Network Security Controls, Protocols, and Devices

Module 03

Lab 1

Encrypting Data at Rest Using VeraCrypt

VeraCrypt adds enhanced security to the algorithms used for system and partitions encryption, making it immune to new developments in brute-force attacks.

Lab Scenario

Administrators should ensure the security of data at rest through an encryption mechanism. It helps them to prevent data from unauthorized access. Administrators may require to encrypt data such as files, folders, disk volume, partitions, etc. Administrators should consider encrypting all the sensitive data of an organization that needs protection. As an administrator you should know how to handle various encryption tools used for encrypting data at rest.

ICON KEY

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

Lab Objectives

This lab will demonstrate how to encrypt data at rest.

Lab Environment

To carry out the lab, you need:

- VeraCrypt, located at **Z:\CND-Tools\CND Module 03 Network Security Controls, Protocols, and Devices\Data Encryption Tools\VeraCrypt**
- You can also download the latest version of VeraCrypt from the link https://veracrypt.codeplex.com/
- If you decide to download the latest version, then screenshots shown in the lab may differ
- A virtual machine running Windows 10
- A Web browser with Internet connection
- **Administrative** privileges to run tools

Lab Duration

Time: 20 Minutes

Overview of Veracrypt

VeraCrypt is a software application used for on-the-fly encryption (OTFE). It is distributed without cost, and the source code is available. It can create a virtual encrypted disk within a file, or encrypt a partition or an entire storage device.

Lab Tasks

TASK 1

Launching VeraCrypt

1. Turn on **Windows 10** and log in as an **Administrator**
2. Navigate to **Z:\CND-Tools\CND Module 03 Network Security Controls, Protocols, and Devices\Data Encryption Tools\Veracrypt** and double-click **VeraCrypt Setup 1.17.exe** file to start the installation
3. The **User Account Control** window pops up. Click **Yes** and follow the wizard driven installation steps to install VeraCrypt.

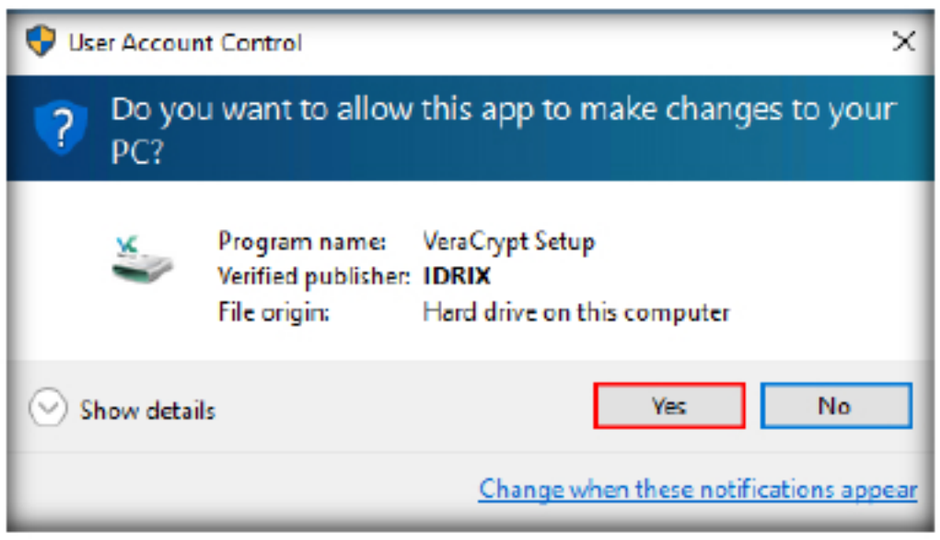

FIGURE 1.1: User Account Control

4. Open the **Start** → **All Apps** and click VeraCrypt to launch it.

FIGURE 1.2: Launch VeraCrypt

VeraCrypt is a software application used for on-the-fly encryption (OTFE). It is distributed without cost and the source code is available.

5. The VeraCrypt **main window** appears; click **Create Volume**.

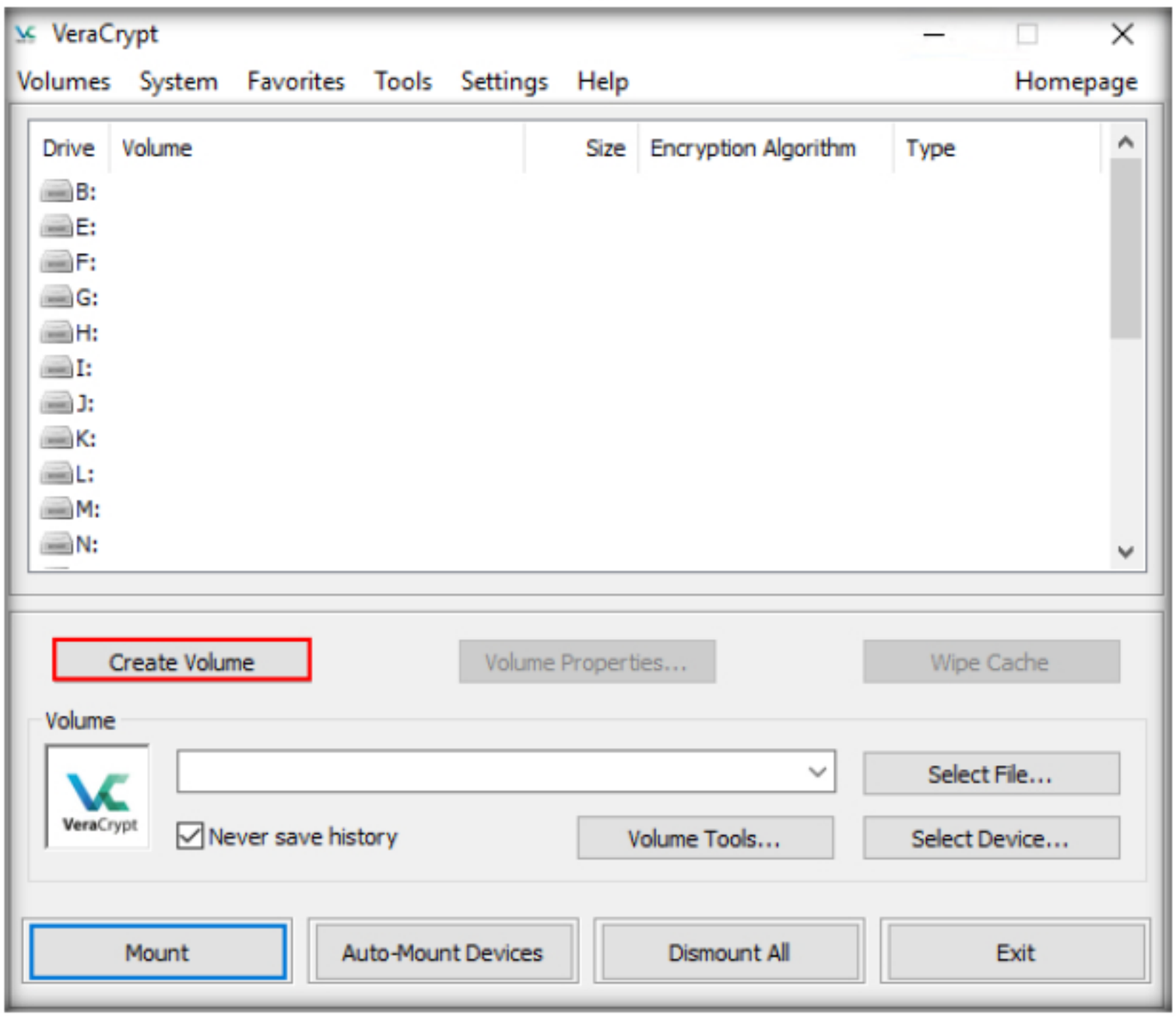

FIGURE 1.3: VeraCrypt Main Window

VeraCrypt have the ability to create and run a hidden encrypted operating system whose existence may be denied.

6. The **VeraCrypt Volume Creation Wizard** window appears.
7. Select **Create an encrypted file container** to create a file containing a virtual, encrypted disk.

IMPORTANT: Note that VeraCrypt will not encrypt any existing files (when creating a VeraCrypt file container). If you select an existing file in this step, it will be overwritten and replaced by the newly created volume (so the overwritten file will be lost, not encrypted). You will be able to encrypt existing files (later on) by moving them to the VeraCrypt volume that we are creating now.

FIGURE 1.4: VeraCrypt Volume Creation Wizard

8. In the **Volume Type** wizard, select **Standard VeraCrypt volume**. This creates a **normal** VeraCrypt volume.
9. Click **Next** to proceed.

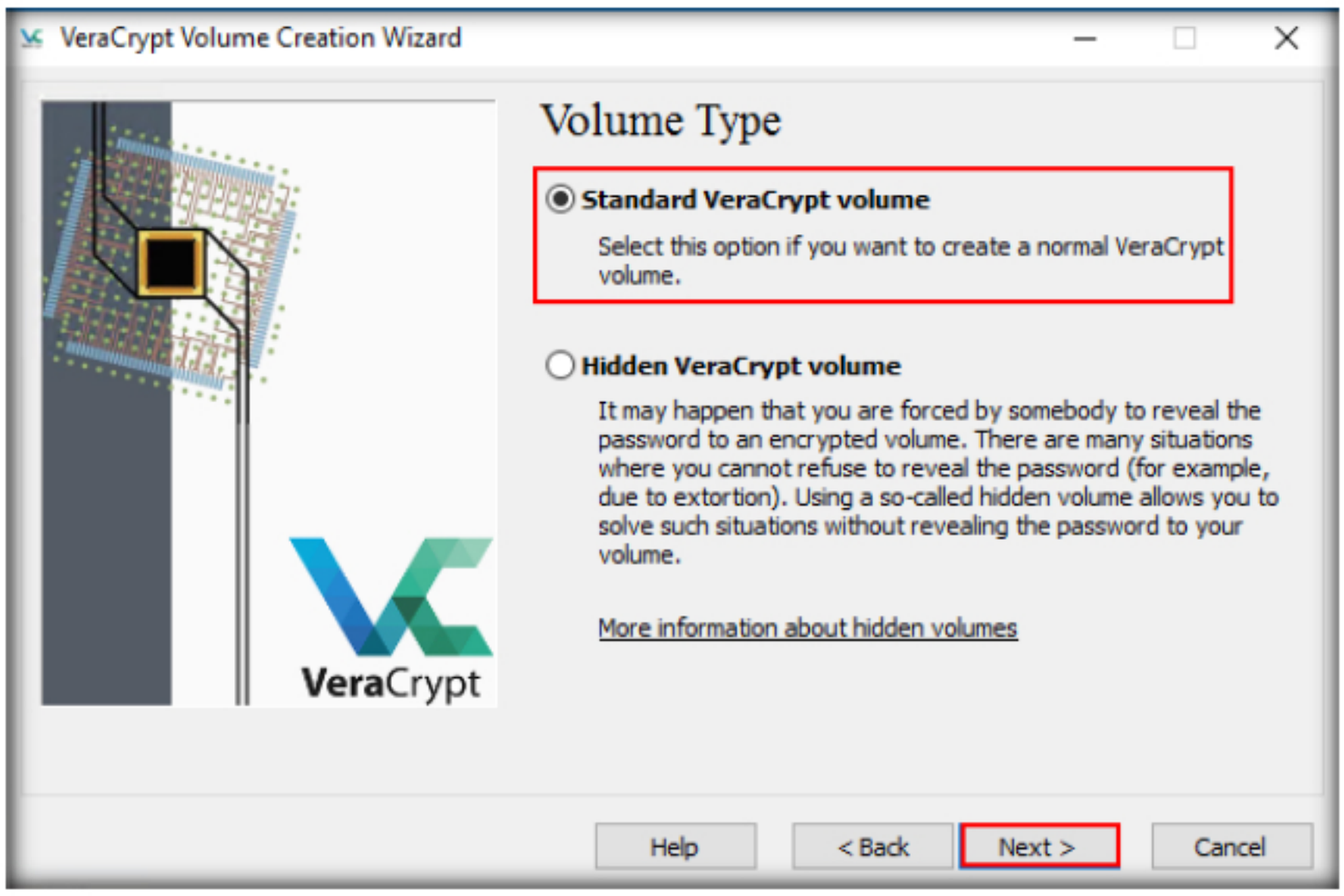

FIGURE 1.5: VeraCrypt Volume Creation Wizard-Volume Type

Note: After you copy existing unencrypted files to a VeraCrypt volume, you should securely erase (wipe) the original unencrypted files. There are software tools that can be used for the purpose of secure erasure (many of them are free).

10. In the **Volume Location** wizard, click **Select File...**.

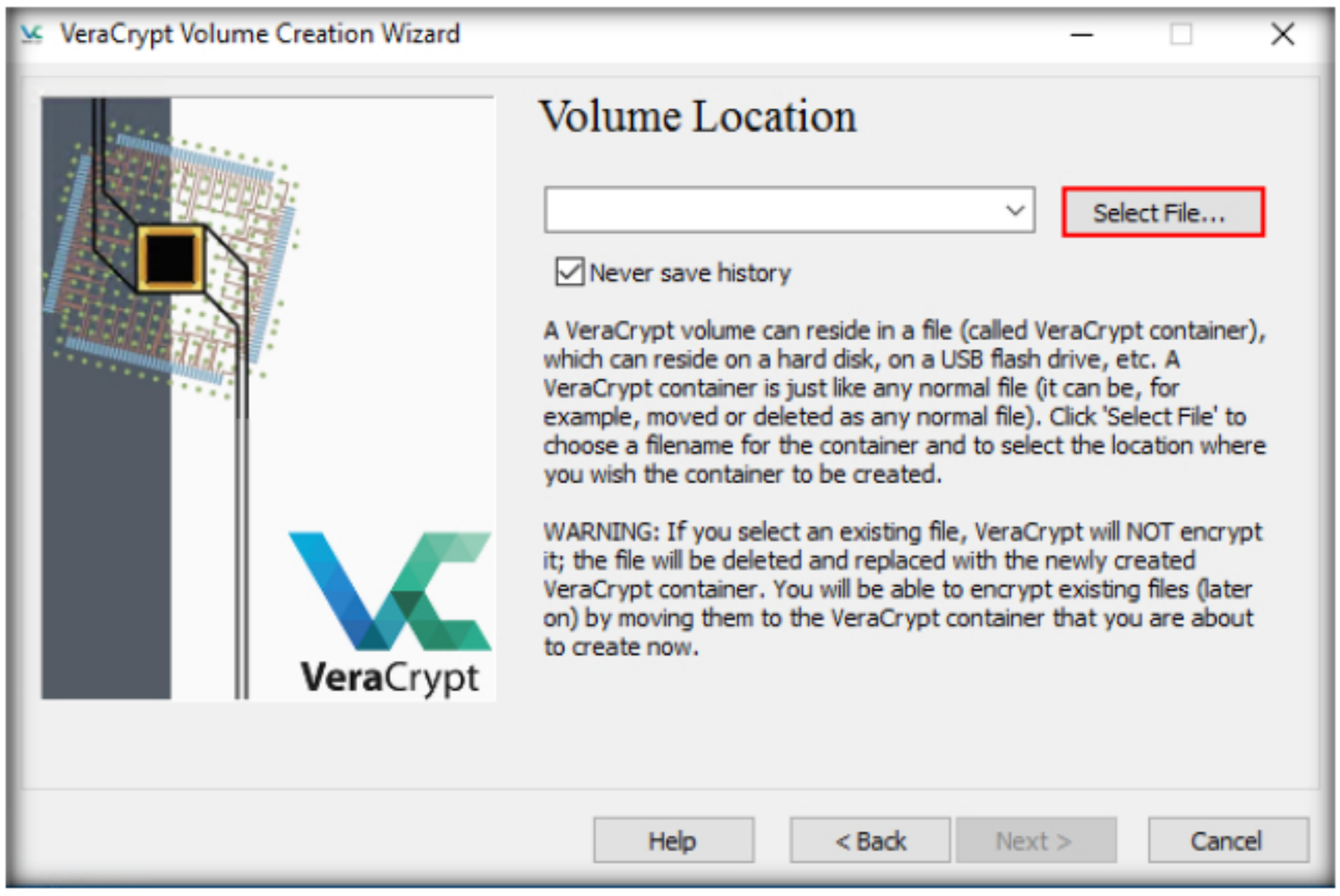

FIGURE 1.6: VeraCrypt Volume Creation Wizard-Volume Location

VeraCrypt supports a concept called plausible deniability.

11. The **Specify Path and File Name** window appears; navigate to the desired location (here, **Desktop**), provide the File name as **MyVolume,** and click **Save**.

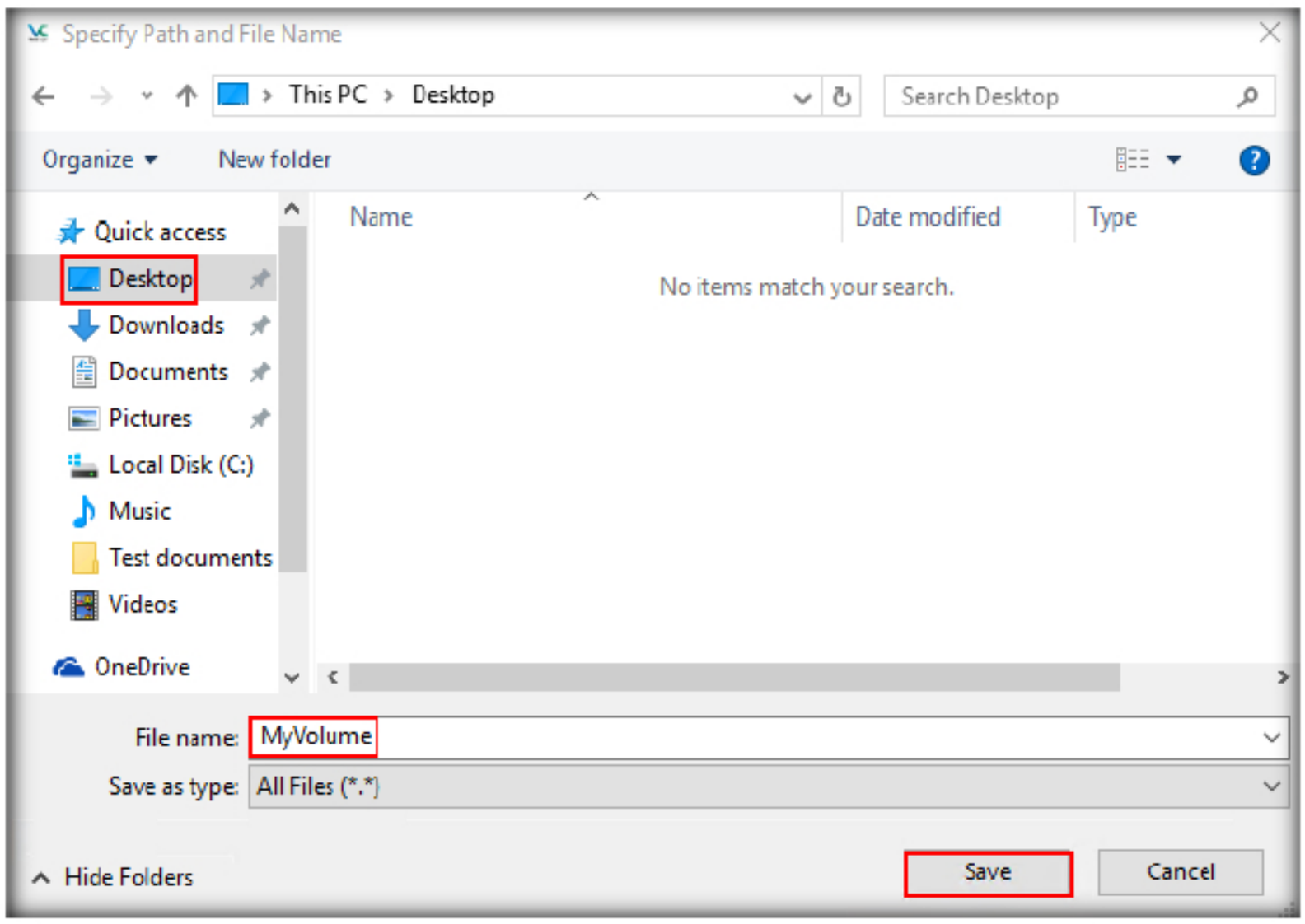

FIGURE 1.7: Windows Standard-Specify Path and File Name Window

The mode of operation used by VeraCrypt for encrypted partitions, drives, and virtual volumes is XTS.

12. After **saving** the file, the location of file containing the **VeraCrypt** volume is set; click **Next**.

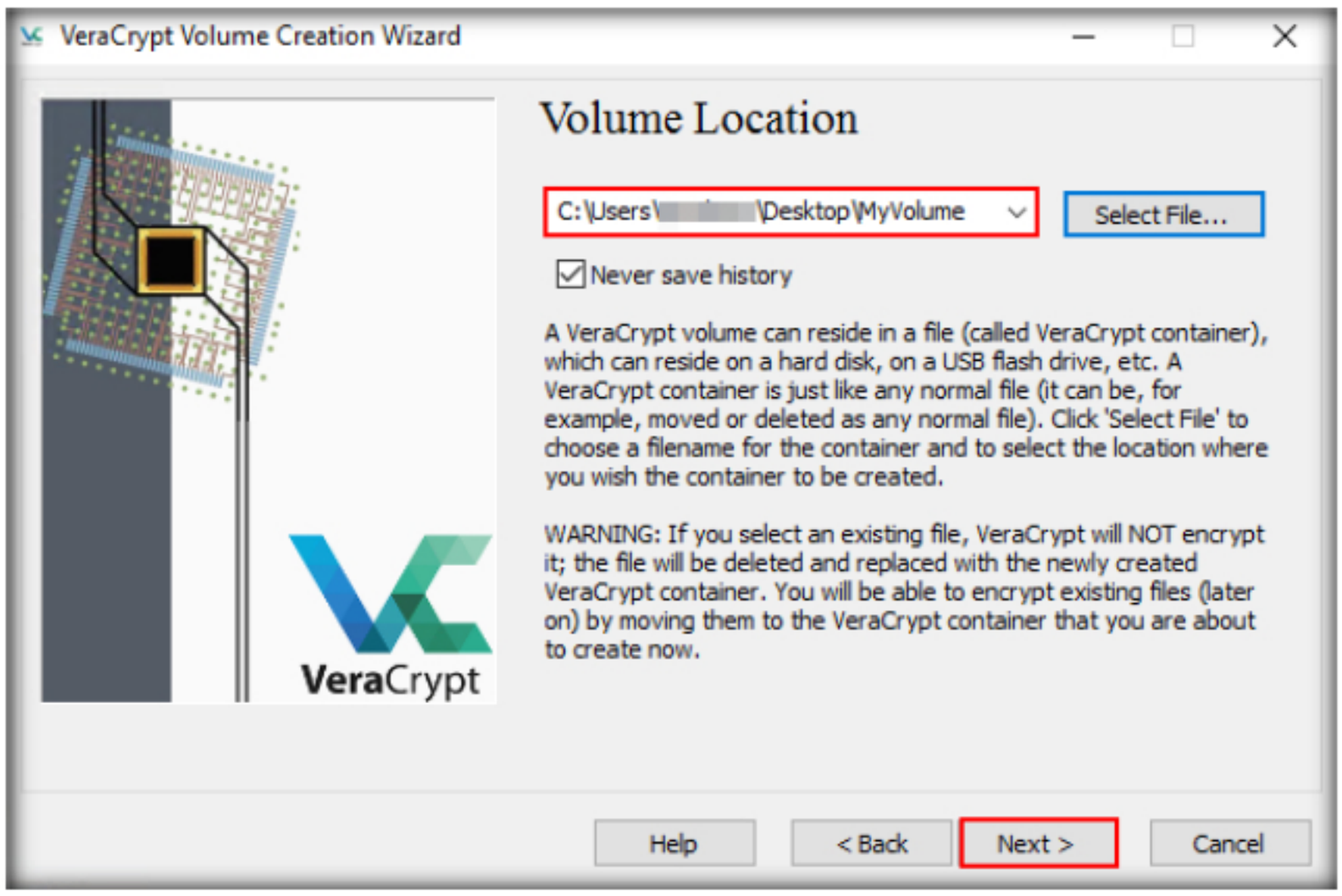

FIGURE 1.8: VeraCrypt Volume Creation Wizard-Volume Location

VeraCrypt volumes do not contain known file headers and their content is indistinguishable from random data.

13. In the **Encryption Options** wizard, select the **AES** Encryption Algorithm and **SHA-512** Hash Algorithm, and click **Next**.

