

1. What is the owner's name?
2. In what city and state do the owner live?
3. When was the certificate created? ("Valid from: " date)
4. Is the owner the same as the issuer?
5. What is this type of certificate called?
6. What is the name of the signature algorithm used?

Import A Certificate

Answers:

1. Albert Hannover
2. Birmingham, Alabama
3. Tuesday, October, 26, 2010
4. Yes
5. Self-authenticated
6. SHA1withDSA

Import A Certificate

Now you should be able to create keys and certificates, export certificates to a file, and import certificates to a keystore using Java's keytool application.