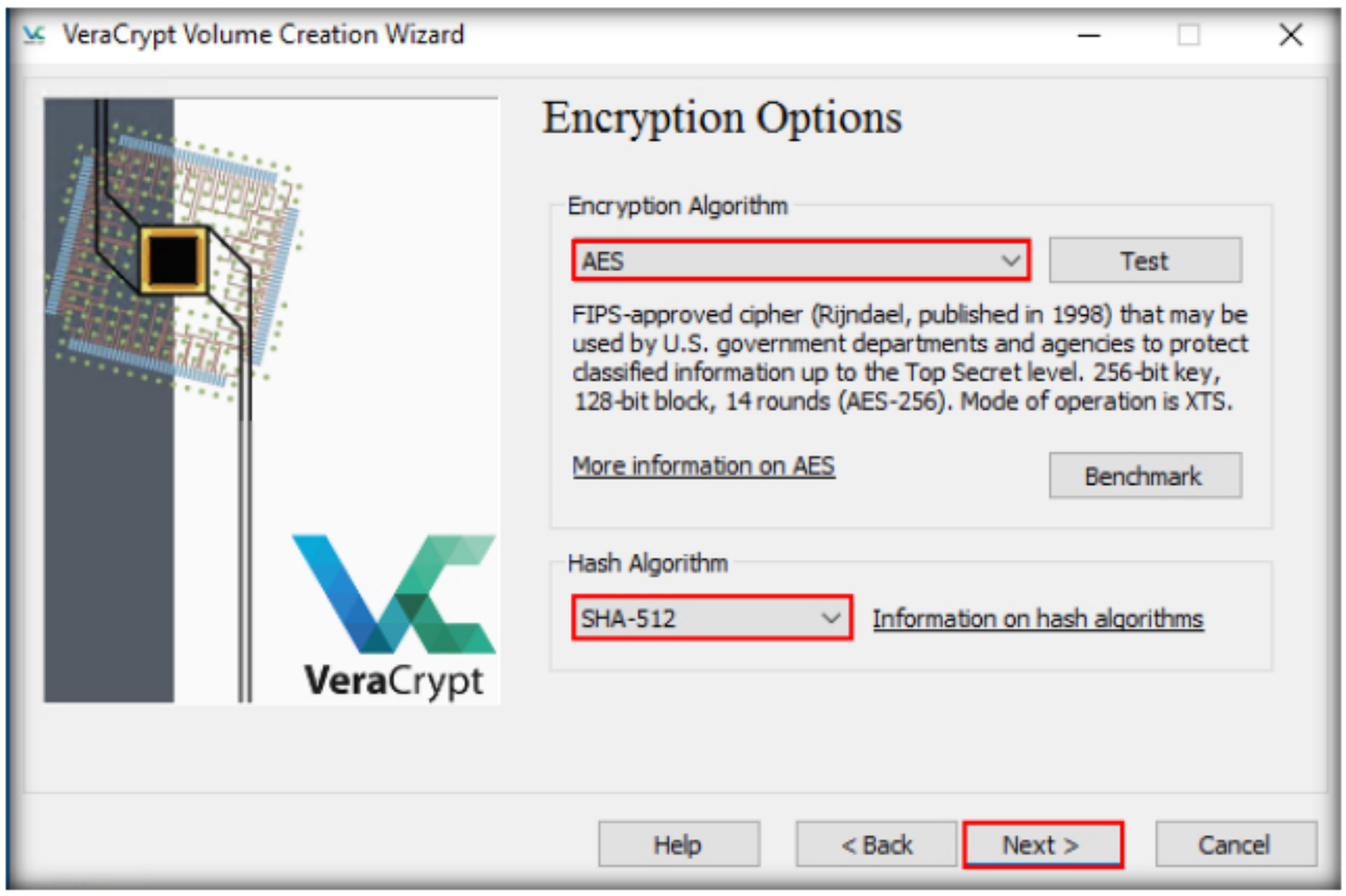

FIGURE 1.9: VeraCrypt Volume Creation Wizard-Encryption Options

VeraCrypt currently supports the following hash algorithms:

- RIPEMD-160
- SHA-512
- Whirlpool

14. In the **Volume Size** wizard, specify the size of the VeraCrypt container as **2 MB,** and click **Next**.

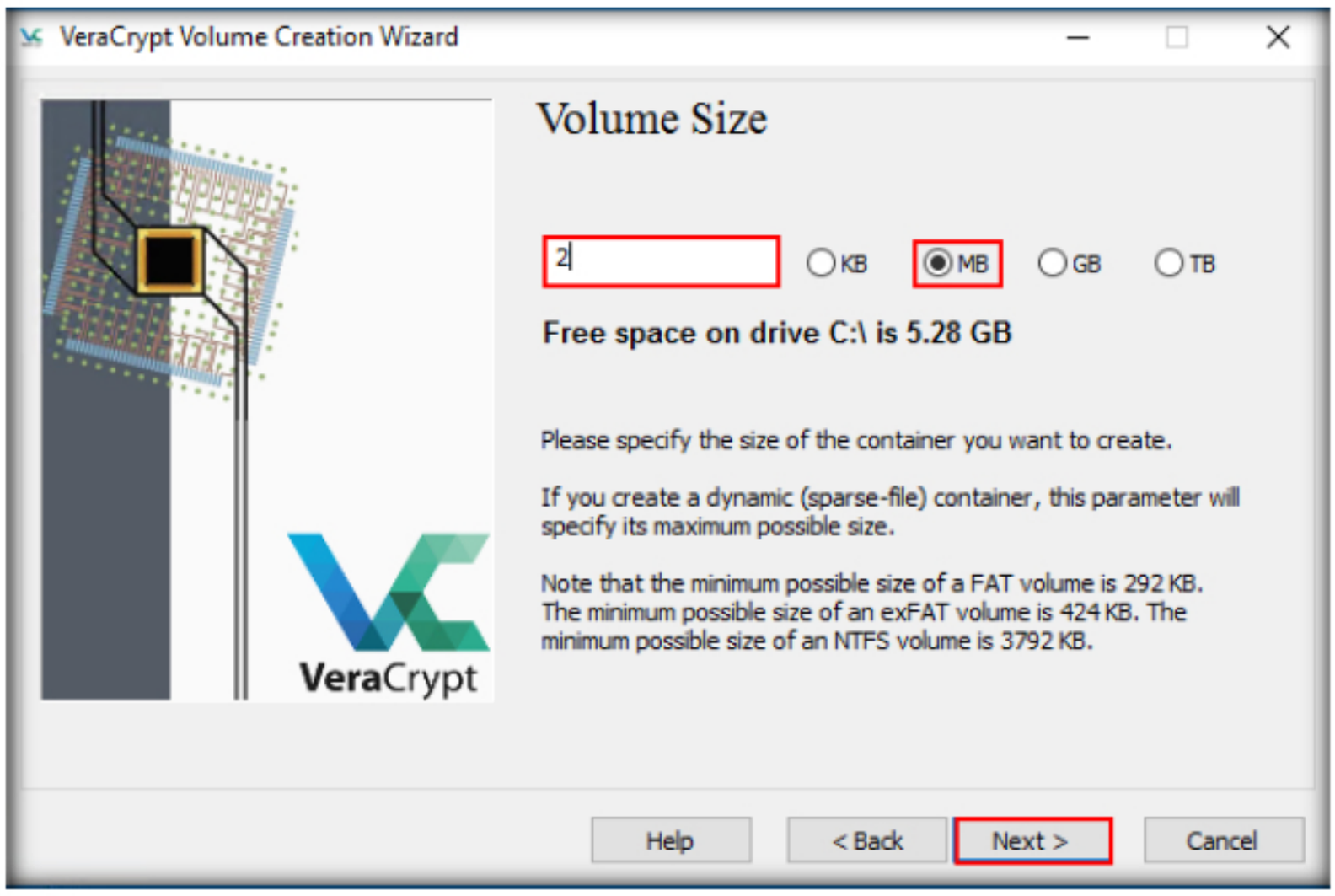

FIGURE 1.10: VeraCrypt Volume Creation Wizard-Volume Size

Note: The button "Next" will be disabled until passwords in both input fields are the same.

15. The **Volume Password** wizard appears; provide a **good password** in the **Password** field, retype it in the **Confirm** field, and click **Next**.

16. In this lab, the password used is **qwerty@123**.

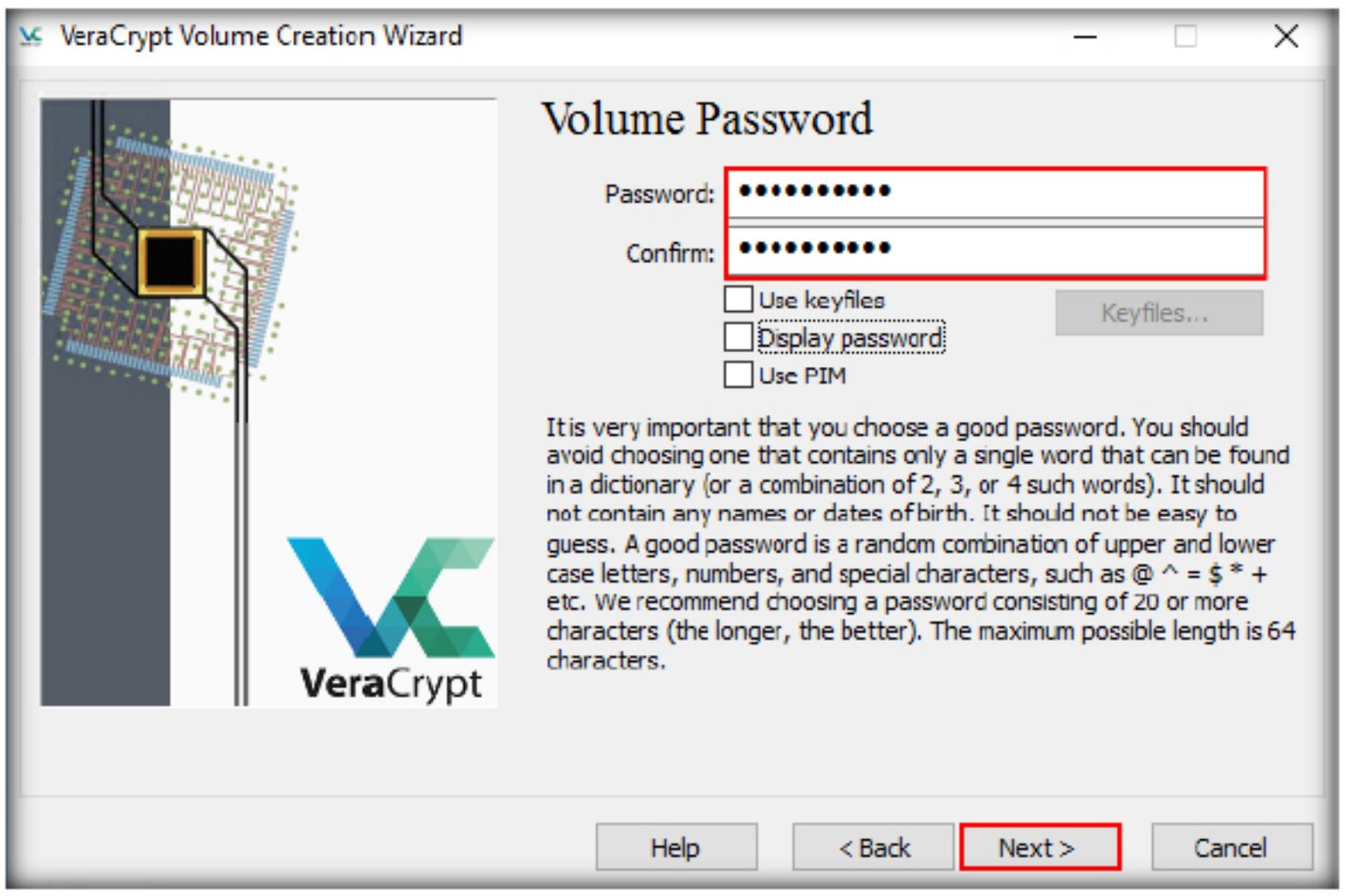

FIGURE 1.11: VeraCrypt Volume Creation Wizard-Volume Password

Note: A **VeraCrypt Volume Creation Wizard** warning pop-up appears; click **Yes**.

17. The Volume Format option appears. Select **FAT Filesystem**, and set the cluster to **Default**.
18. Move your mouse as **randomly** as possible within the Volume Creation Wizard window for at least **30 seconds**.
19. Click **Format**.

The longer you move the mouse, the better. This significantly increases the **cryptographic strength** of the encryption keys.

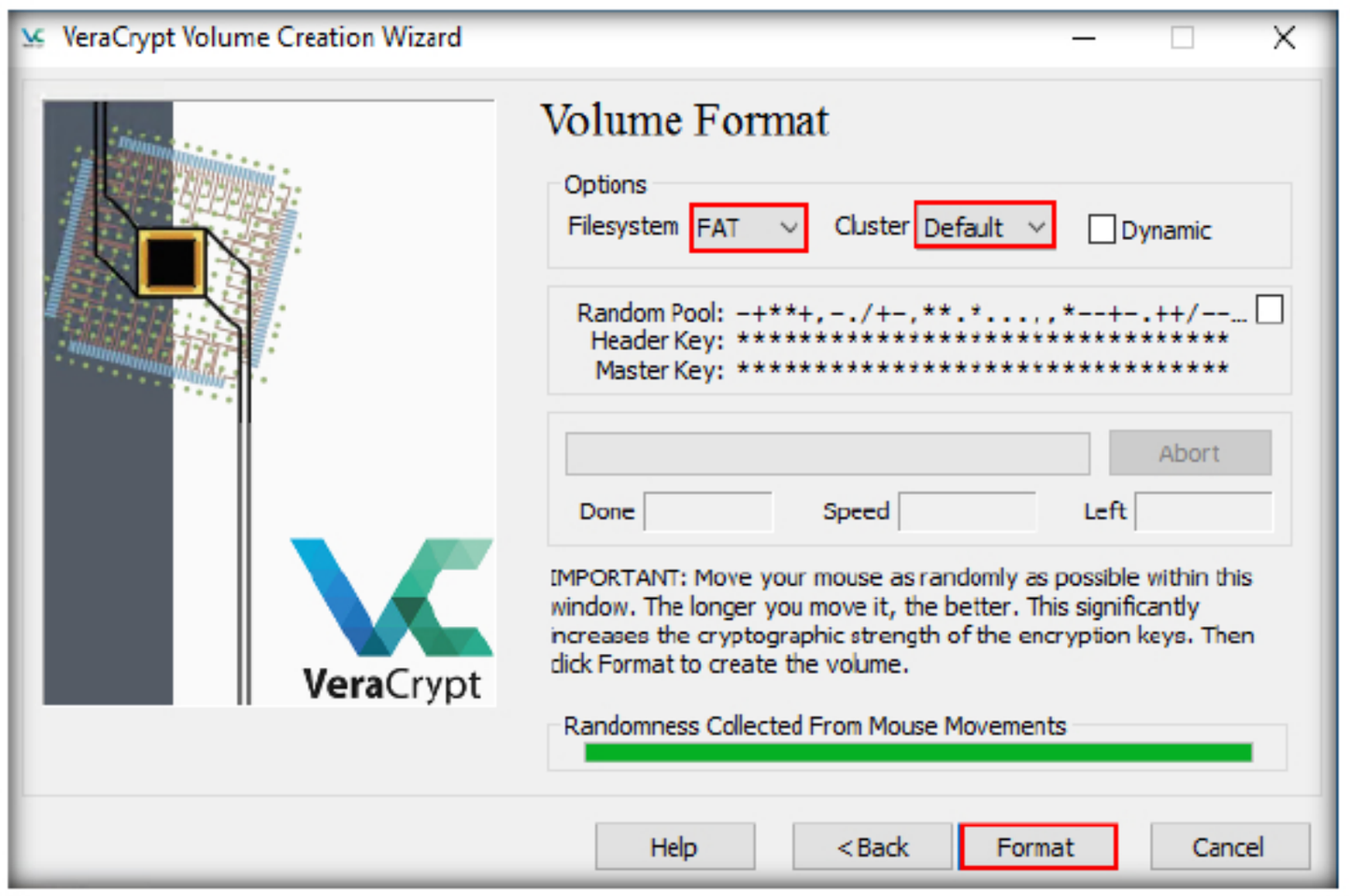

FIGURE 1.12: VeraCrypt Volume Creation Wizard-Volume Format

20. After clicking **Format,** VeraCrypt will create a file called **MyVolume** in the provided folder. This file depends on the VeraCrypt container (it will contain the encrypted VeraCrypt volume).
21. Depending on the **size of the volume**, it may take some time for volume creation.
22. Once the volume is created, a **VeraCrypt Volume Creation Wizard** dialog-box appears; click **OK**.

Free space on each VeraCrypt volume is filled with random data when the volume is created.

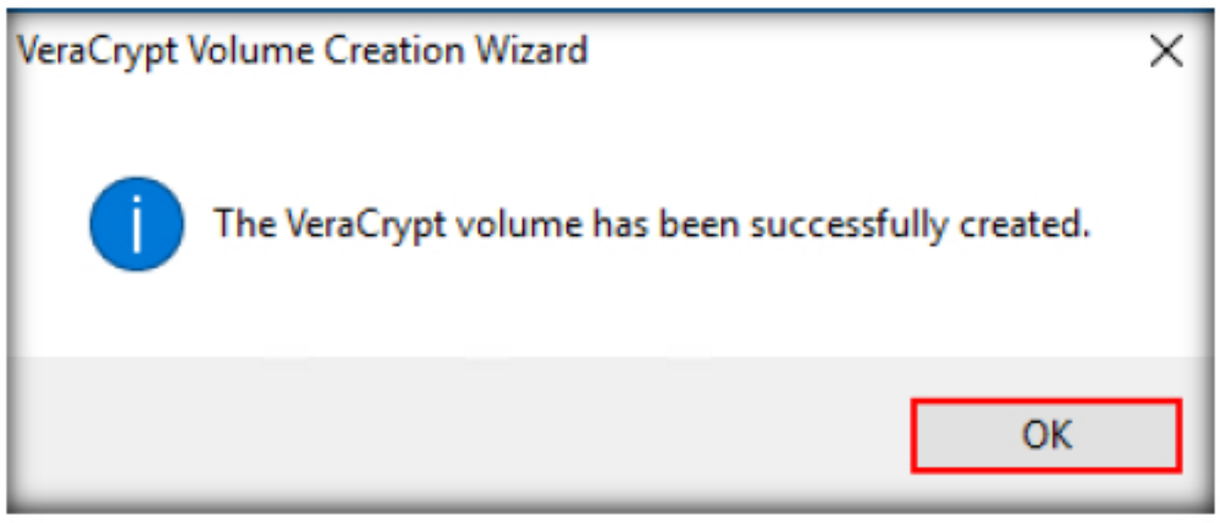

FIGURE 1.13: VeraCrypt Volume Creation Wizard Dialog Box

23. Click **OK** to close the dialog box.
24. You have **successfully** created a **VeraCrypt volume** (file container).
25. In the VeraCrypt Volume Creation wizard window, click **Exit**.

VeraCrypt is unable to secure data on a computer if an attacker physically accessed it and VeraCrypt is used on the compromised computer by the user again.

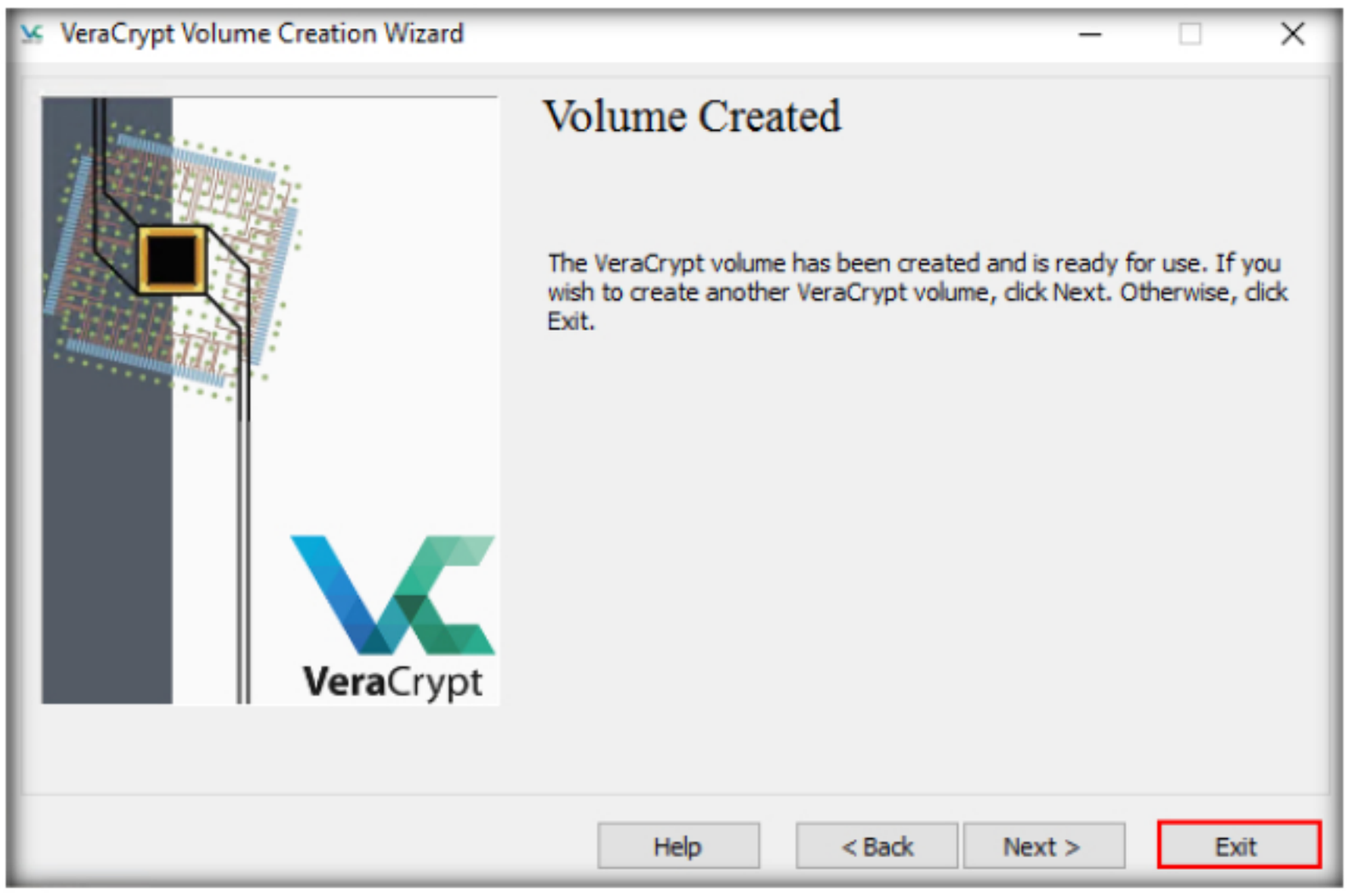

FIGURE 1.14: VeraCrypt Volume Creation Wizard-Volume Created

TASK 2

Mount a Volume

26. The **VeraCrypt** main window appears; select a drive (here, **E:**), and click **Select File...**.

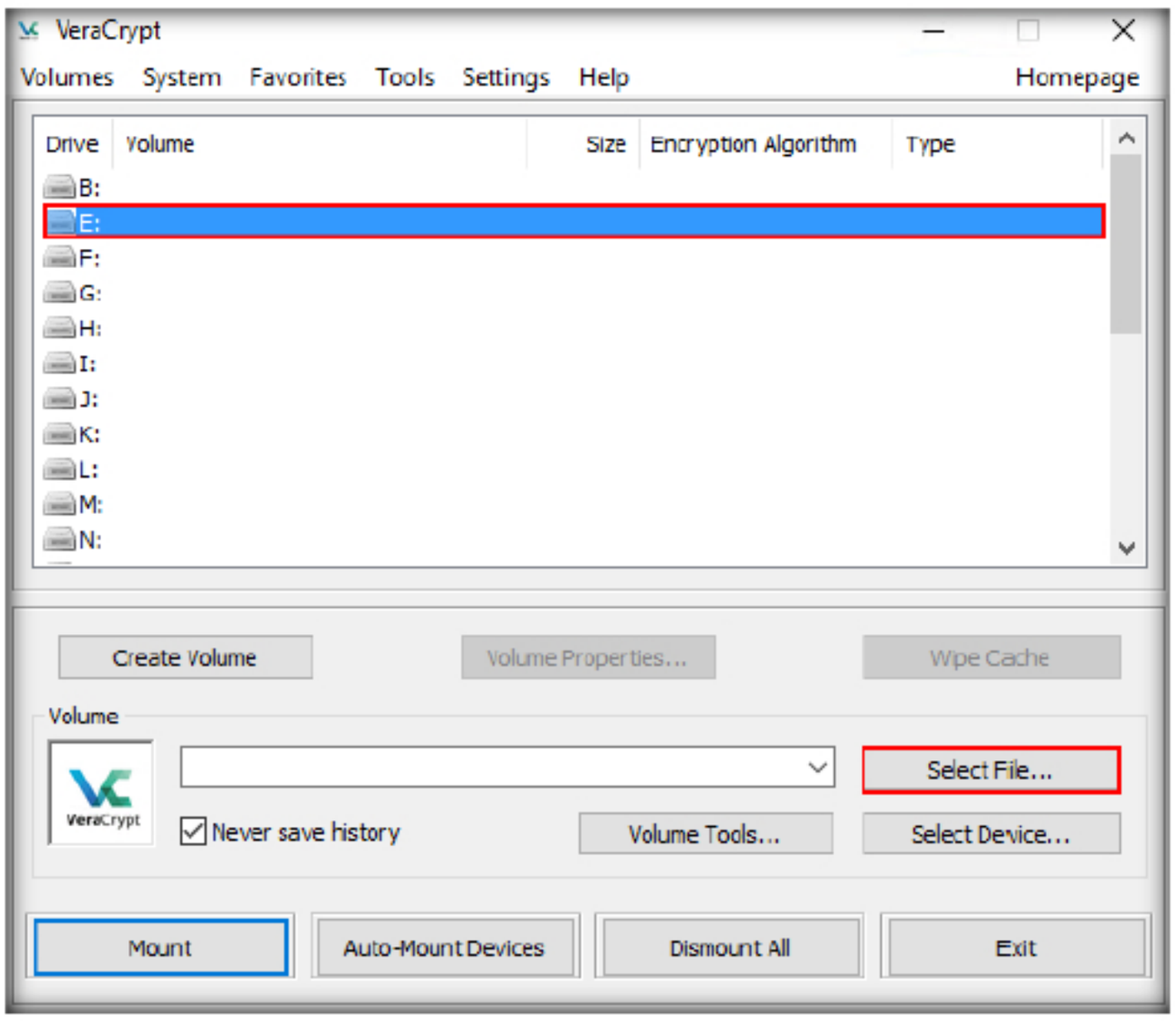

FIGURE 1.15: VeraCrypt Main Window with Select File Button

27. The **Select a VeraCrypt Volume** window appears; navigate to **This PC\Desktop**, click **MyVolume,** and click **Open**.

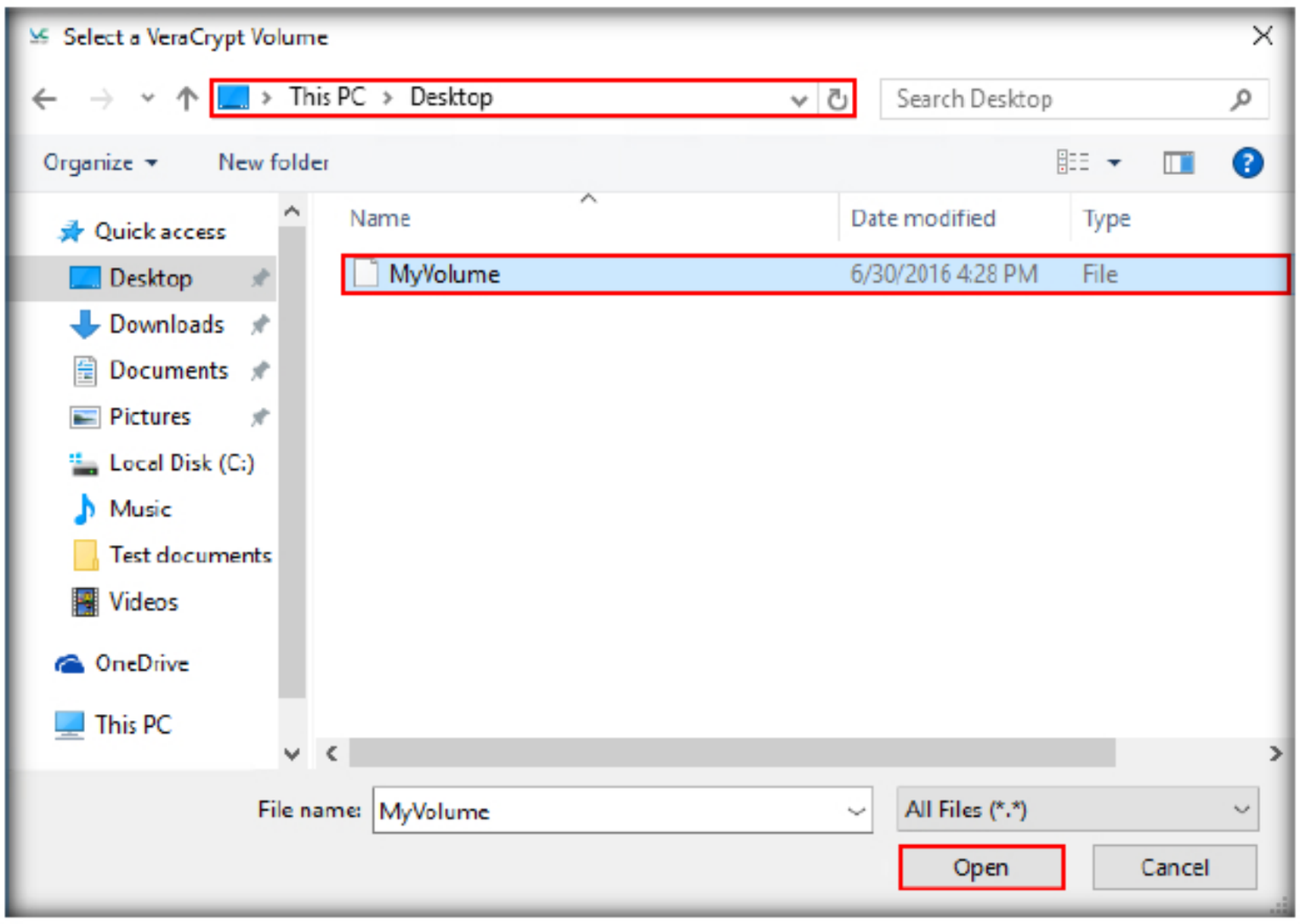

FIGURE 1.16: Windows Standard File Selector Window

Default mount options can be configured in the main program preferences (Settings → Preferences).

28. The window **closes** and you are returned to the **VeraCrypt** window. Click **Mount**.

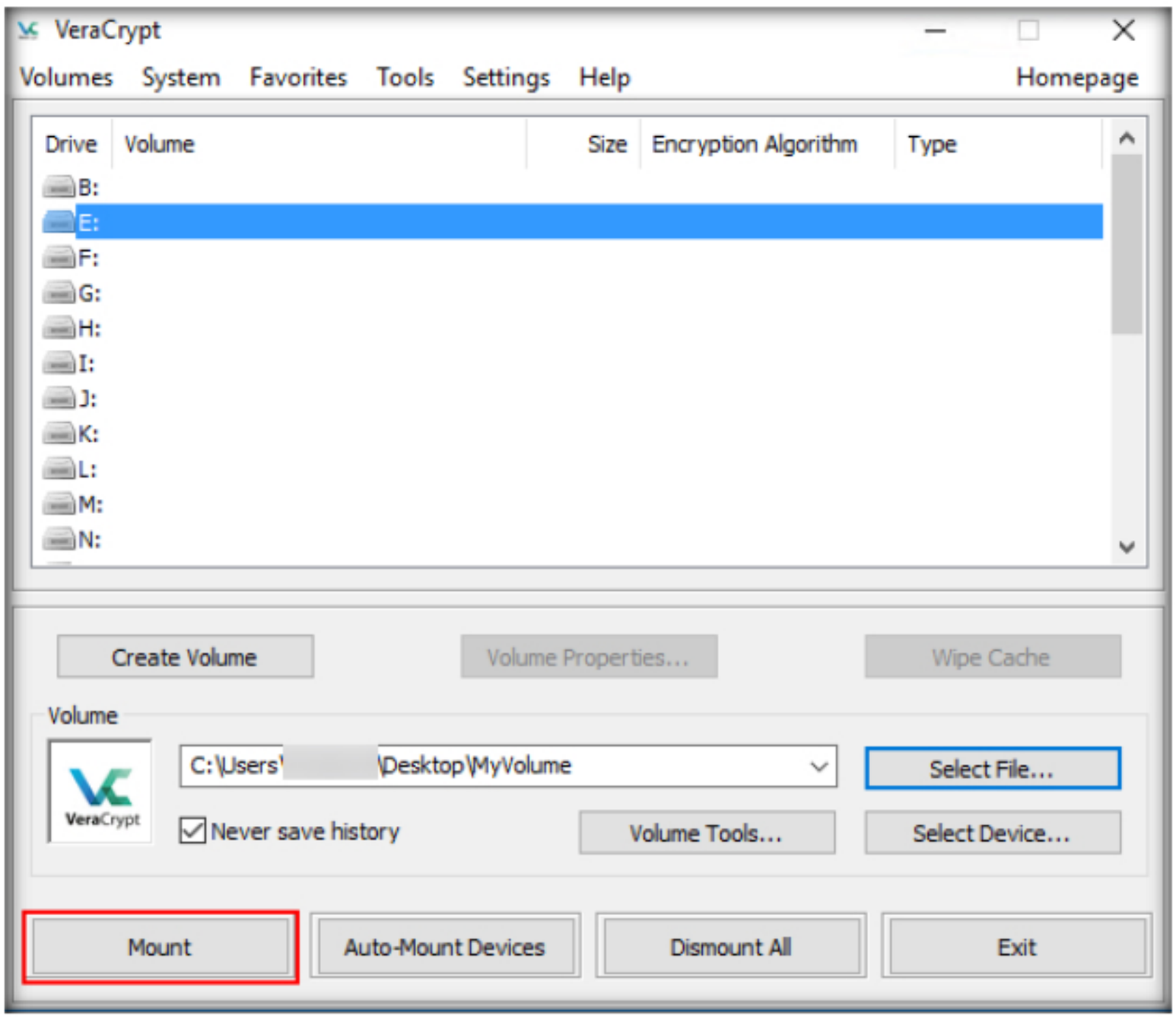

FIGURE 1.17: VeraCrypt Main Window with Mount Button

This option can be set in the password entry dialog so that it will apply only to that particular mount attempt. It can also be set as default in the Preferences.

29. The **Enter Password** dialog-box appears; type the password you specified earlier for this volume (in this lab, **qwerty@123**) in the **Password** input field, and click **OK**.

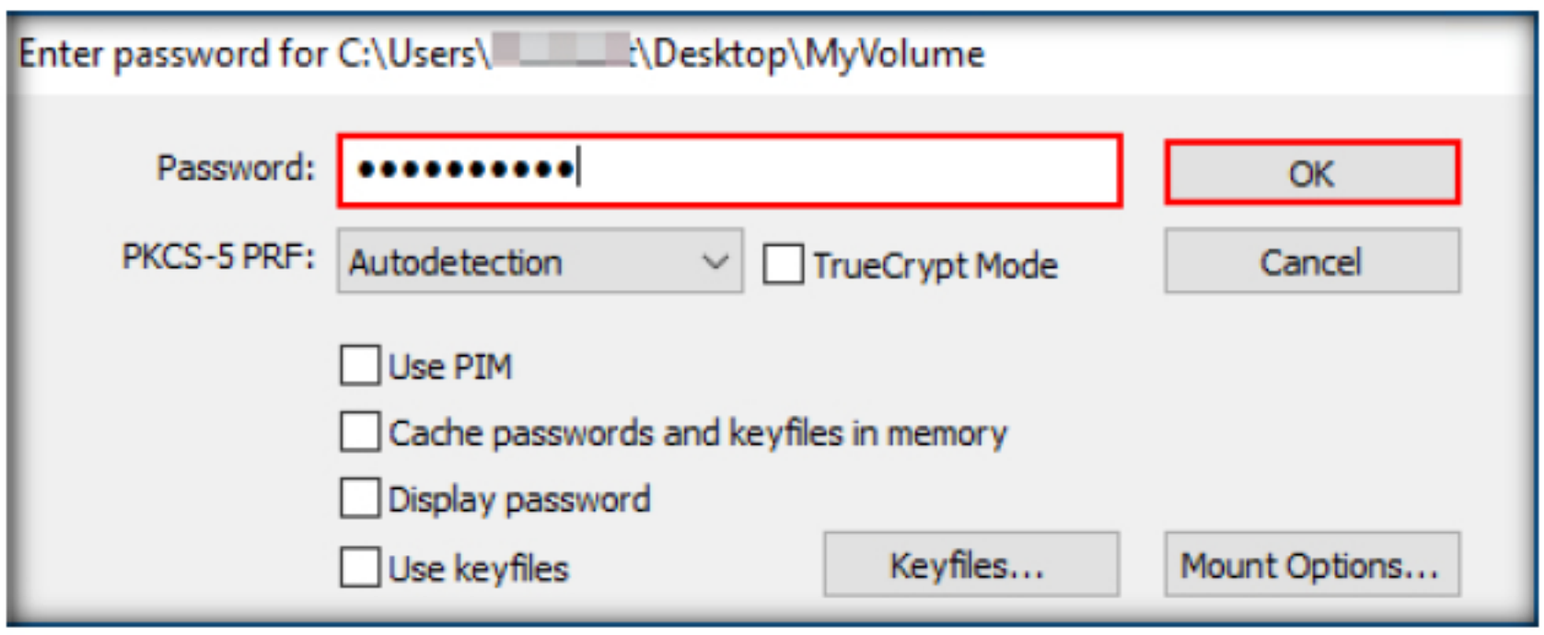

FIGURE 1.18: VeraCrypt Password Window

When a correct password is cached, volumes are automatically mounted after you click Mount. If you need to change mount options for a volume being mounted using a cached password, hold down the Control (Ctrl) key while clicking Mount, or select Mount with Options from the Volumes menu.

30. After the password is **verified**, VeraCrypt will **mount the volume,** as shown in the screenshot:

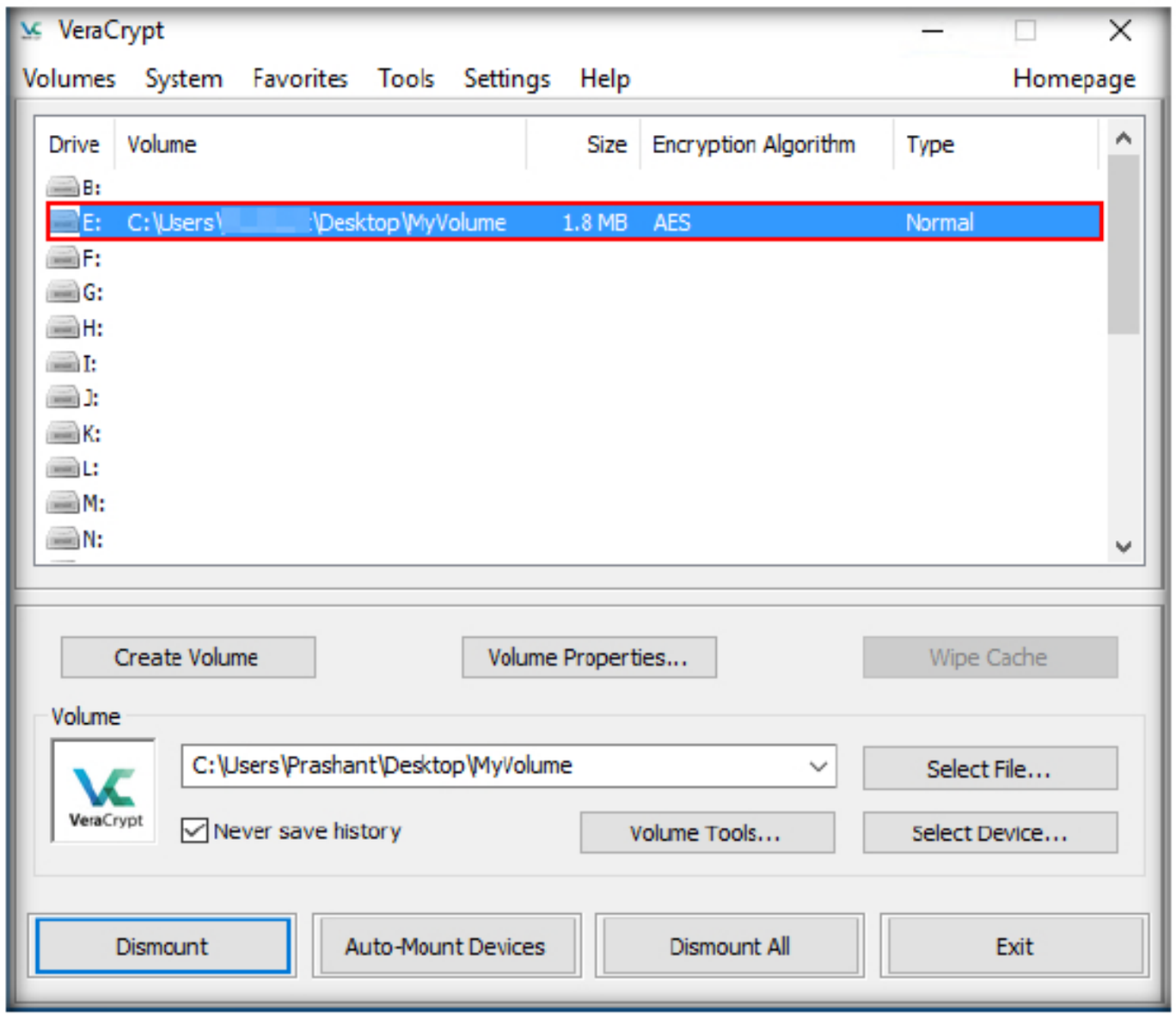

FIGURE 1.19: VeraCrypt Main Window

31. **MyVolume** has **successfully** mounted the container as a virtual disk (**E:**).
32. The virtual disk is entirely **encrypted** (including file names, allocation tables, free space, etc.) and behaves like a **real disk**.
33. You can copy or move files to this virtual disk to encrypt them.
34. Create a text document on the **Desktop** and name it **Test**.
35. Open the text document, and enter some text in it.
36. Click **File** in the menu bar, and click **Save**.

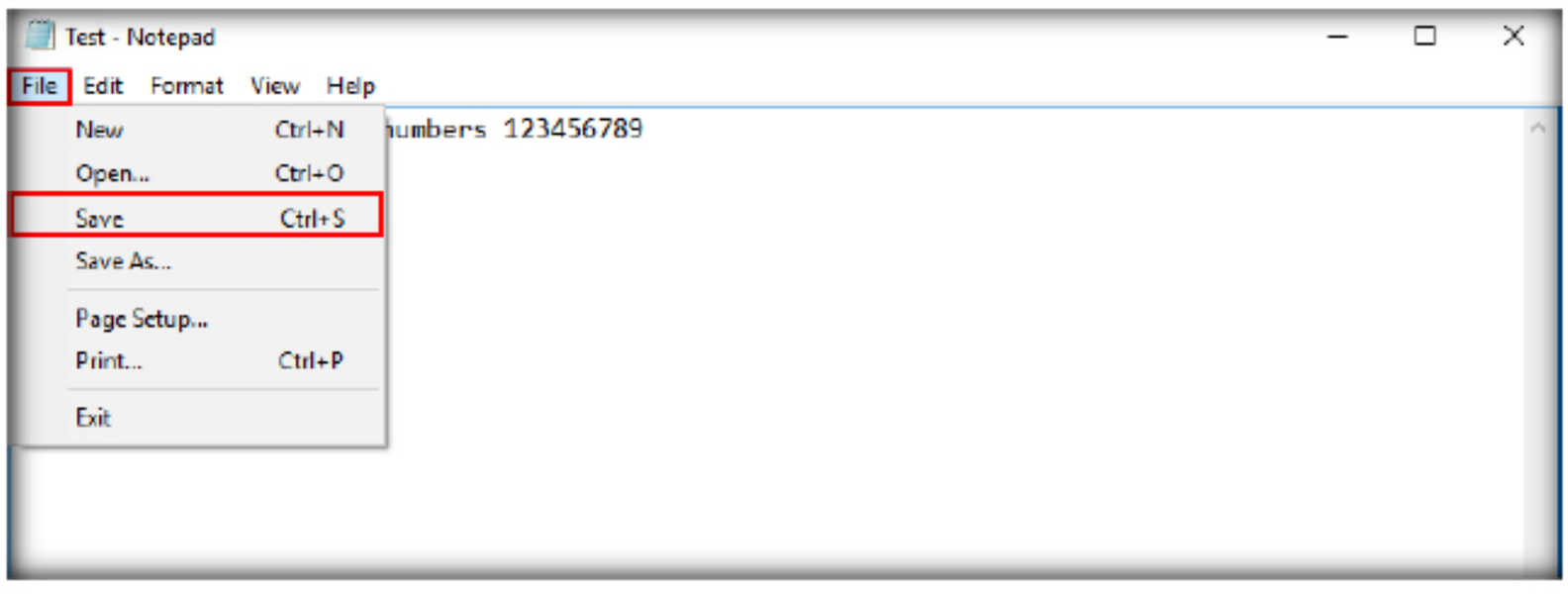

FIGURE 1.20: VeraCrypt Main Window with Dismount Button

No data stored on an encrypted volume can be read (decrypted) without using the correct password or correct encryption key.

VeraCrypt cannot automatically dismount all mounted VeraCrypt volumes on system shutdown/restart.

37. Copy the file from the **Desktop,** and paste it in **E:**. **Close** the window.

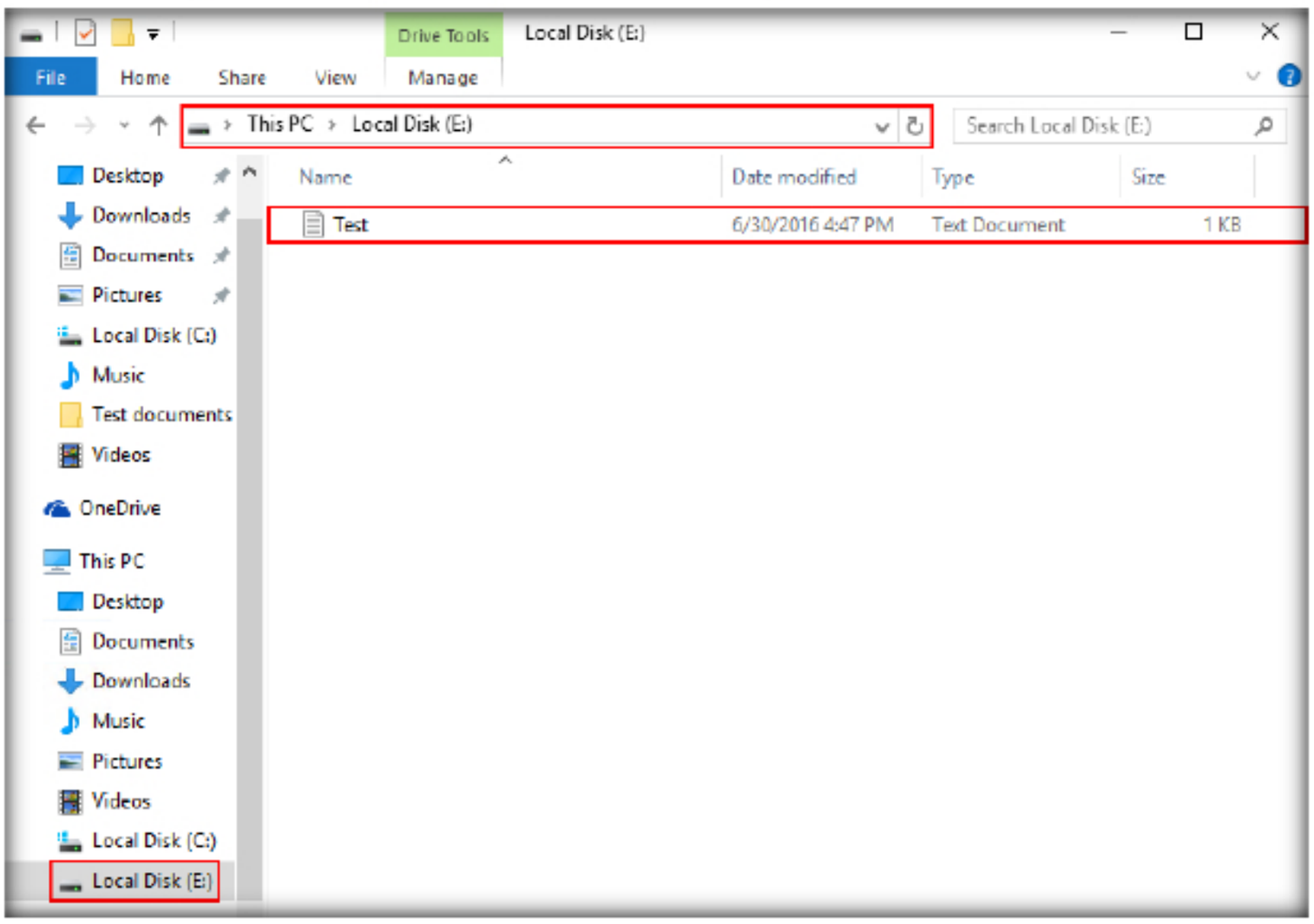

FIGURE 1.21: Test.txt file in Encrypted Container

38. Switch to **VeraCrypt** window, click **Dismount** and then click **Exit**.

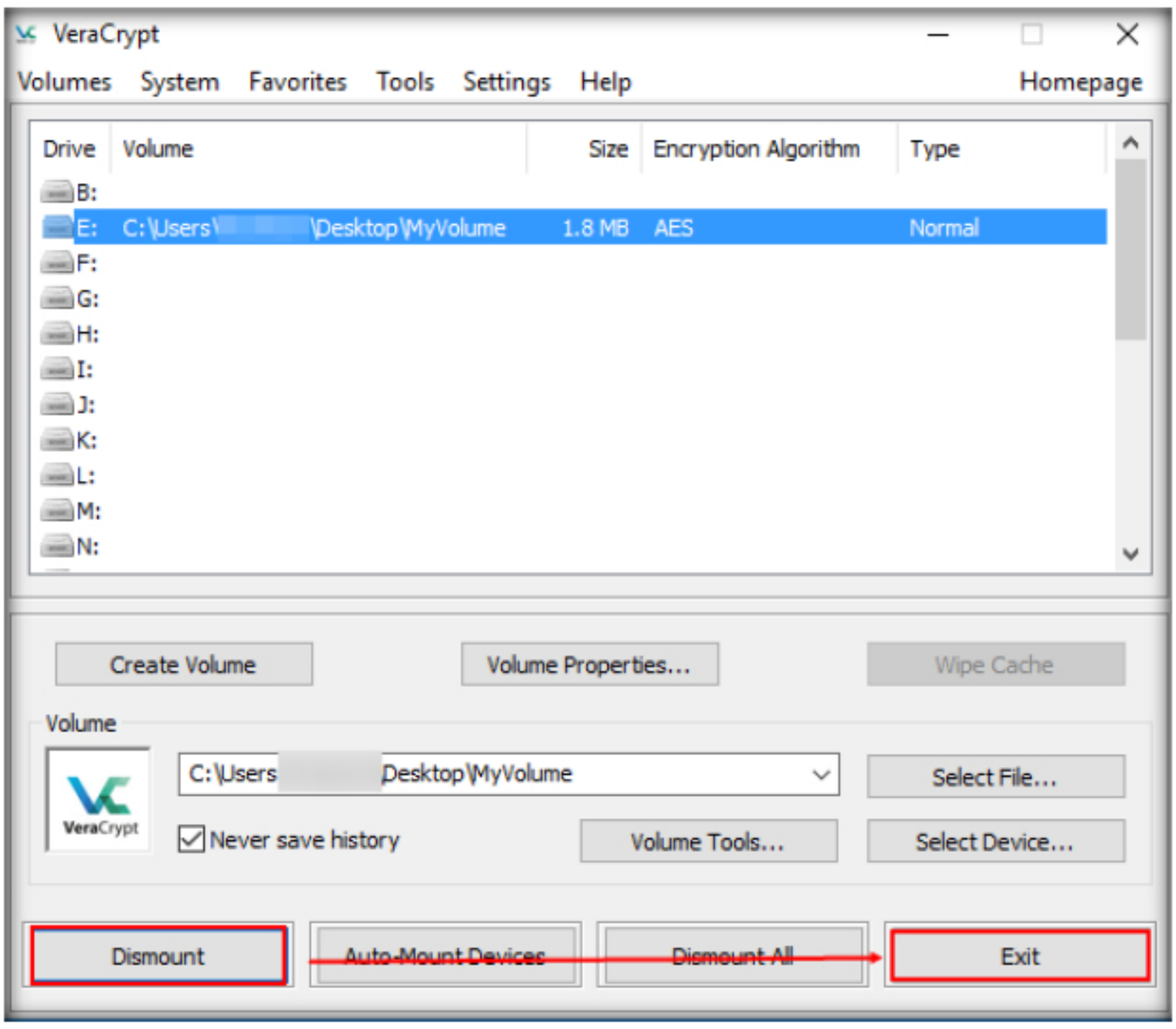

FIGURE 1.22: VeraCrypt Main Window with Dismount Button

The **E:** located in **This PC** disappears. In this way, you can encrypt entire data volumes to secure your data from prying eyes. Thus, all sensitive information located on the encrypted volume is safeguarded.

Lab Analysis

Analyze and document the results of the lab exercise. Give your opinion on your target's security posture and exposure through free public information.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS ABOUT THIS LAB.

Internet Connection Required	
☐ Yes	☑ No
Platform Supported	
☑ Classroom	☑ iLabs

Lab 2

Encrypting Data in Transit Using SSL

ICON KEY

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

SSL (Secure Sockets Layer) is the standard security technology for establishing an encrypted link between a web server and a browser.

Lab Scenario

Ensuring security of data at rest is not enough as data can be tampered with during transit if not properly secured. Organizations should implement SSL on their websites in order to encrypt data during transit. As a network administrator, you should know how to create a SSL enabled website for the organization.

Lab Objectives

The objective of this lab is to demonstrate how to enable SSL on a Website to transfer data over a secure channel. In this lab, you will learn to:

- Build a http website (WordPress)
- Provide SSL encryption to a website implemented on HTTP

Lab Environment

To complete this lab, you will need:

- You can download the latest version of WAMP Server from http://www.Wampserver.com/en/ and Microsoft Visual C++ 2010 from http://www.microsoft.com/en-in/download/details.aspx?id=5555
- If you decide to download the latest version, screenshots and steps might differ in your lab environment.
- Run this lab in a Windows Server 2008 virtual machine
- Administrative privileges to run the tool
- A web browser with Internet access in both the machines

Lab Duration

Time: 45 Minutes

Overview of Https

SSL/TLS provides communication security and privacy over the Internet for applications such as web, email, instant messaging (IM), and virtual private networks (VPNs). Data flowing through the channel is encrypted and is difficult to decode.

Lab Tasks

TASK 1

Stop IIS Service and World Wide Web Publishing Service

Note: Before running this lab, log into **Windows Server 2008** and ensure that you stop the IIS admin service and World Wide Web Publishing Service (if you have the service installed on the machine.). To stop the service, go to **Start → Administrative Tools → Services**, right-click **IIS Admin Service** and click **Stop**, right-click **World Wide Web Publishing Service** and click **Stop**. Also ensure that you stop the Internet Information Services (IIS) Manager and Internet Information Services (IIS) 6.0 Manager. To stop Internet Information Services (IIS) Manager, go to **Start → Administrative Tools → Internet Information Services (IIS) Manager**, right-click on the server name in the left pane and click **Stop** to stop the manager. To stop the Internet Information Services (IIS) 6.0 Manager, go to **Start → Administrative Tools → Internet Information Services (IIS) 6.0 Manager**, right-click on the server name in the left pane and click **Disconnect** to disconnect the manager.

1. In **Windows Server 2008**, click the **Start** button on the lower left corner of the screen, and then click **start WampServer** to launch the WampServer application.

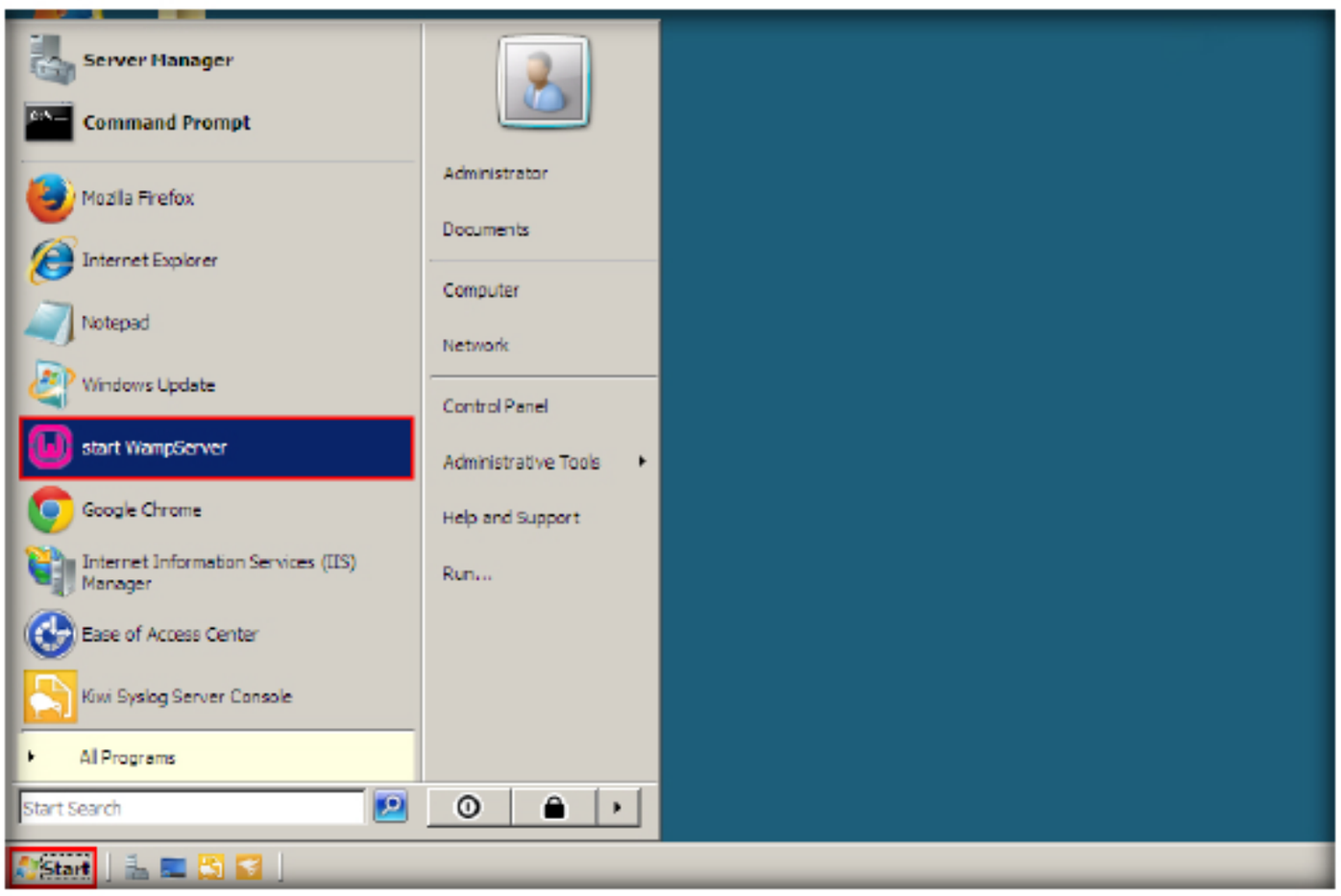

FIGURE 2.1: Attempting to browse on https

2. Open a web browser, type the URL https://localhost/cnd in the address bar, and press **Enter**.

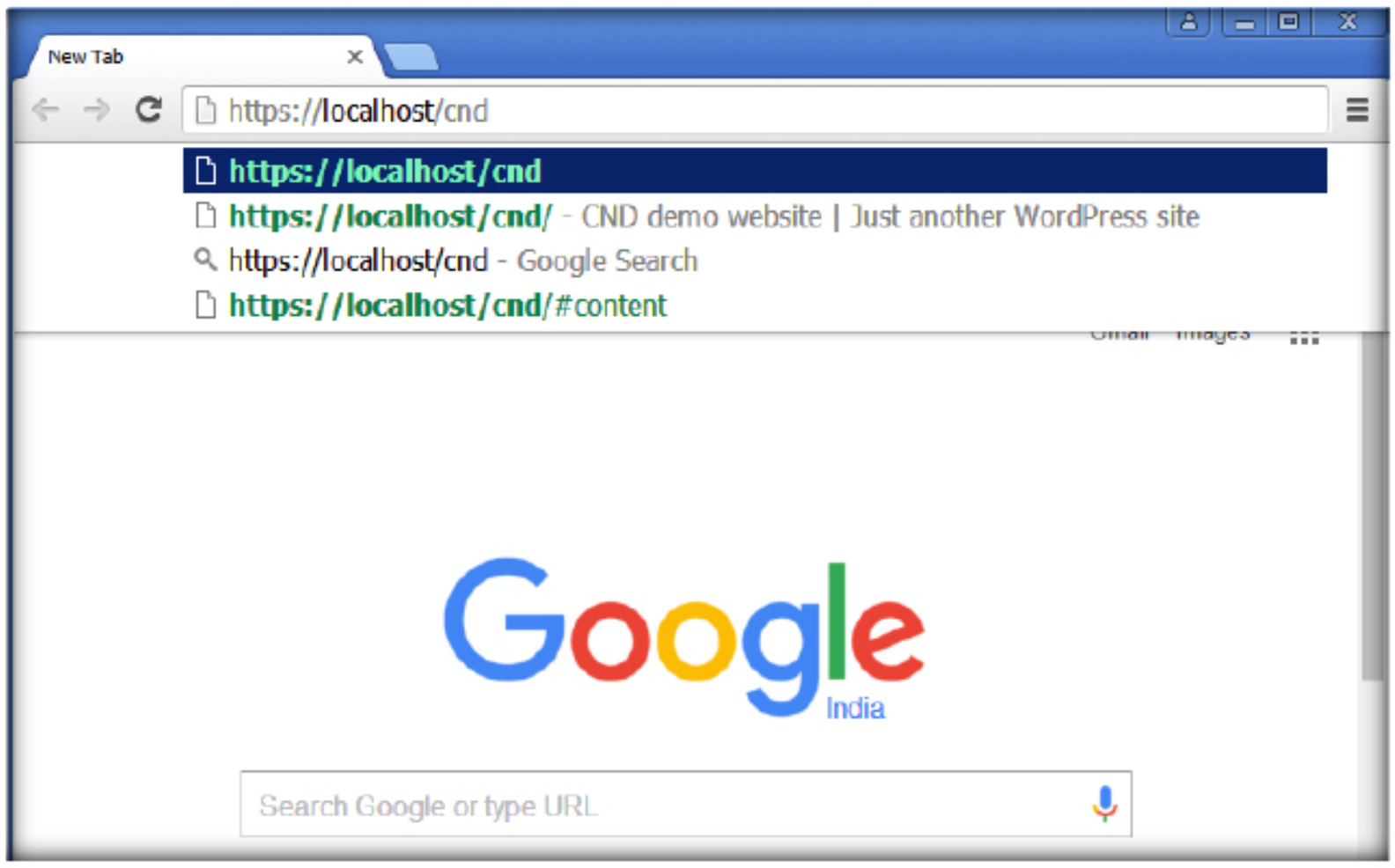

FIGURE 2.2: Attempting to browse on https

3. You won't be able to access the webpage, as SSL in not enabled on the server where the cnd WordPress website is deployed. So, to browse the cnd WordPress site over a secure channel (https/SSL), you need to enable SSL on WordPress.

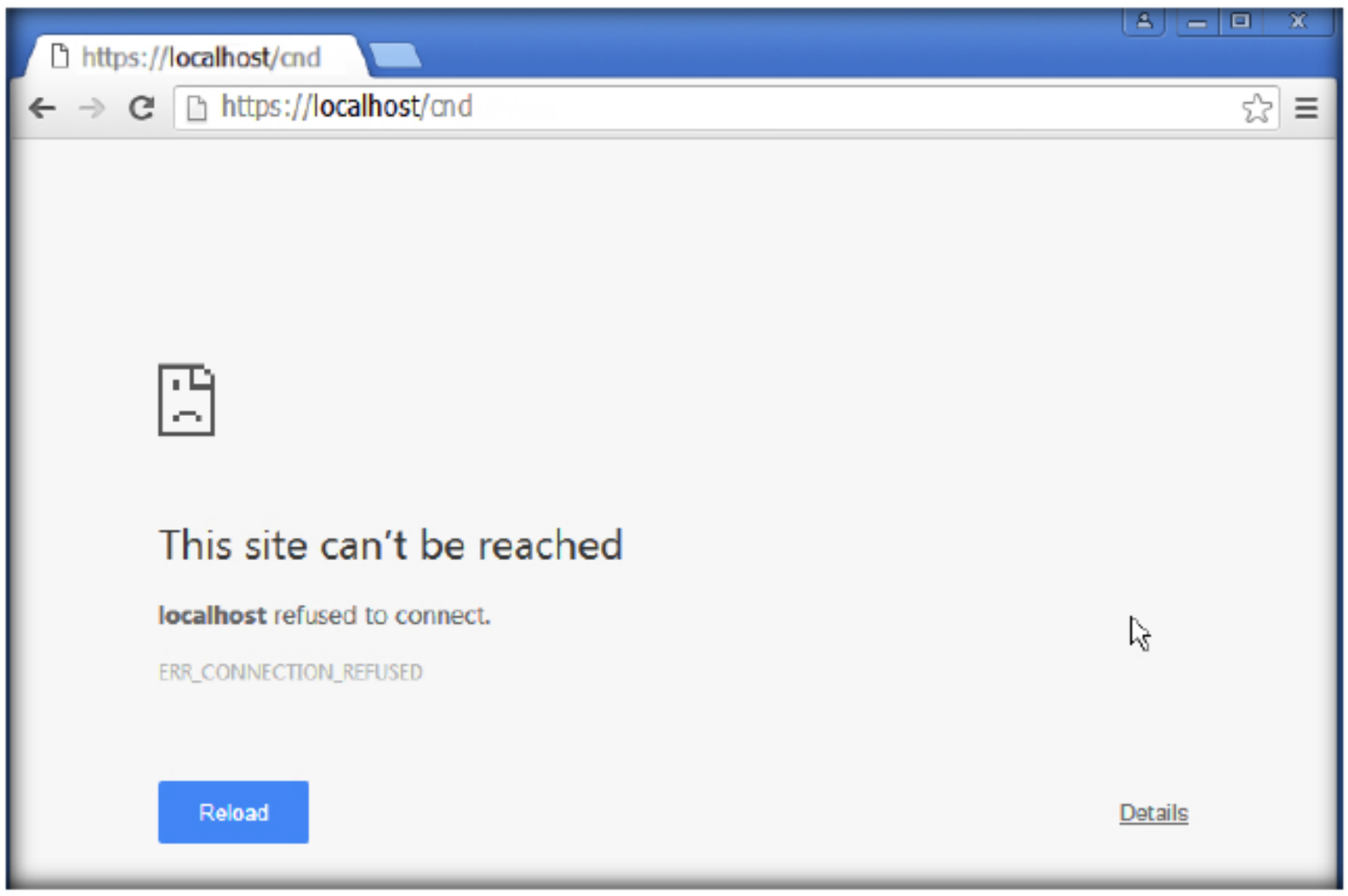

FIGURE 2.3: SSL disabled

4. Go to the **Start** menu, right-click **Computer,** and select **Properties** in the menu.

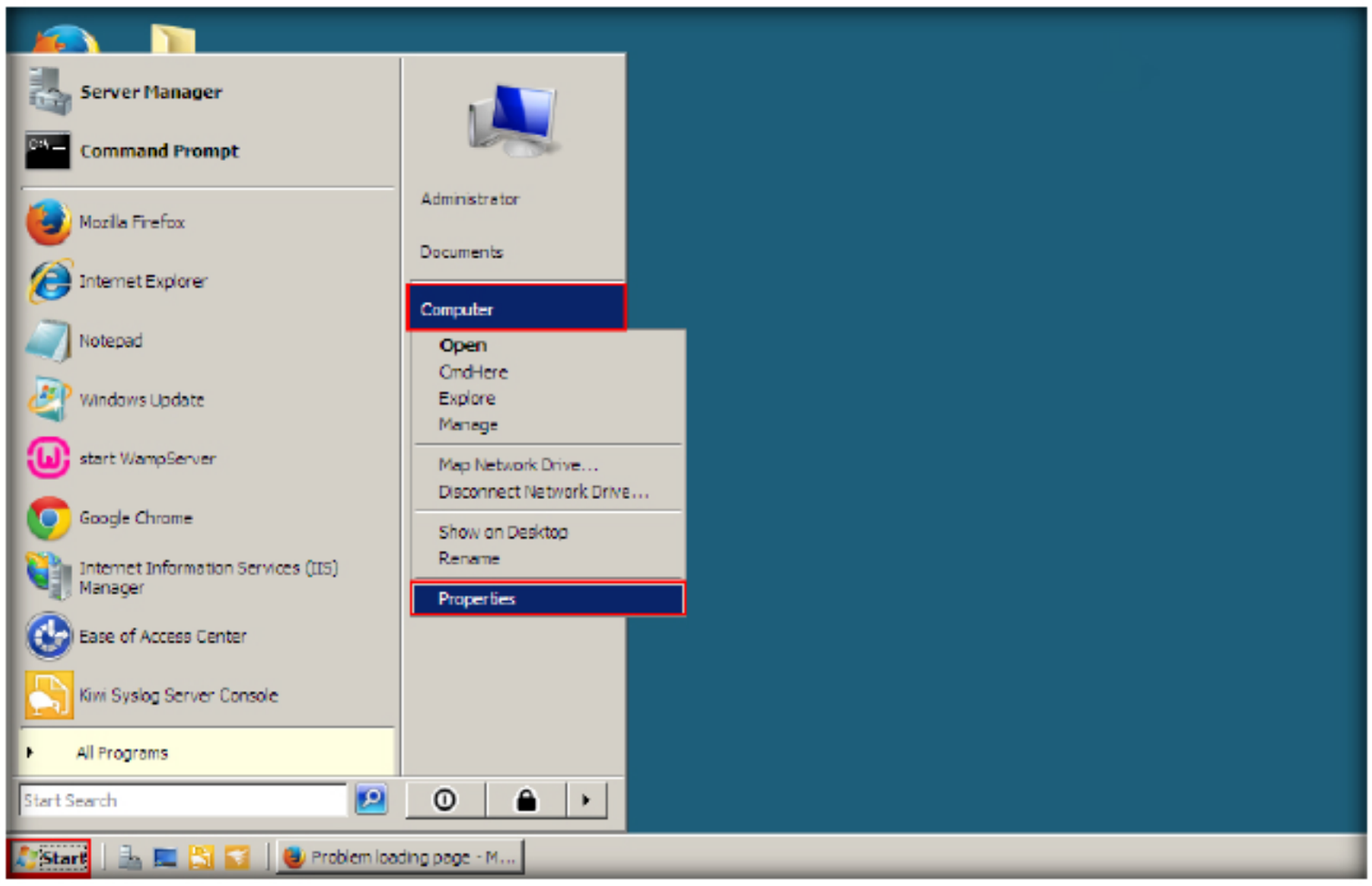

FIGURE 2.4: Selecting Computer Properties

TASK 2

Add an Environment Variable

5. **System Control Panel** appears on the screen, click the **Advanced system settings** link.

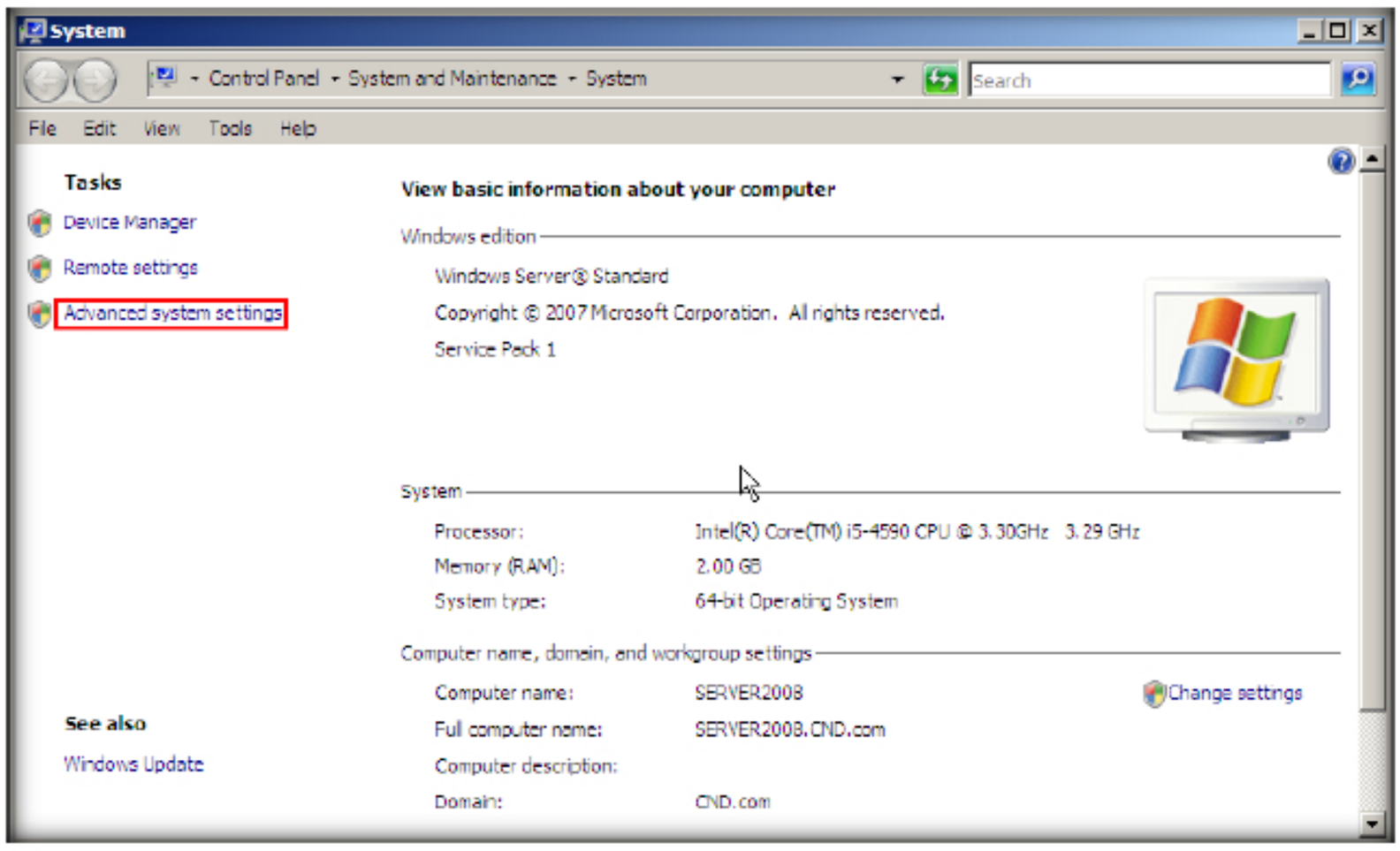

FIGURE 2.5: Advanced system settings

6. The **System Properties** window appears; go to the **Advanced** tab, and click **Environment Variables...**.

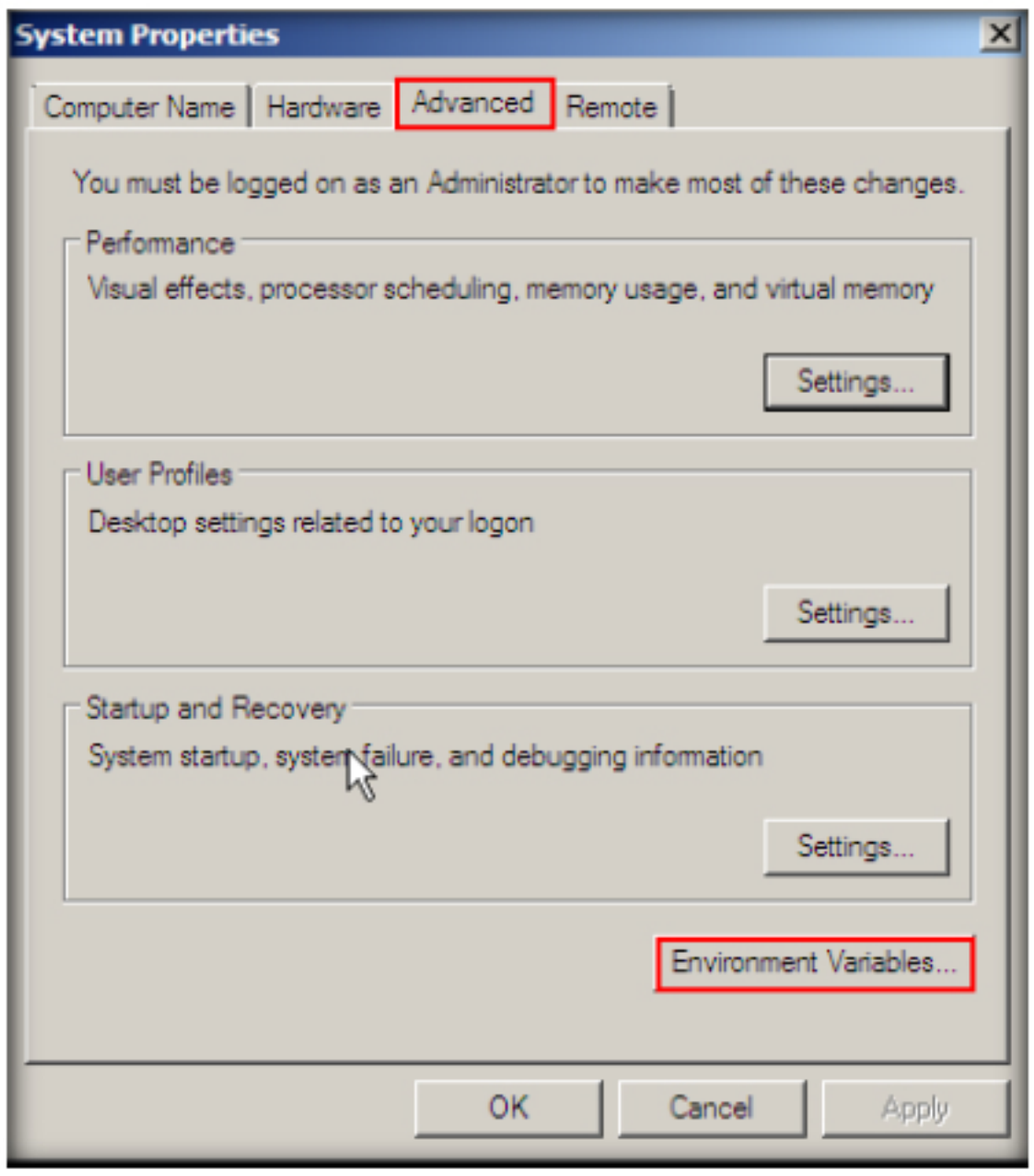

FIGURE 2.6: Selecting Environmental Variables

7. The **Environment Variables** window appears; click **New...**.

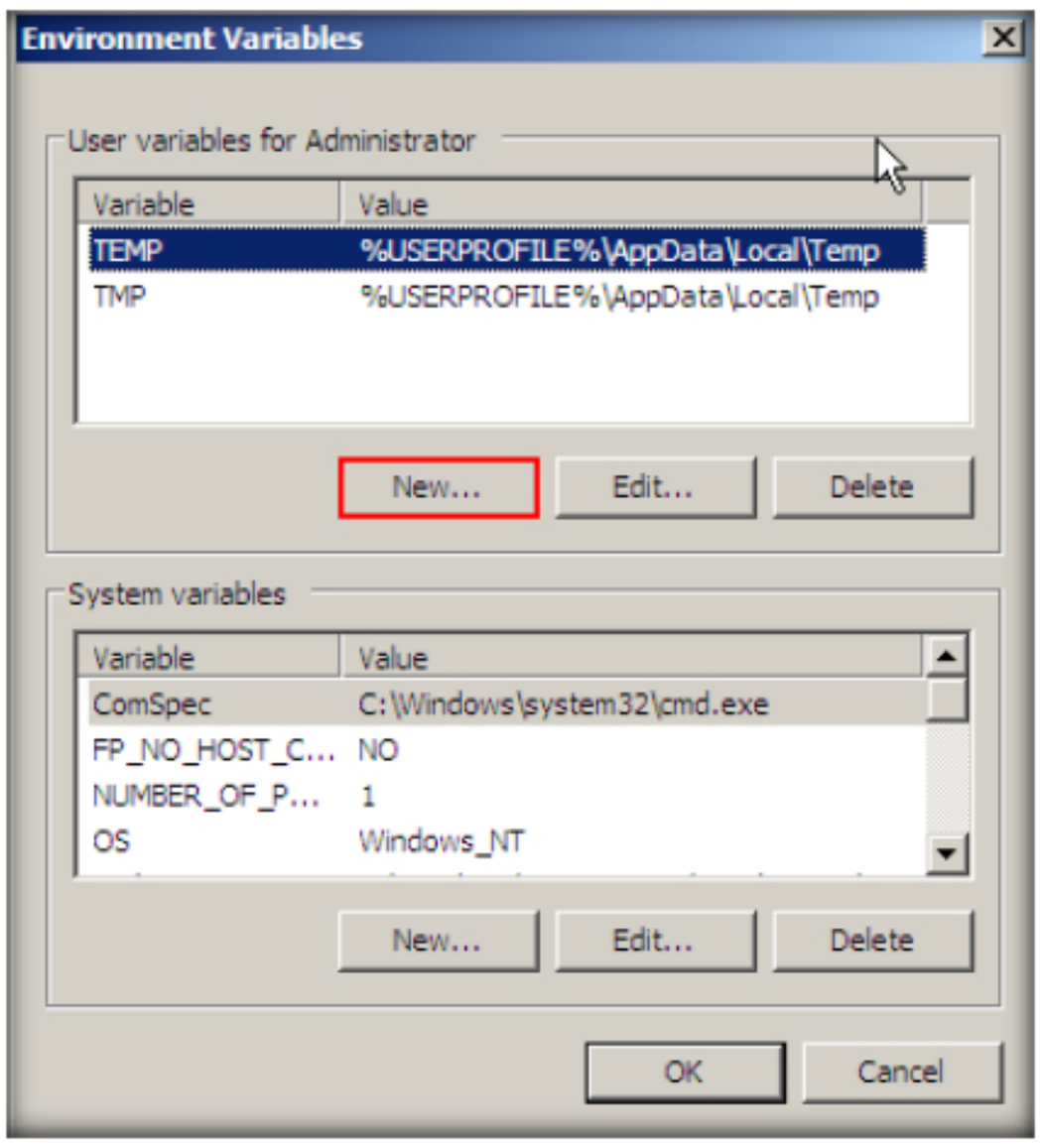

FIGURE 2.7: Adding a new environmental variable

8. In the **New User Variable** window, enter the variable name **openssl_conf**, enter the variable value **C:\Wamp\bin\apache\apache2.4.2\conf\openssl.cnf** and click **OK**.

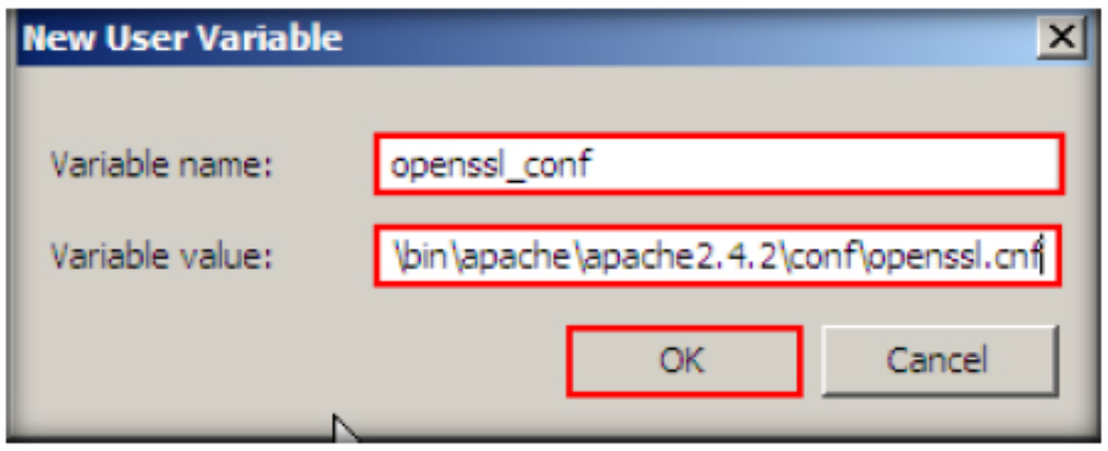

FIGURE 2.8: Adding a new environmental variable

9. Click **OK** in the **Environment Variables** window, and then click **OK** in the **System Properties** window.

TASK 3

Configure php.ini

10. Navigate to the location **C:\Wamp\bin\apache\apache2.4.2\bin** and open **php.ini** with **Notepad++**.
11. Uncomment the **line no. 970** by removing "**;**" before the code.

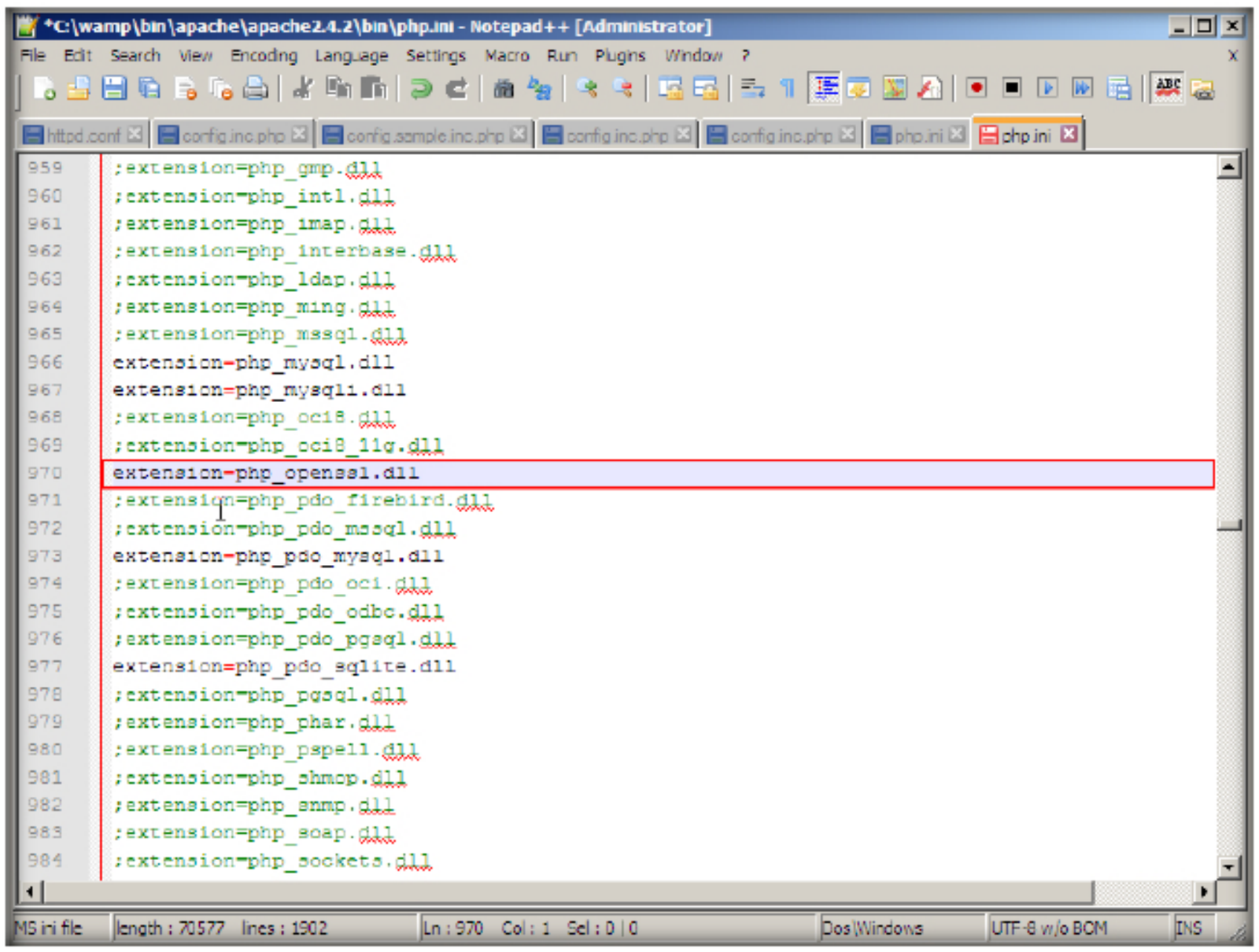

FIGURE 2.9: Enabling openssl.dll

12. **Save** the notepad file.

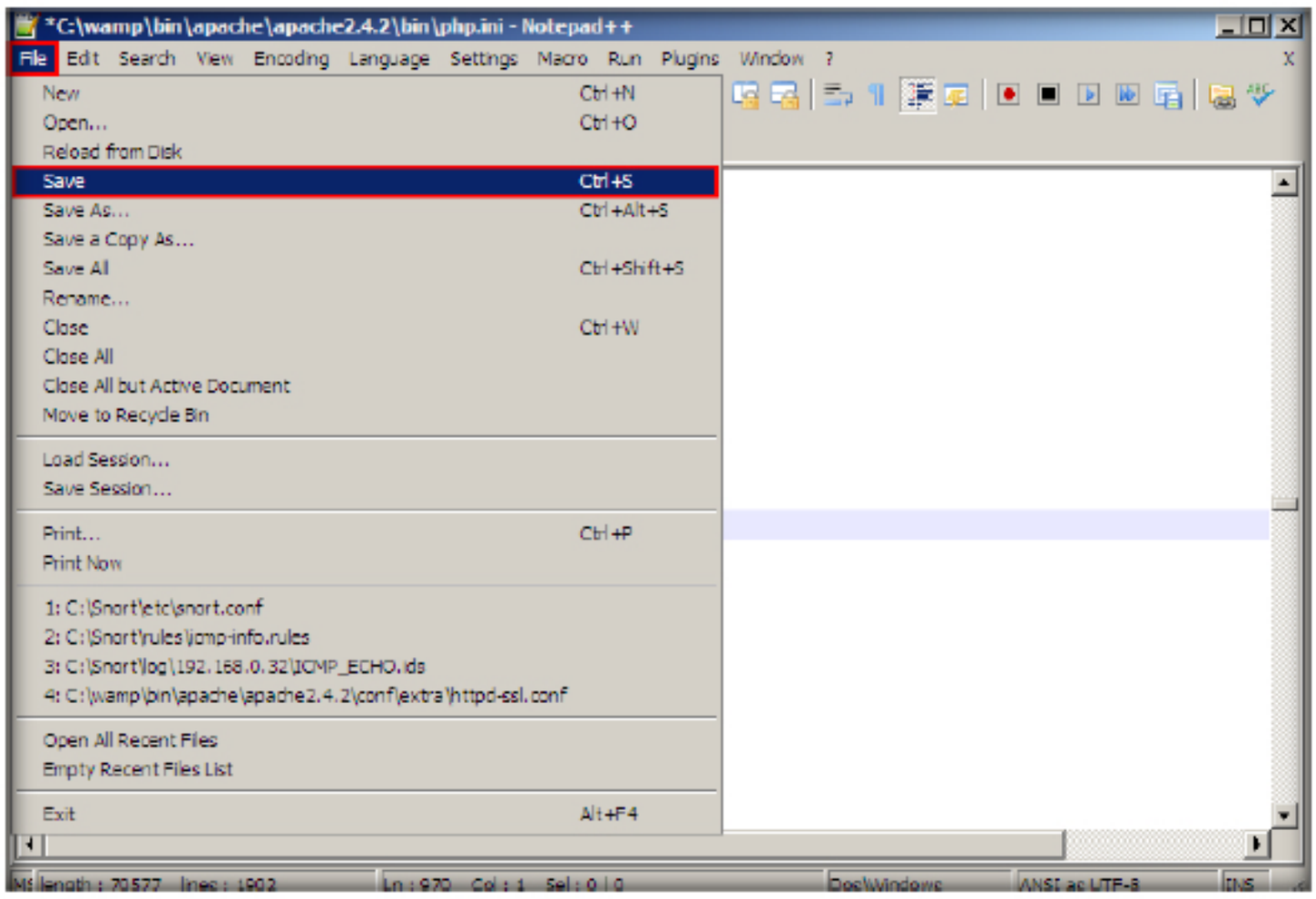

FIGURE 2.10: Saving the php.ini file

13. **Restart** the machine.

14. Click **Start,** and then click **WampServer**.

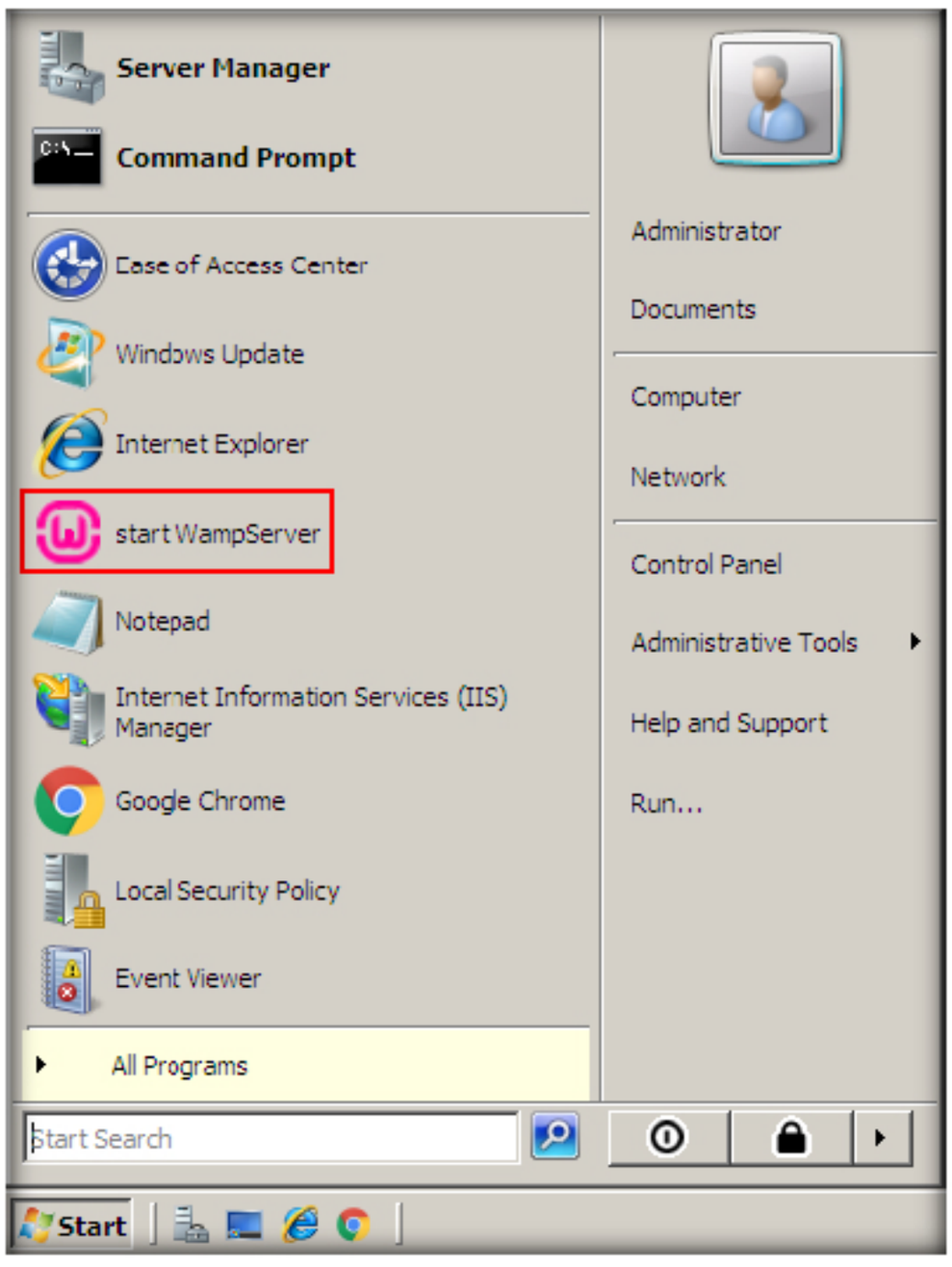

FIGURE 2.11: Starting WampServer

15. The **WampServer** icon appears in the notification area, as shown in the screenshot:

FIGURE 2.12: WampServer activated

Note: If the icon doesn't turn green, go to **Start → Administrative Tools → Internet Information Services (IIS) Manager**, right-click on the server name in the left pane, and click **Stop** to stop the manager. Then, click Wamp Server icon in the notification area, and select **Restart All Services**. You can even stop the World Wide Publishing Service.

TASK 4

Create a Private Key

16. Navigate to **C:\wamp\bin\apache\apache2.4.2**, press Shift + Right-click **bin** folder, and select **Open Command Window Here**.

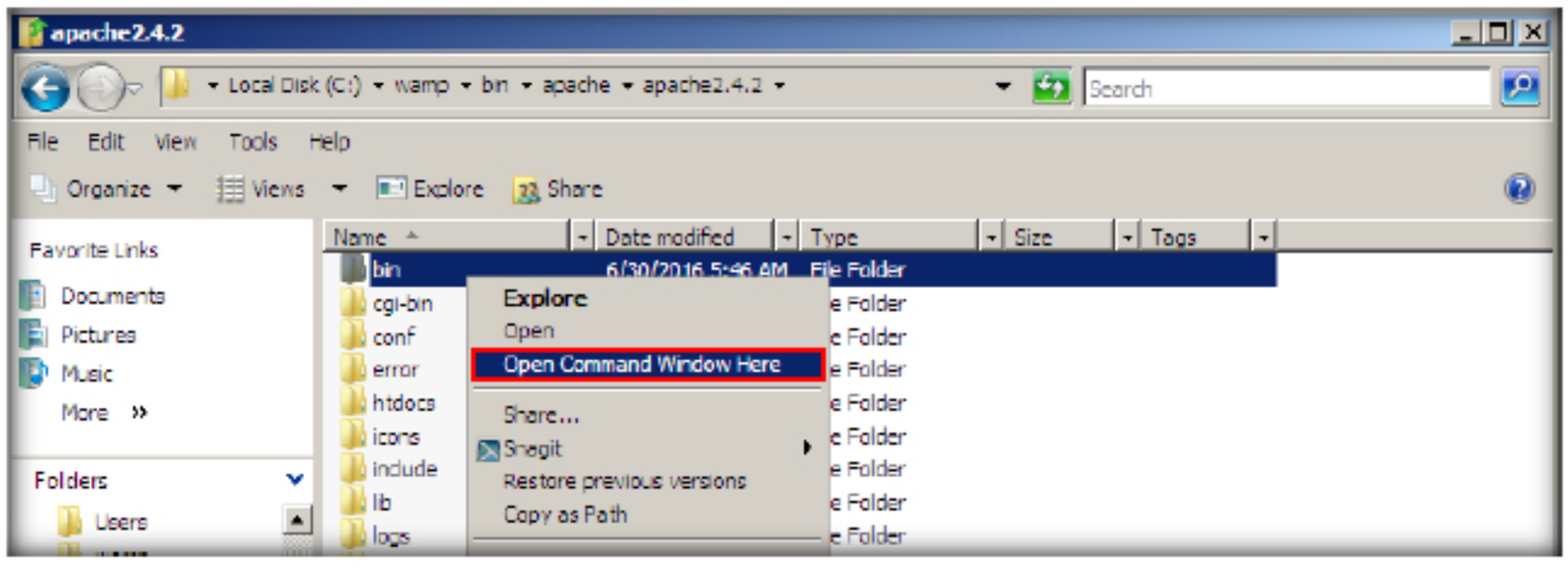

FIGURE 2.13: Launching command window Here

17. The command prompt appears, pointing to the directory location **C:\wamp\bin\apache\apache2.4.2\bin**.

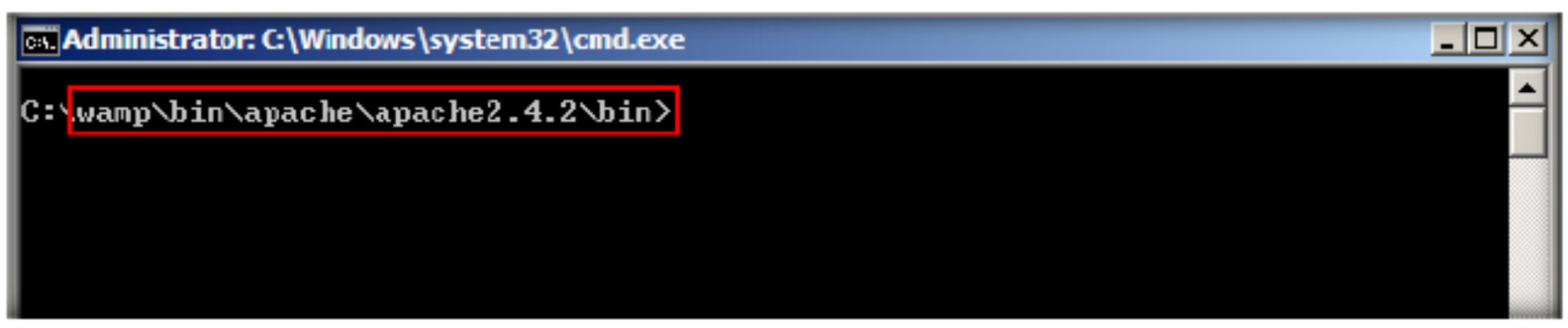

FIGURE 2.14: wamp bin

18. Type **set openssl_conf - C:\wamp\bin\apache\apache2.4.c\conf\openssl.cnf** and press **Enter**.

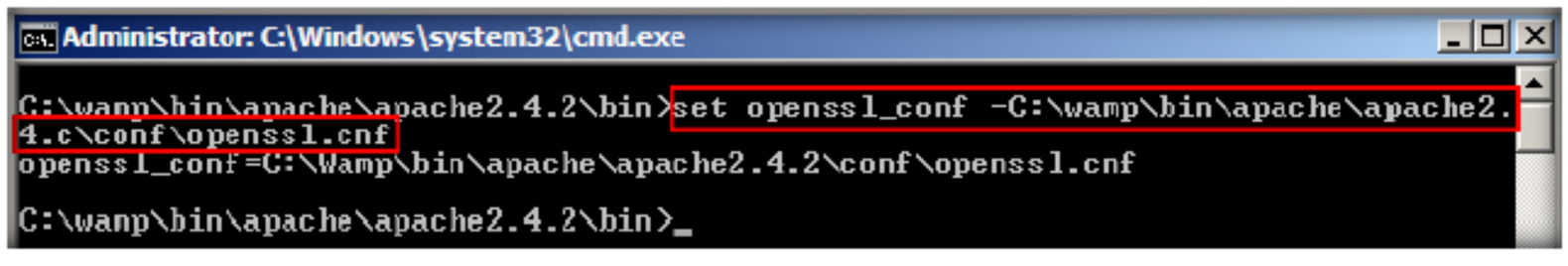

FIGURE 2.15: Setting environment variable to openssl.cnf

19. Now the environment variable is set to **openssl.cnf**.
20. Type **openssl genrsa -des3 -out server.key 1024** and press **Enter** to create a server private key named **server** with **1024** bit encryption.
21. You will be asked to enter a pass phrase (password) for the generated key. Type a password of your choice and press **Enter**.
22. In this lab the password entered is **qwerty@123**.
23. You will be asked to re-enter the same password for the purpose of verification. So, retype the password and press **Enter**.

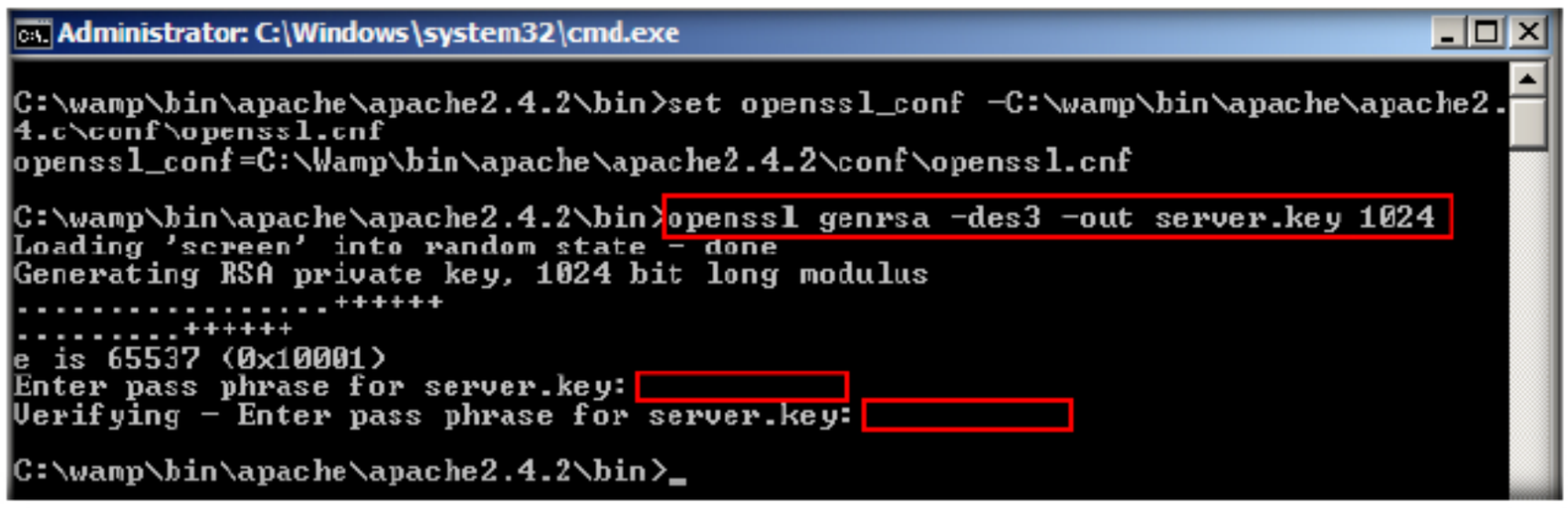

FIGURE 2.16: Creating a server private key

24. Apache for windows does not support private keys that are password protected, so you need to remove the pass phrase from the RSA private key.
25. Type **openssl rsa -in server.key -out server.pem** and press **Enter**.
26. You will be asked to enter the pass phrase for the **server.key**. Type the password you have assigned in **step 24** (here, **qwerty@123**), and press **Enter**.

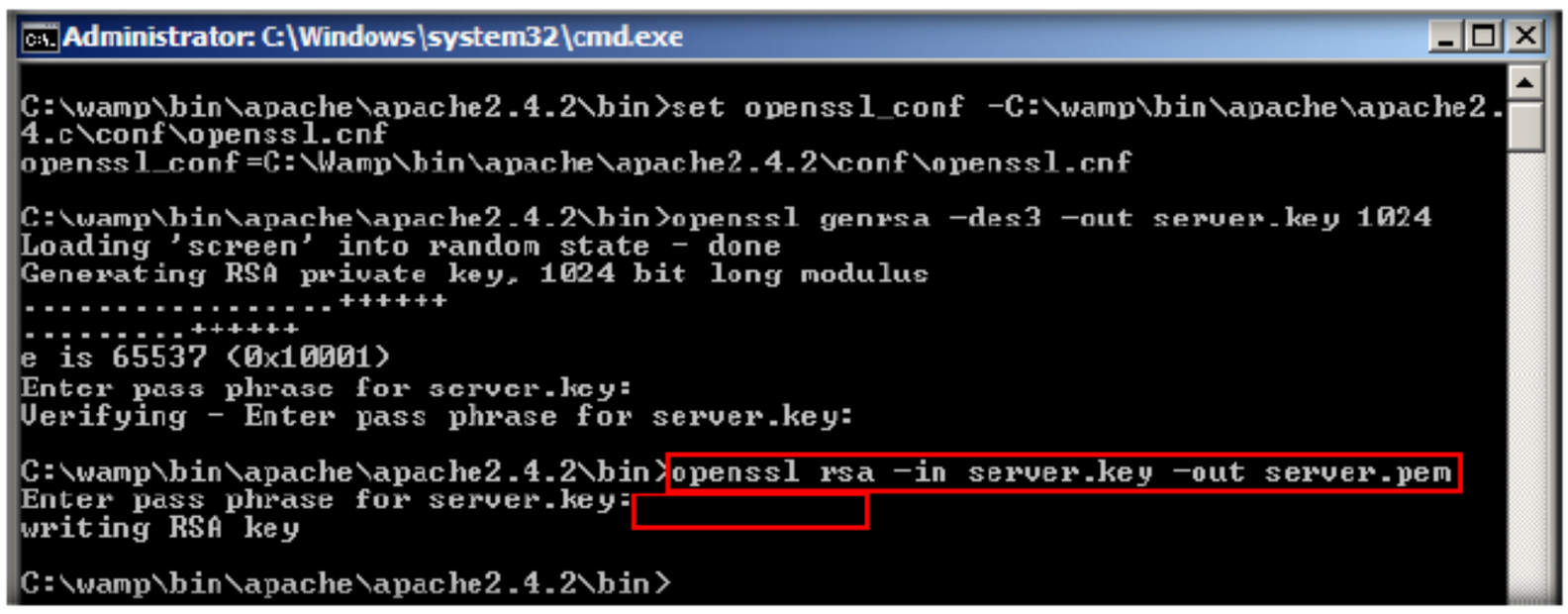

FIGURE 2.17: Removing pass phrase from the RSA private key

TASK 5

Create a self-signed Certificate

27. Type **openssl req -new -key server.key -out server.csr** and press **Enter**.
28. Type the passphrase you have assigned in **step 24** (**qwerty@123**) for the private key (server.key), and press **Enter**.

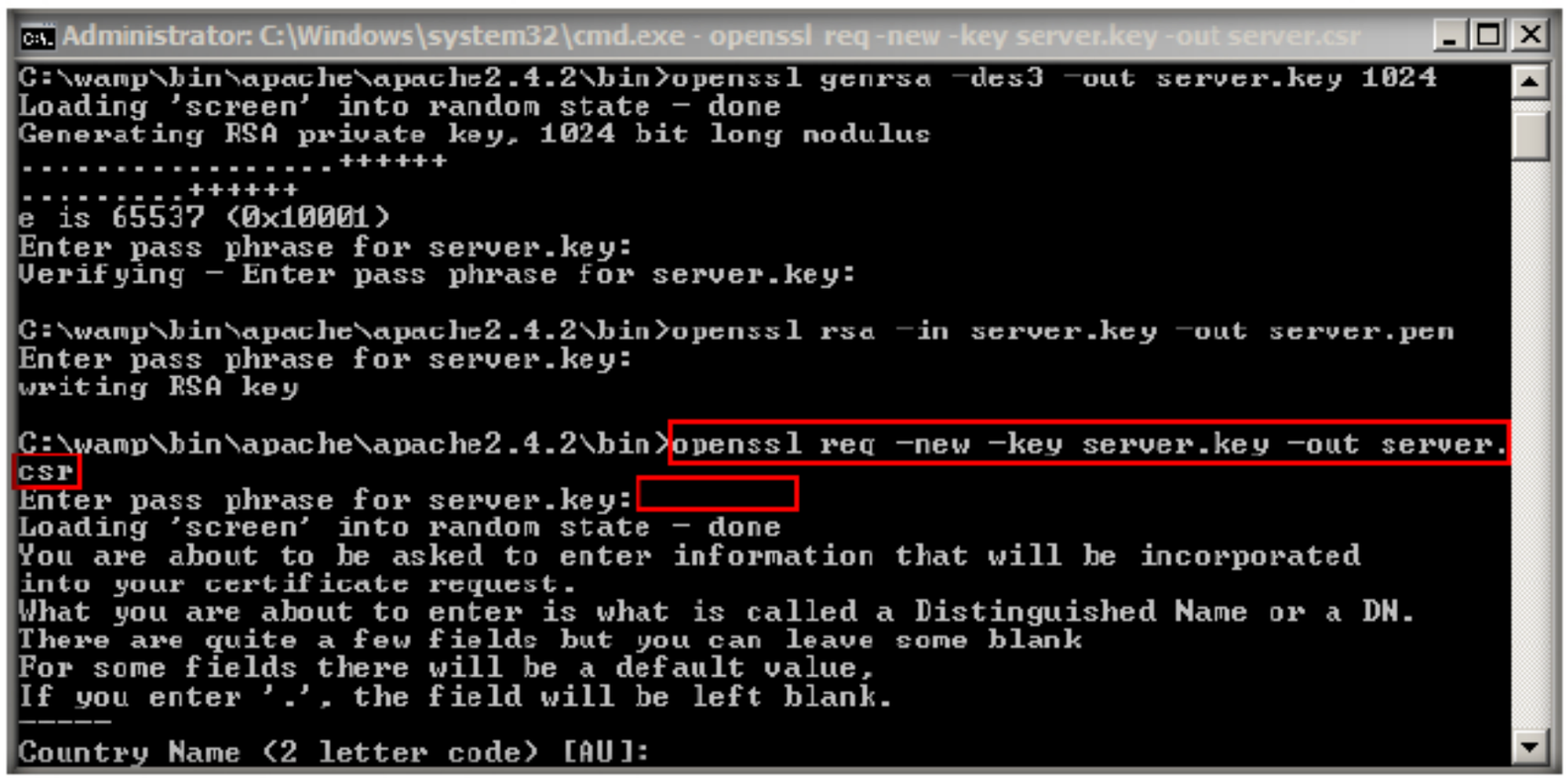

FIGURE 2.18: Removing pass phrase from the RSA private key

29. You will be asked to enter information such as your country, state, city, etc. Fill in your details in the respective fields. The information you provide in these fields will be incorporated into your certificate request.

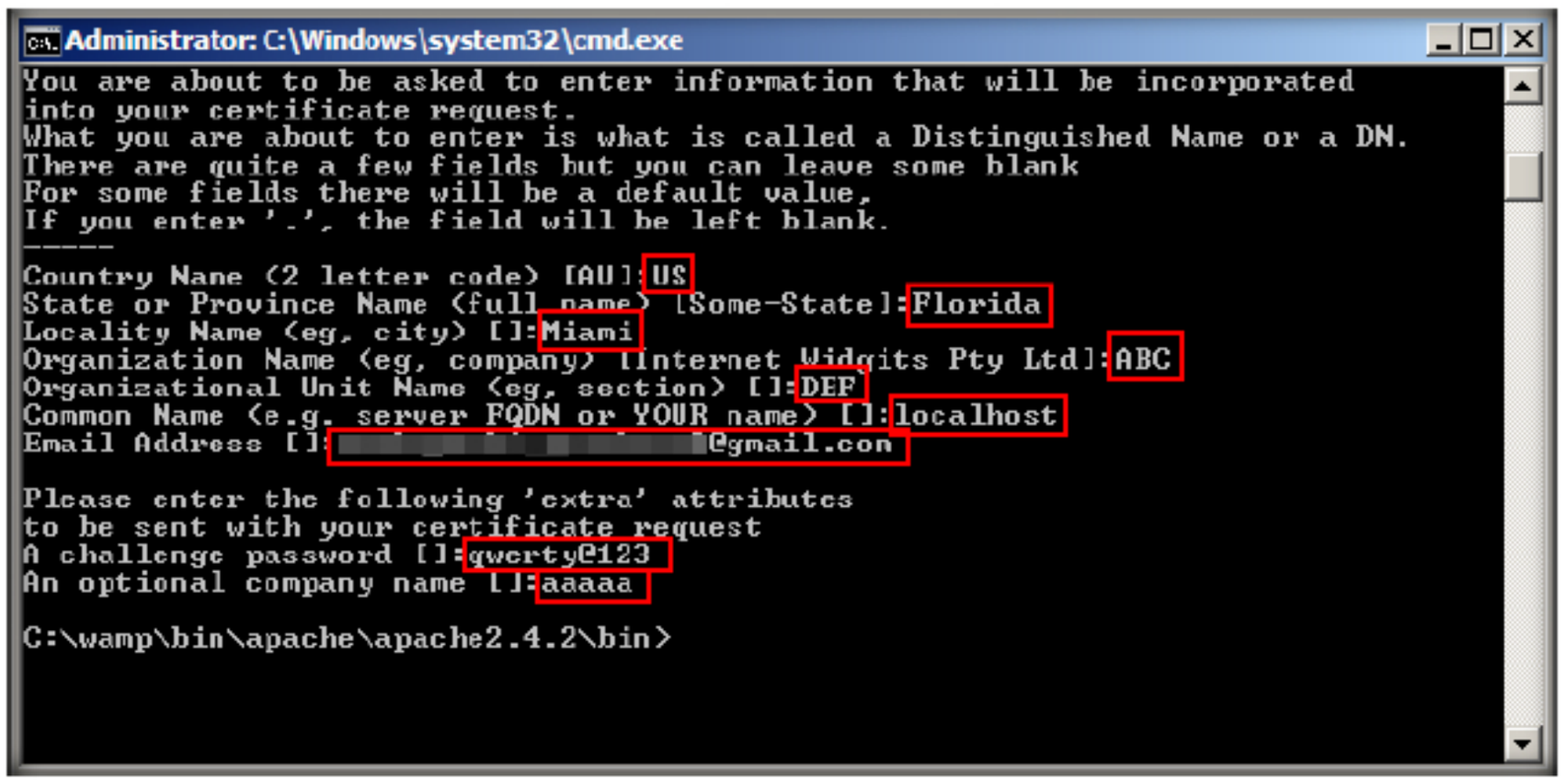

FIGURE 2.19: Assigning certificate value

30. Type **openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt** and press **Enter**.

31. Type the pass phrase (**qwerty@123**) for server.key, and press **Enter**.

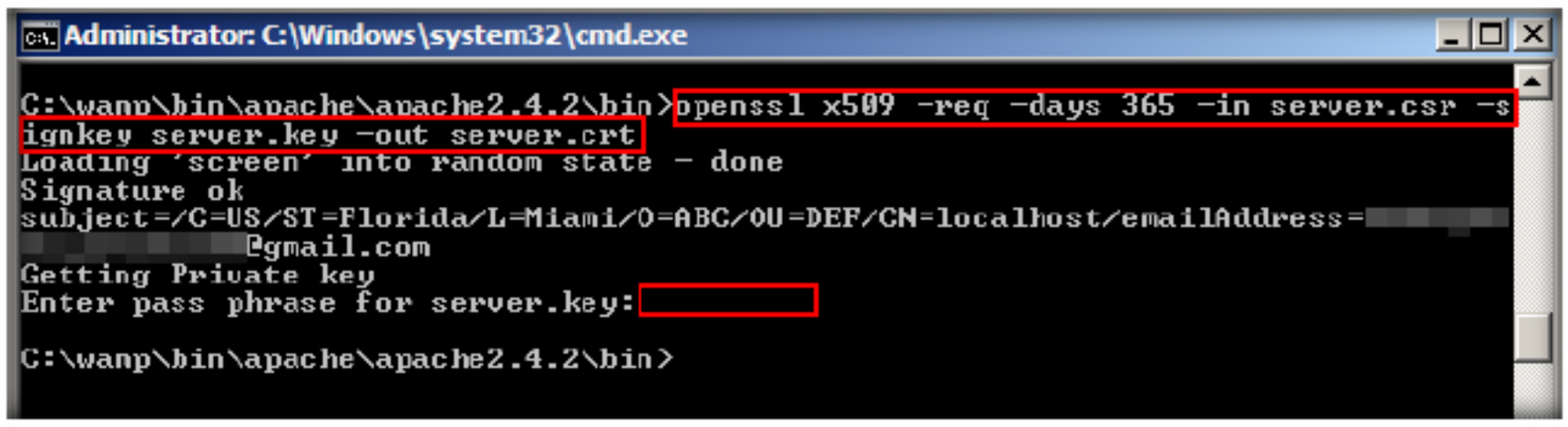

FIGURE 2.20: Entering the passphrase

32. All the keys have been successfully created. These can be viewed in the location **C:\wamp\bin\apache\apache2.4.2\bin**.

FIGURE 2.21: Keys created successfully

33. Create a directory named **ssl** in **C:\Wamp\bin\apache\apache2.4.2\conf** and move all the created keys from **C:\Wamp\bin\apache\apache2.4.2\bin** to **C:\Wamp\bin\apache\apache2.4.2\conf\ssl**.

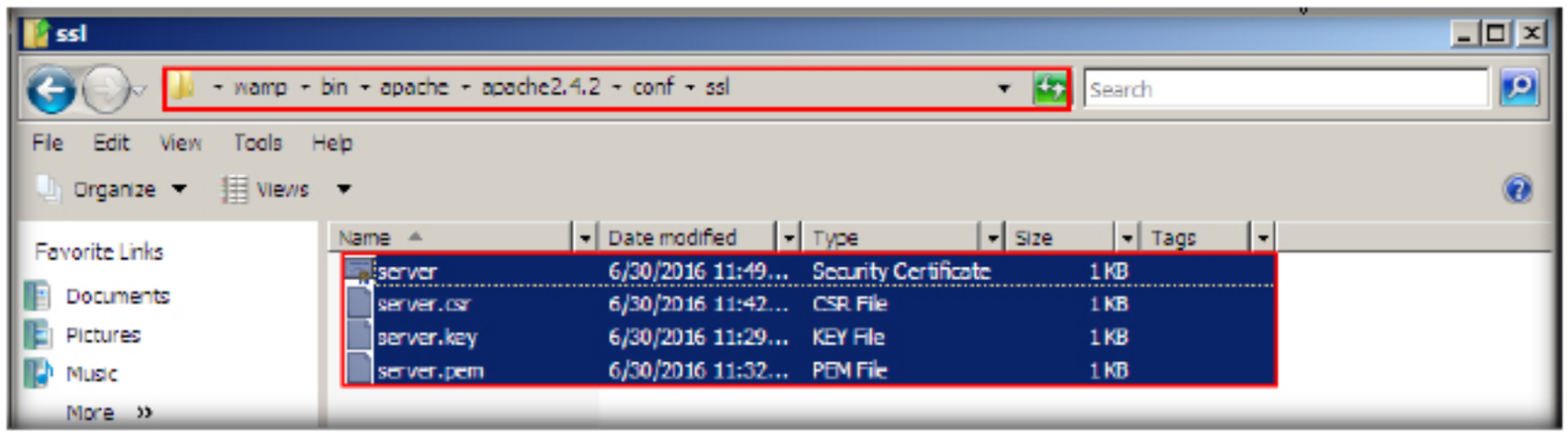

FIGURE 2.22: Copying the files to ssl directory

TASK 6

Configure ssl_module

34. Click **WampServer** icon from the notification area, and select **Apache → Apache modules → ssl_module**.

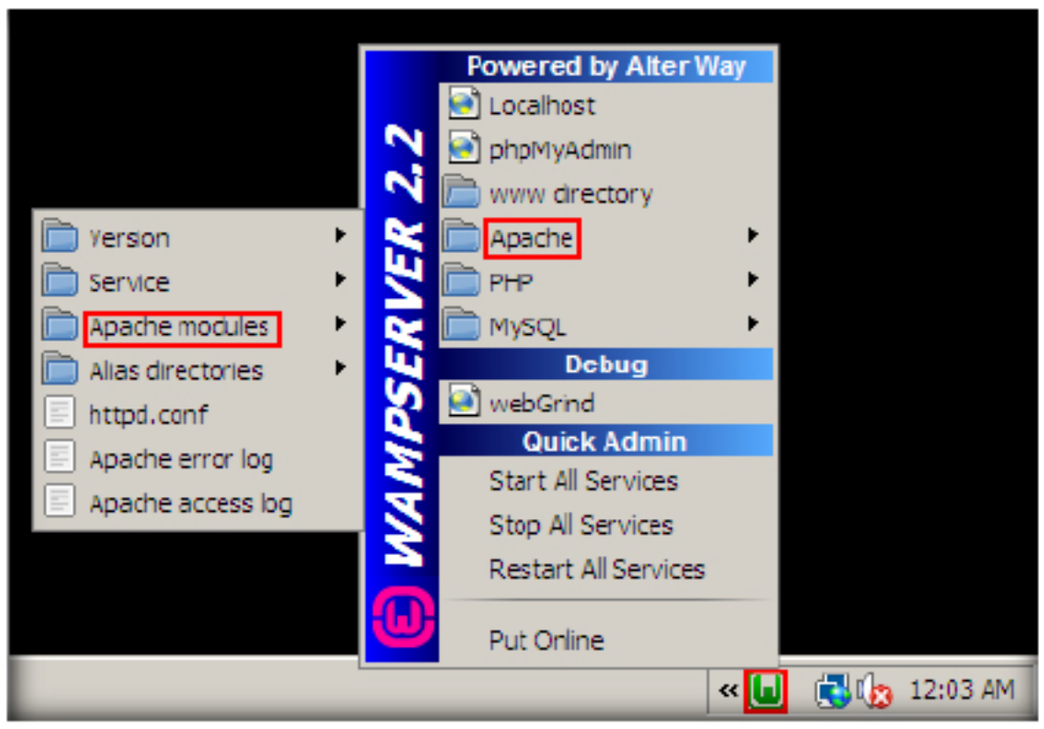

FIGURE 2.23: Selecting Apache modules

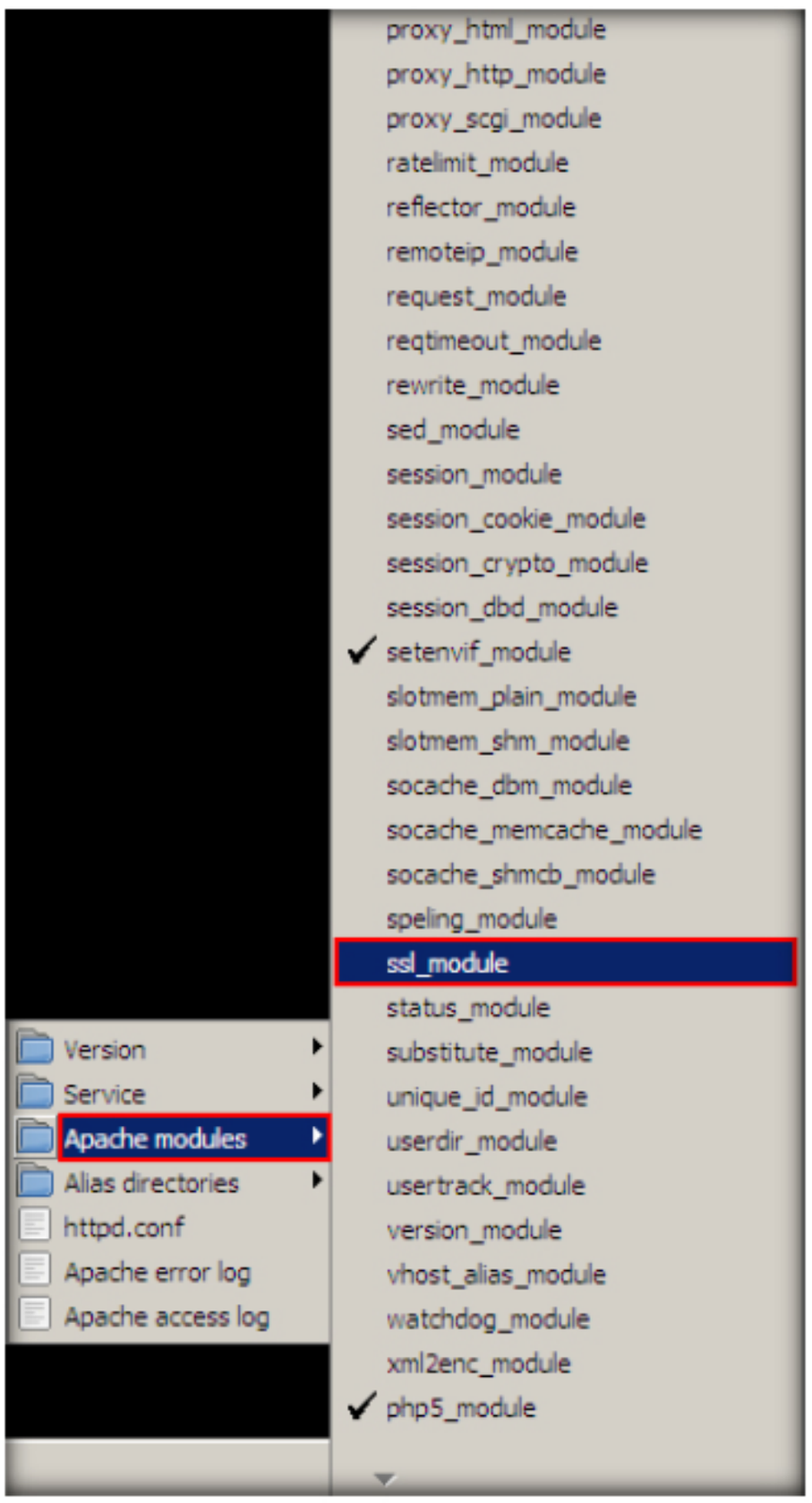

FIGURE 2.24: selecting ssl_module

35. Wampserver restarts as soon as you select **ssl_module**.
36. Navigate to **C:\wamp\bin\apache\apache2.4.2\conf\extra** and open **httpd-ssl.conf** with Notepad++.
37. Scroll down to **line 39** to view the port on which apache is listening. Ensure that the port number is **443**.

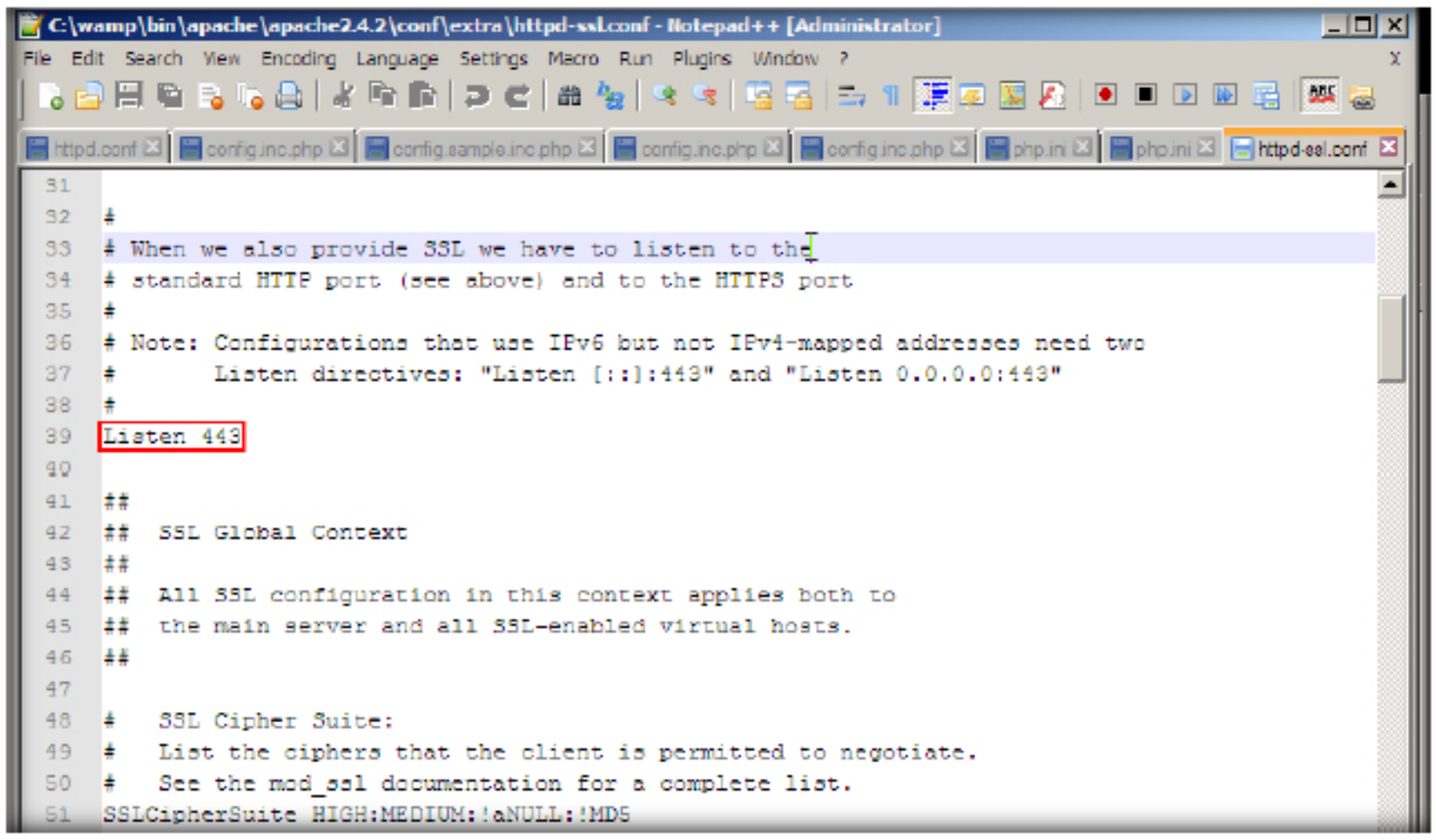

FIGURE 2.25: Viewing the port number

38. Scroll down to **line 76** and comment the line by adding **#** before the code.

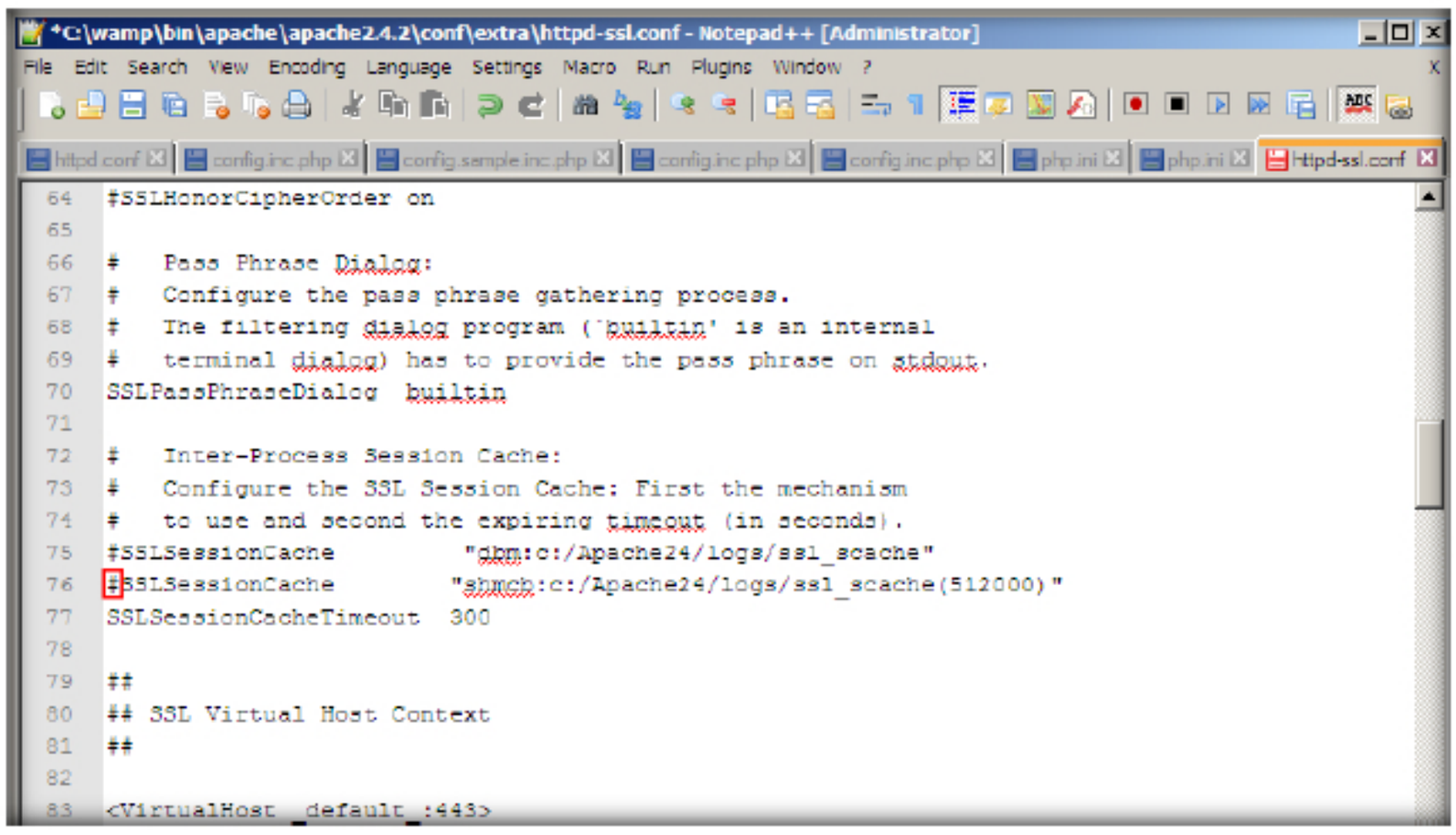

FIGURE 2.26: Editing ssl.conf

39. Scroll down the file and:
    a. In **line 86**, change the path of **DocumentRoot** to **"C:/wamp/www/"**
    b. In **line 87**, change the **ServerName** to **localhost:443**

c. In **line 89**, change the path of **ErrorLog** to **"C:/wamp/logs/ssl_error.log"**

d. In **line 90**, change the path of **TransferLog** to **"C:/wamp/logs/ssl_access.log"**

40. Also ensure that the **SSLEngine** is **on** in **line 94**.

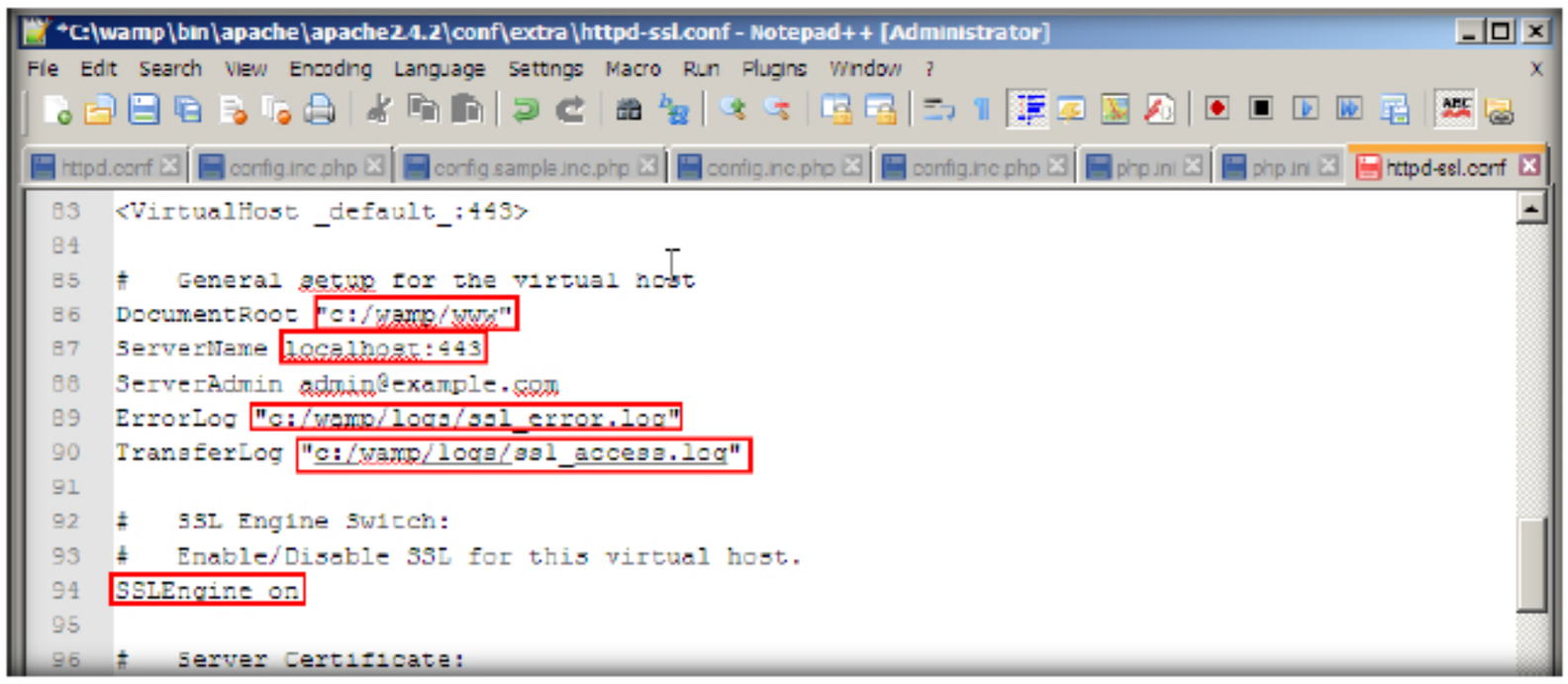

FIGURE 2.27: Editing ssl.conf

41. In **line 106**, change the path of **SSLCertificateFile** to **"C:/wamp/bin/apache/apache2.4.2/conf/ssl/server.crt"**

42. In **line 116**, change the path of **SSLCertificateKeyFile** to **"C:/wamp/bin/apache/apache2.4.2/conf/ssl/server.pem"**

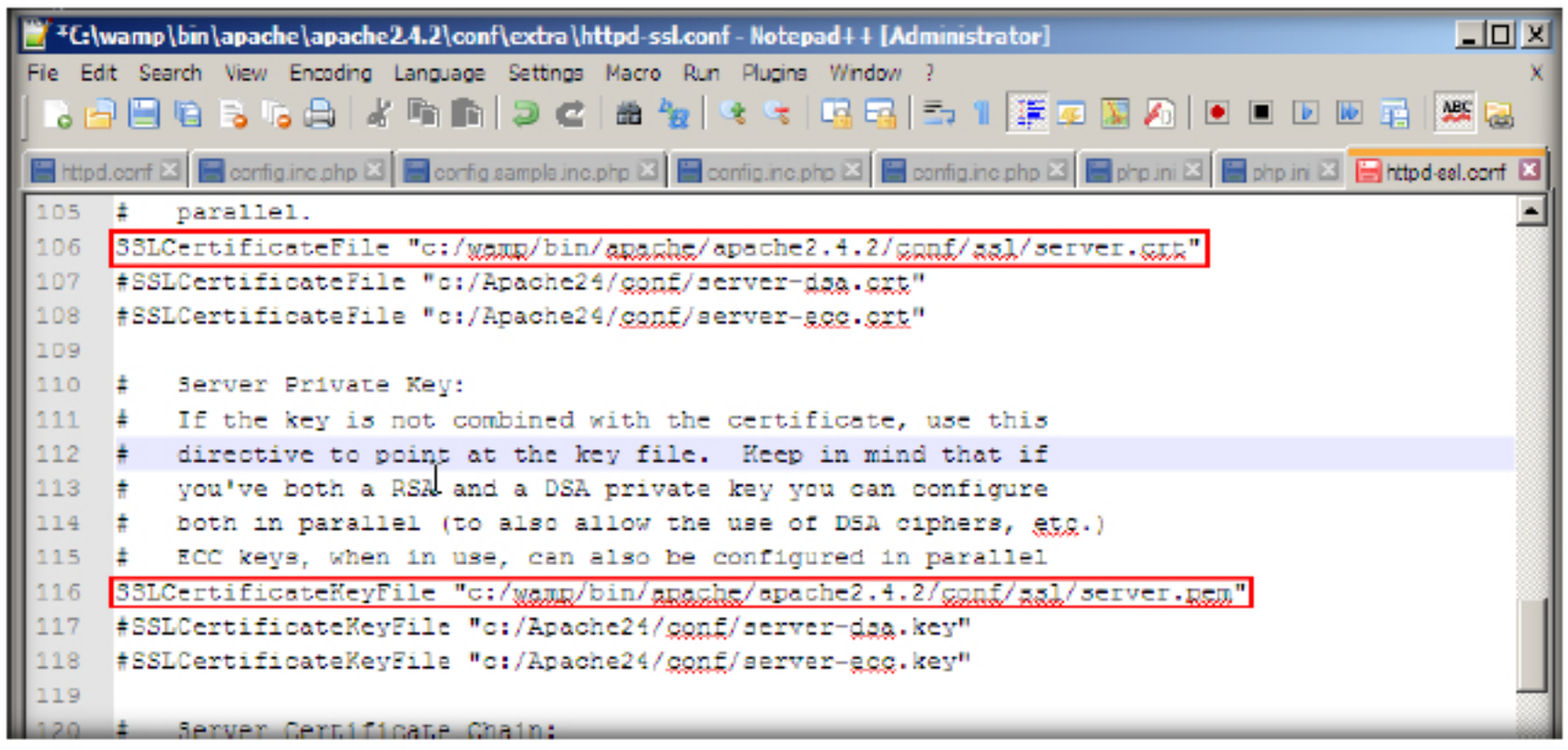

FIGURE 2.28: Editing ssl.conf

43. In **line 206**, change the Directory location to **"C:/Wamp/www/"**.

44. Delete **</Directory>** from the **line 208**.

45. Add the following lines:

    a) line 208: **options Indexes FollowSymLinks MultiViews**

    b) line 209: **AllowOverride All**

    c) line 210: **Order allow,deny**

    d) line 211: **allow from all**

    e) line 212: **</Directory>**

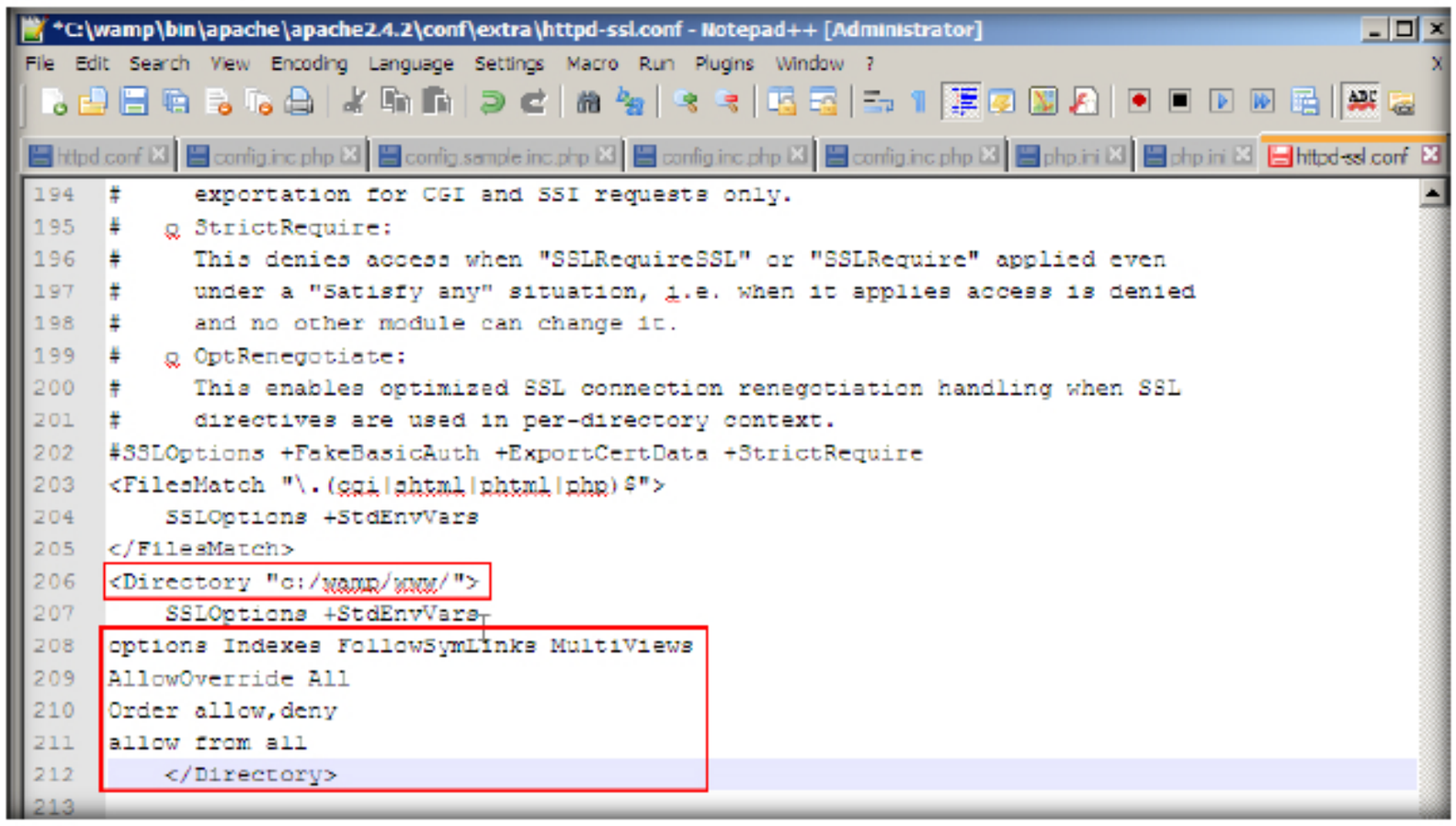

FIGURE 2.29: Editing ssl.conf

46. In **line 245**, change the **CustomLog** path to **"C:/wamp/logs/ssl_request.log"**

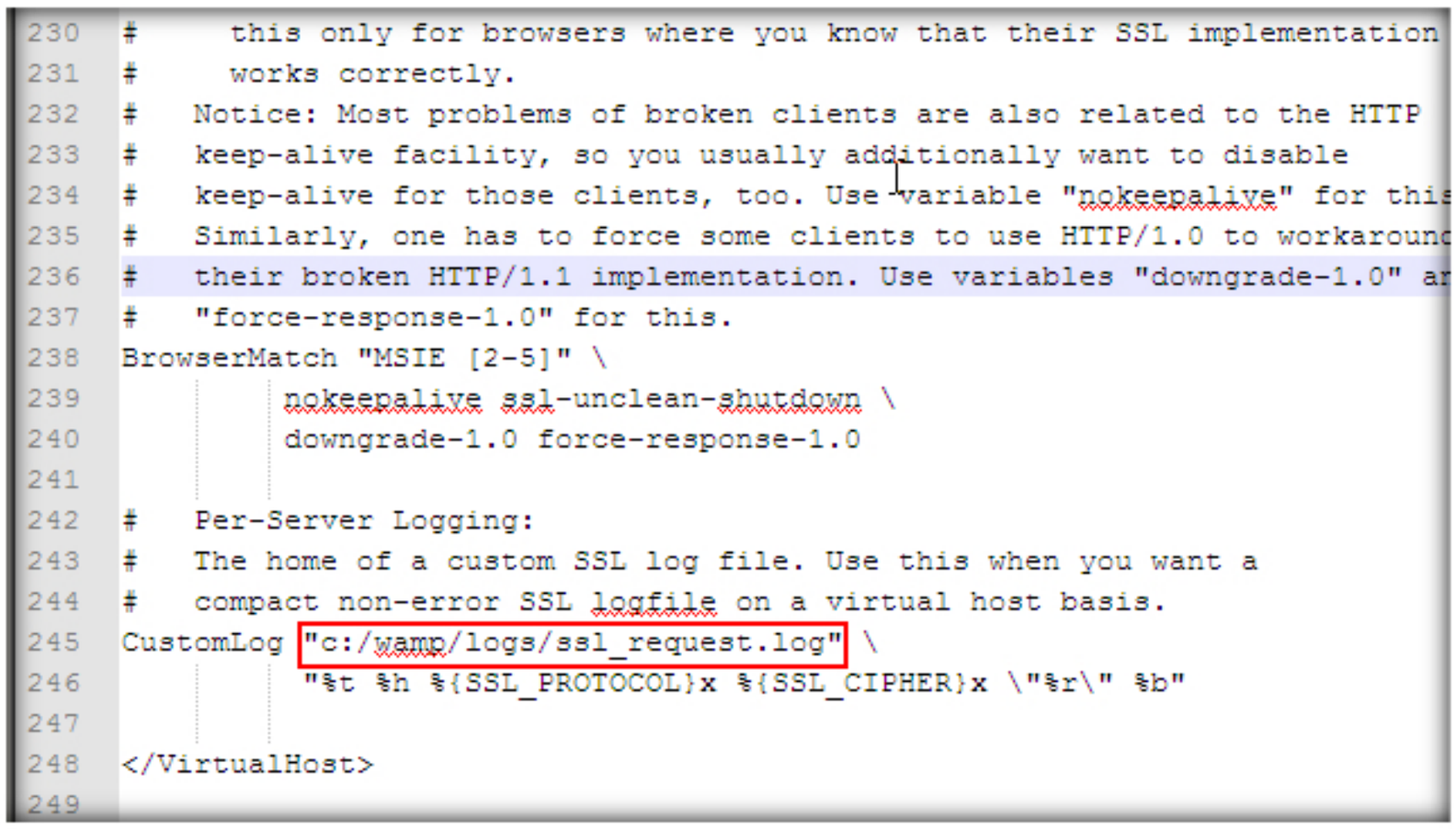

FIGURE 2.30: Editing ssl.conf

47. Save the file.

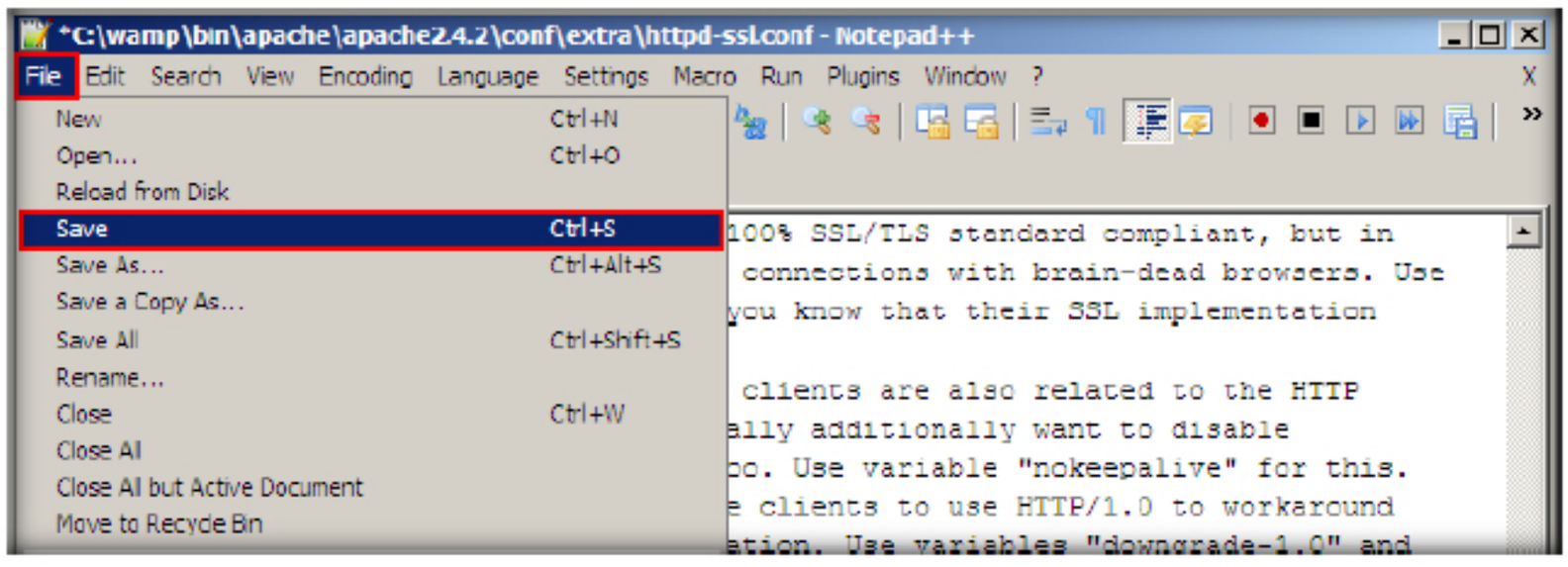

FIGURE 2.31: Saving the file

48. Navigate to **C:\wamp\bin\apache\apache2.4.2\conf,** and open **httpd.conf** file with Notepad++, uncomment **line 511** by removing "**#**" before the code in the line.

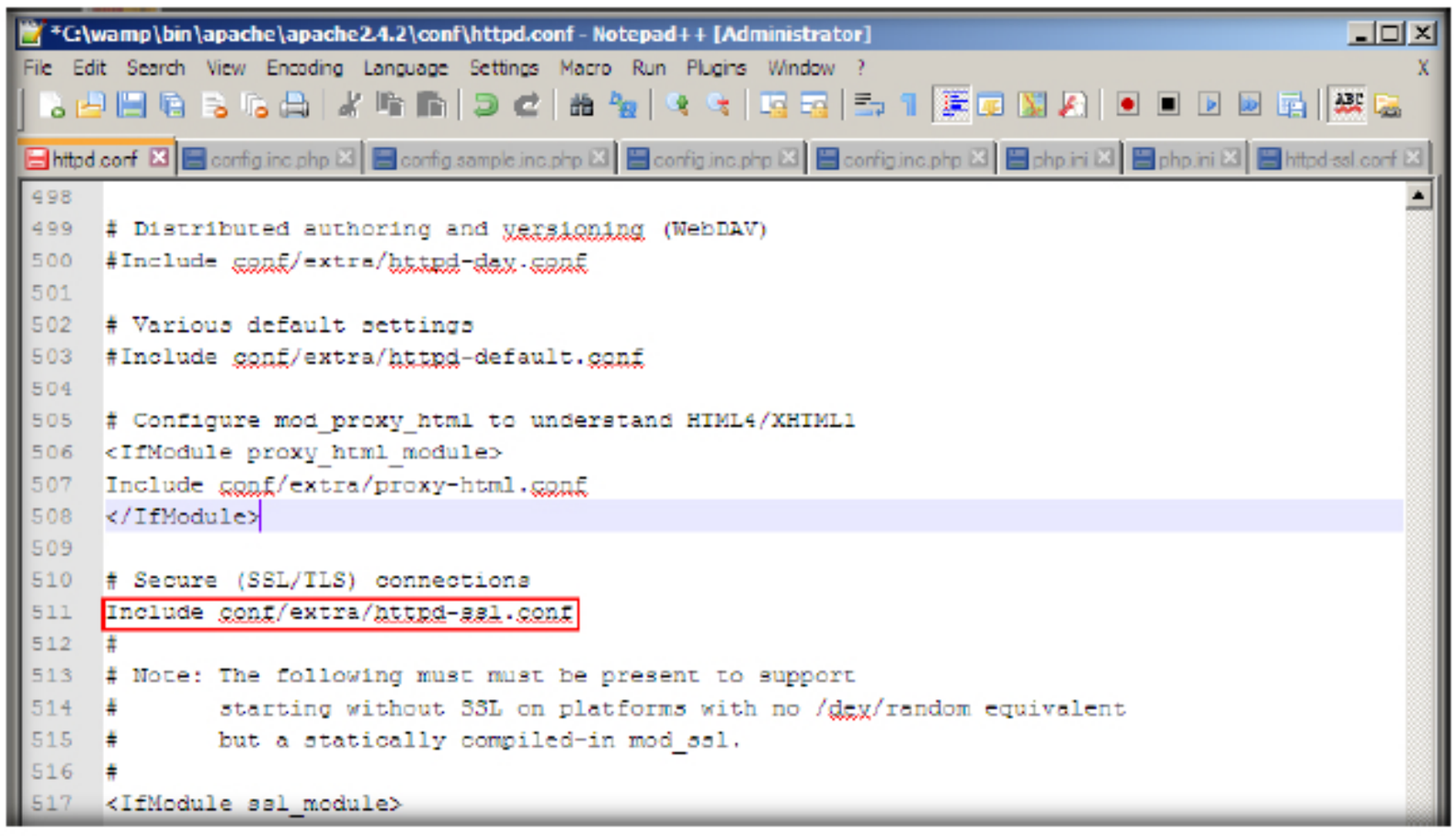

FIGURE 2.32: Saving the file

49. Click **File** from the menu bar, and click **Save**.
50. Navigate to **C:\wamp\bin\apache\apache2.4.2**, press Shift + Right-click **bin** folder, and select **Open command window here**.
51. In the command prompt, type **httpd -t** and press **Enter**. If all the syntax you entered is correct, it returns a message stating **Syntax OK**. This command lets you know if there are any syntax errors. Repeat the procedure until the command returns the message **Syntax OK**.

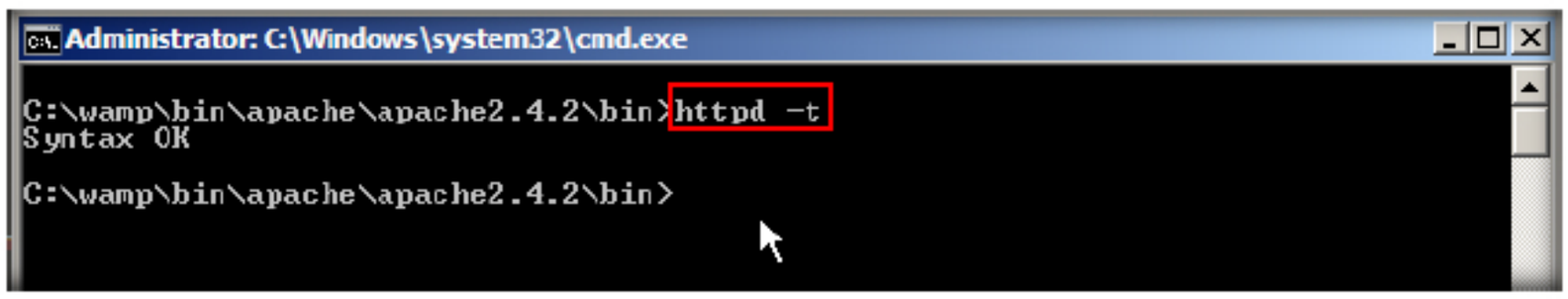

FIGURE 2.33: Checking for syntax errors

52. Click the **WampServer** icon in the notification area, and click **Restart All Services**. Wait until the icon turns **green**.
53. Launch a command prompt, type the command **netstat -an | more** and press **Enter**. This will list all the ports running on the machine. Ensure that **port 443** is listening.

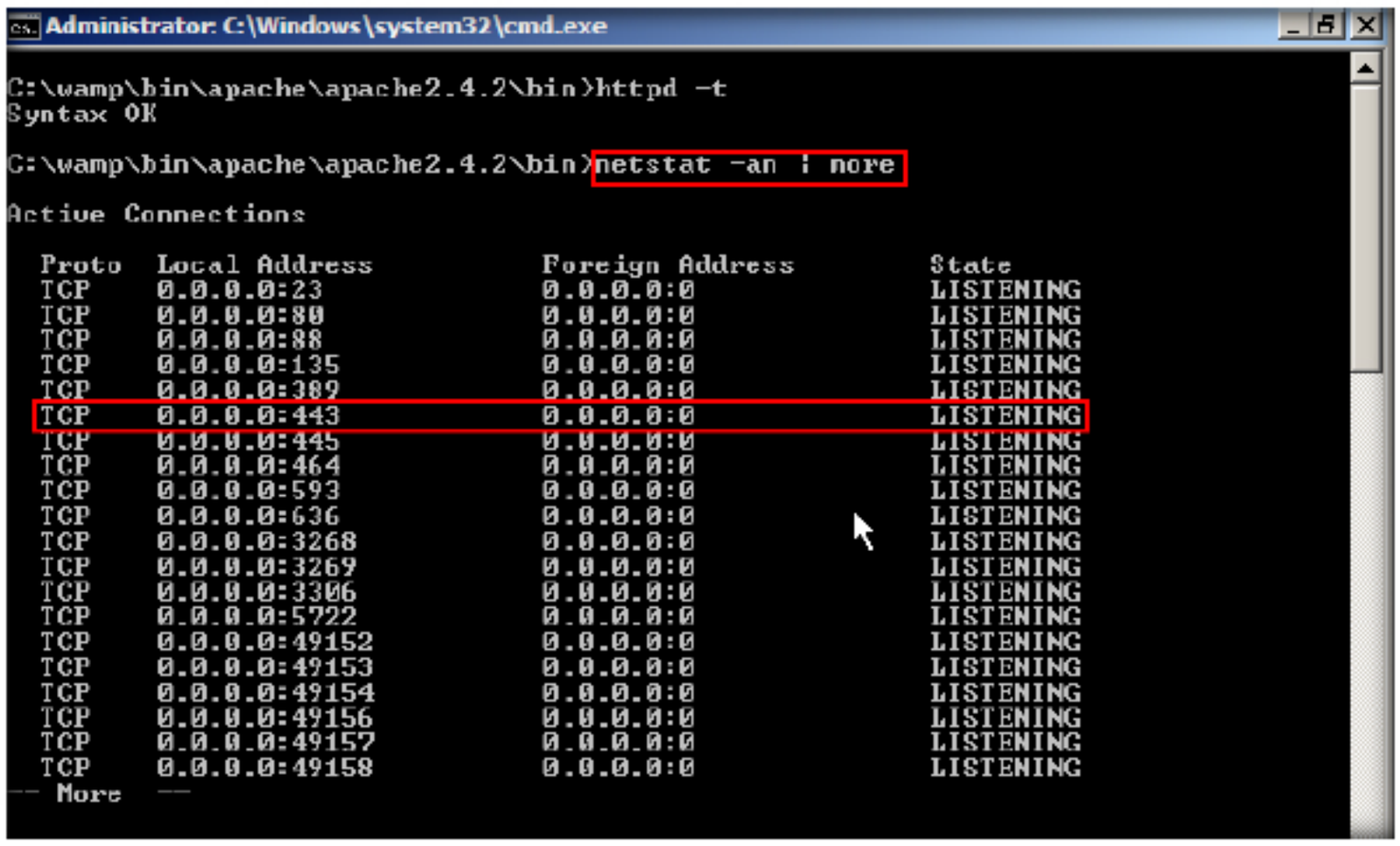

FIGURE 2.34: Issuing netstat command in command prompt

TASK 7

Browse on Https Channel

54. Now, Open a web browser, type the URL https://localhost/cnd in the address bar, and press **Enter**.
55. A webpage might appear, stating that the connection is not private .Click **Advanced**.

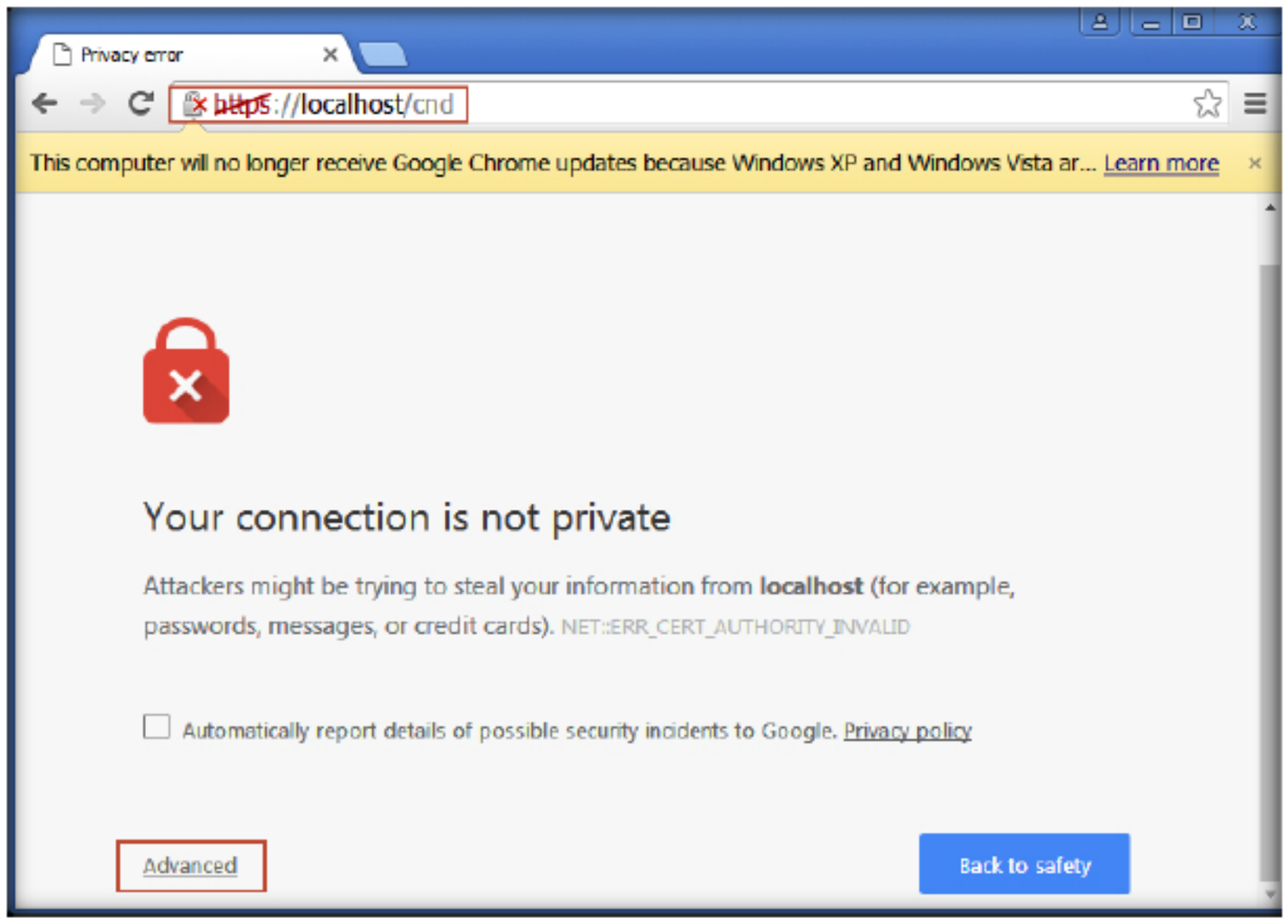

FIGURE 2.35: SSL certificate error

56. Click **Proceed to localhost(unsafe)** to at the bottom of browser screen.

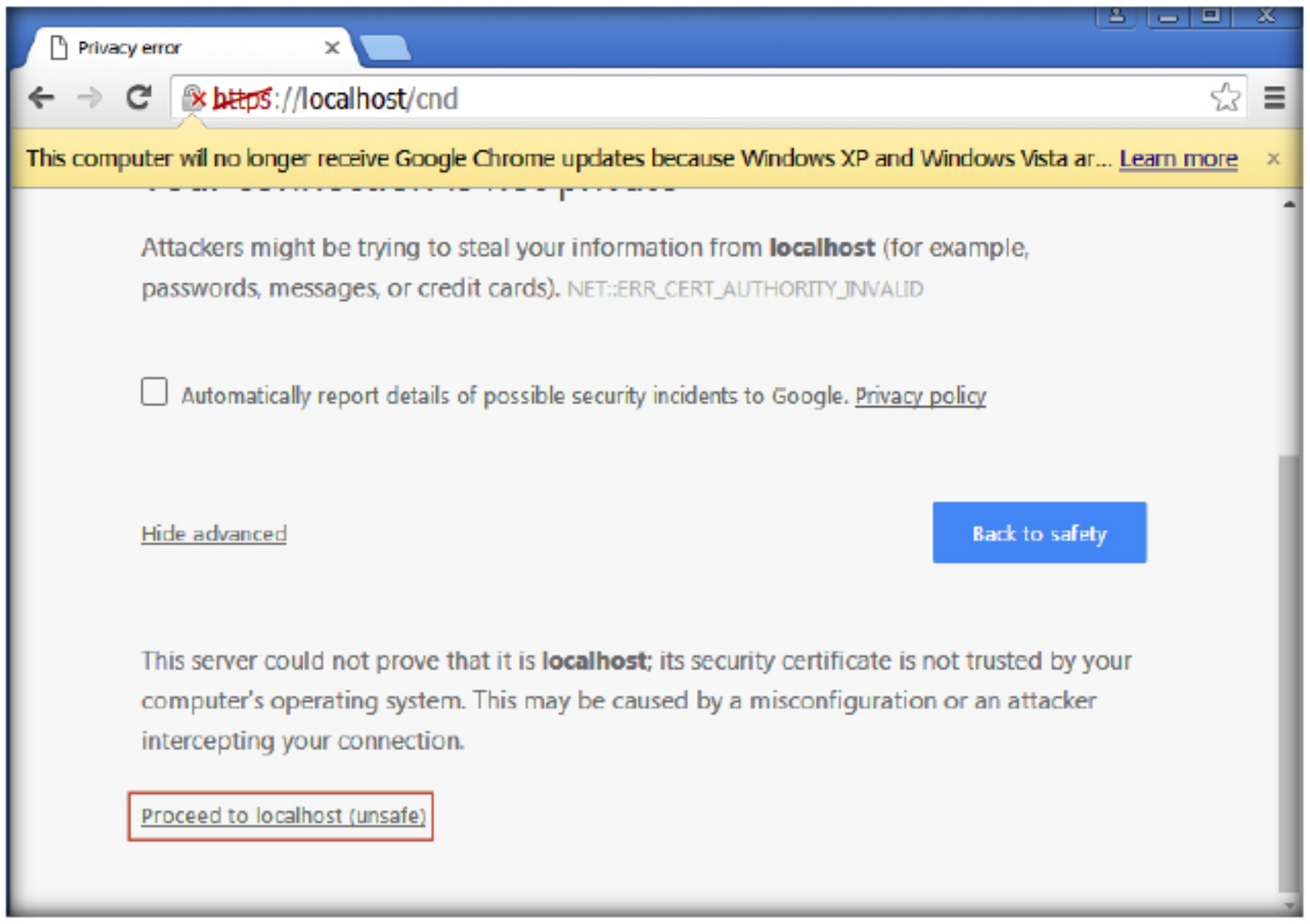

FIGURE 2.36: SSL certificate error

57. You will be redirected to the CND Wordpress website, as shown in the screenshot:

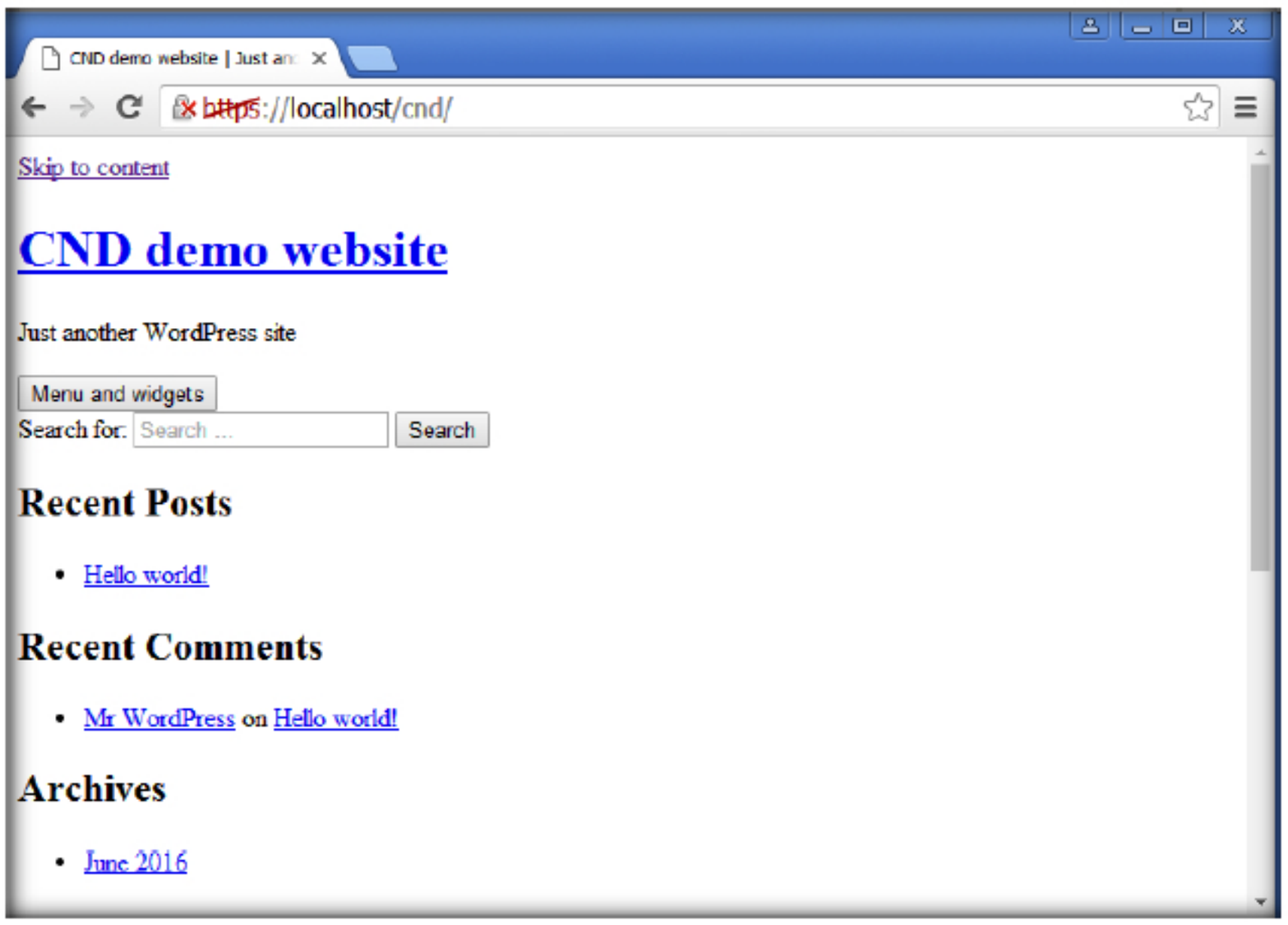

FIGURE 2.37: Browsing website on https channel

58. Now you can transfer data over the secure channel to prevent hackers from sniffing information in plain text, as the https channel offers encryption to the data traversing through it.

Lab Analysis

Analyze and document the results related to this lab exercise. Provide your opinion of your target's security posture and exposure through public and free information.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
☐ Yes	☑ No
Platform Supported	
☑ Classroom	☑ iLabs

Lab 3

Ensuring Secure Email Communication using PGP

Gpg4win (GNU Privacy Guard for Windows) is encryption software for files and emails. Gpg4win enables users to securely transport emails and files with the help of encryption and digital signatures.

Lab Scenario

Encryption protects the contents against unauthorized access. Digital signatures make sure that it is not modified and comes from a specific sender. It is essential to use digital signatures for email communication in order to avoid having the email sniffed.

ICON KEY

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

Lab Objectives

This lab will teach you how to secure email communication using PGP.

Lab Environment

To carry out the lab, you need:

- **Gpg4win** located at **Z:\CND-Tools\CND Module 03 Network Security Controls, Protocols, and Devices\Data Encryption Tools\Gpg4win**
- You can also download the latest version of Gpg4win from the following link: https://www.gpg4win.org/download.html
- If you decide to download the latest version, then the screenshots shown in the lab may differ
- A virtual machine running Windows Server 2012
- A virtual machine running Windows 10
- A Web browser with an Internet connection
- **Administrative** privileges to run tools

Tools demonstrated in this lab are available in Z:\CND-Tools\CND Module 03 Network Security Controls, Protocols, and Devices

Lab Duration

Time: 45 Minutes

Overview of PGP

Pretty Good Privacy (PGP) is a data encryption and decryption computer program that provides cryptographic privacy and authentication for data communication. PGP is often used for signing, encrypting, and decrypting texts, emails, files, directories, and whole disk partitions as well as to increase the security of email communication.

Lab Tasks

TASK 1

Install Gpg4win

1. To perform this lab, you need to have two email accounts. Create two new email accounts if you don't have them before starting the lab.
2. In this lab, we are going to use two email accounts that we have created for demonstration purposes. They are **cnduser1@gmail.com** and **cnduser2@gmail.com**.
3. Launch **Windows Server 2012** and create two folders on the desktop with the names **PGP keys** and **Public keys**
4. Navigate to **Z:\CND-Tools\CND Module 03 Network Security Controls, Protocols, and Devices\Data Encryption Tools\Gpg4win** and double-click the **gpgforwin-2.3.2.exe**
5. If an **Open File - Security Warning** window appears, click **Run**.

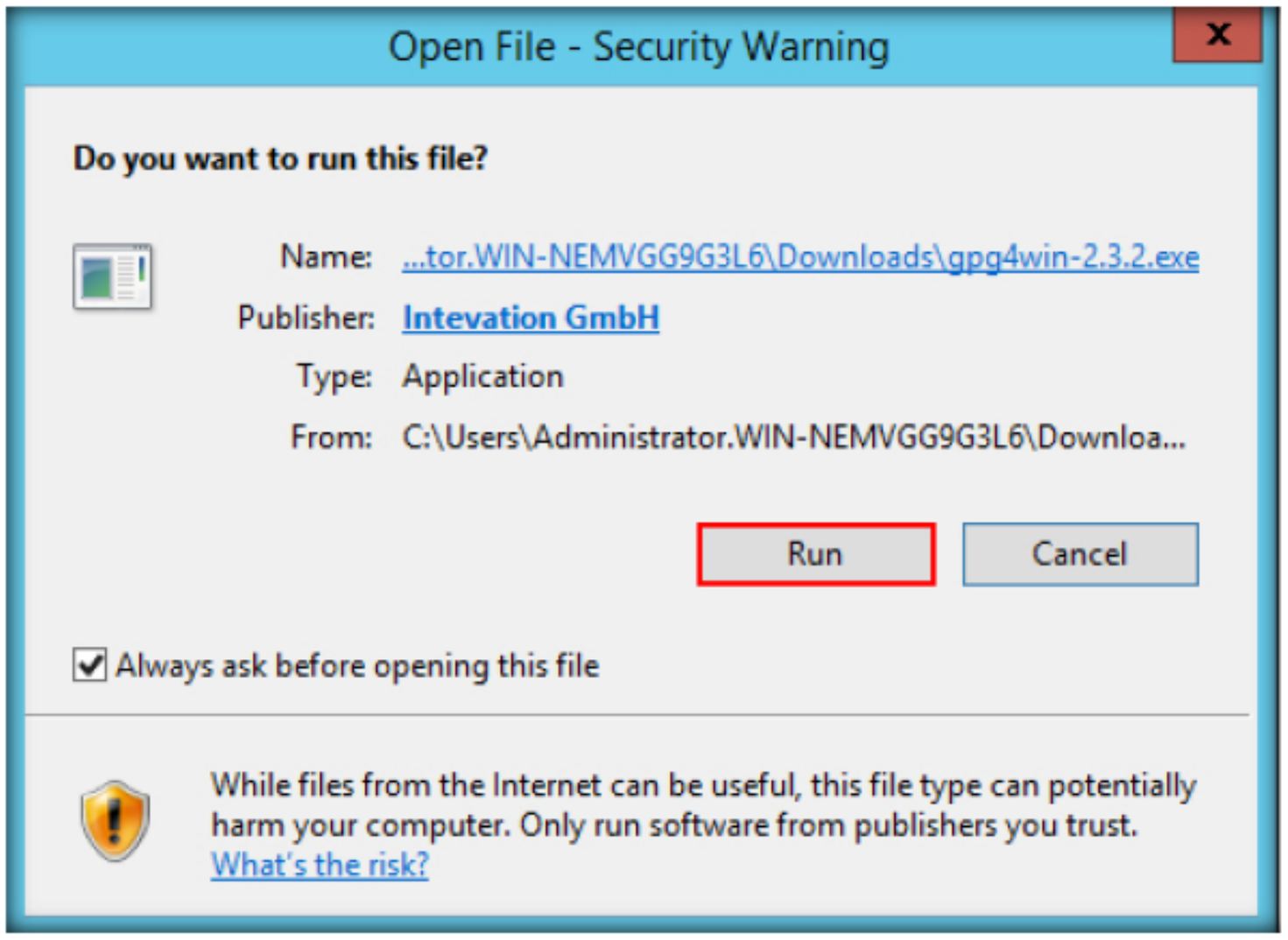

FIGURE 3.1: Windows Security Warning

6. The **Installer Language** window pops up. Click **OK**.

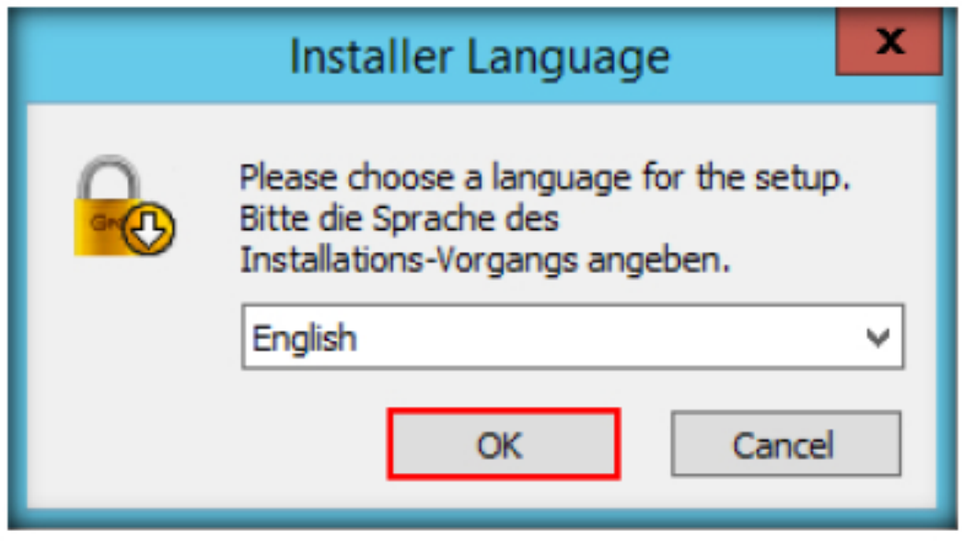

FIGURE 3.2: Selecting Language

Note: The default language is English. You can change the language to something else if you want to.

7. The **Gpg4win Setup** window appears. Click **Next**.

FIGURE 3.3: Installing Gpg for Windows

8. The **License Agreement** window appears. Click **Next**.

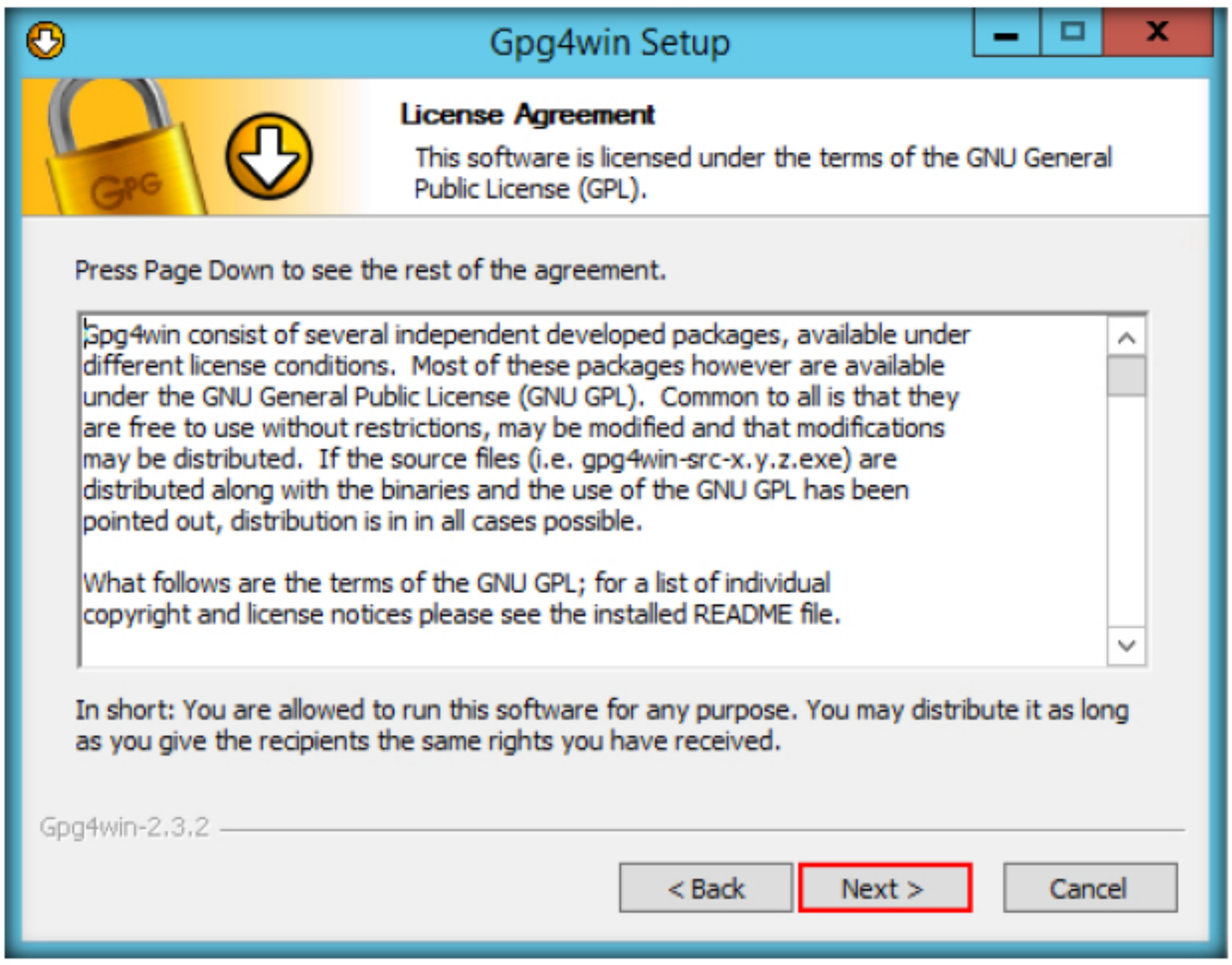

FIGURE 3.4: Accepting the License Agreement

9. The **Choose Components** window appears. Ensure that only the **GPA** component is selected and click **Next**.

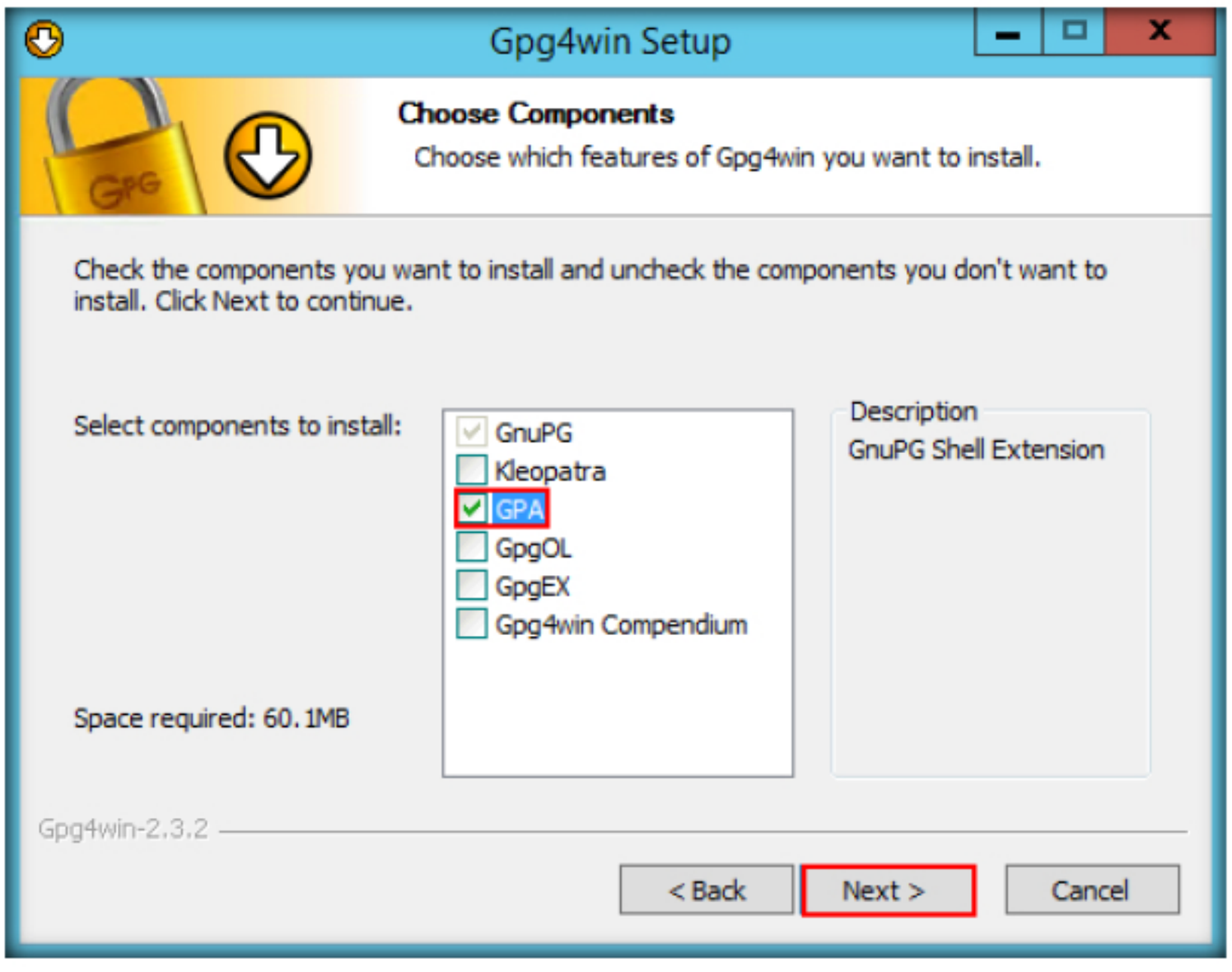

FIGURE 3.5: Selecting the Components

TASK 2

Create a Public Key

10. Follow the wizard driven instructions and complete the installation. In the final step of the installation, ensure the **Show the README File** option is unchecked and click **Finish**.
11. Navigate to Windows the **Apps** screen and click **GPA**.

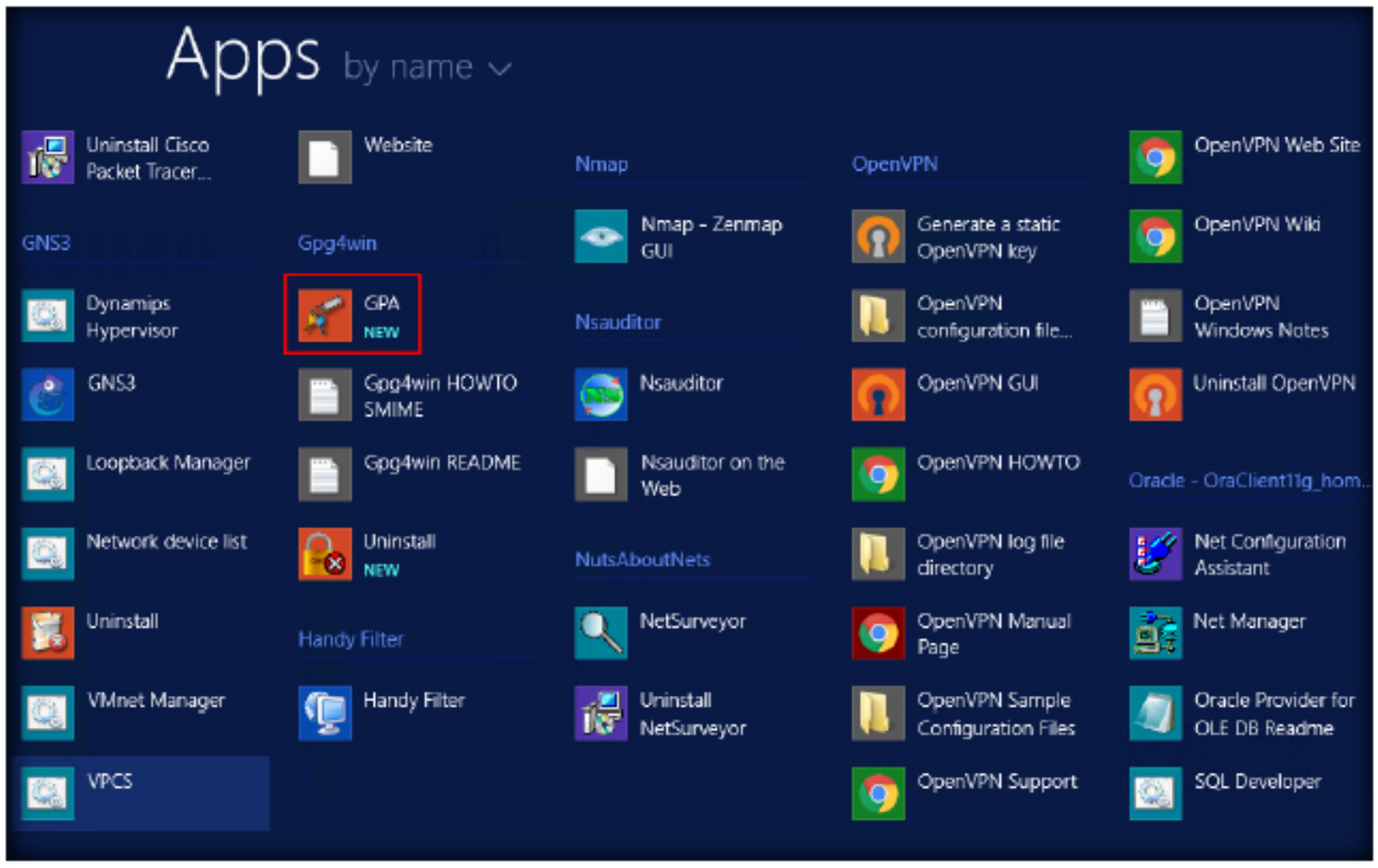

FIGURE 3.6: Navigating to GPA

12. The **GNU Privacy Assistant - Key Manager** window appears. Click **Generate key now**.

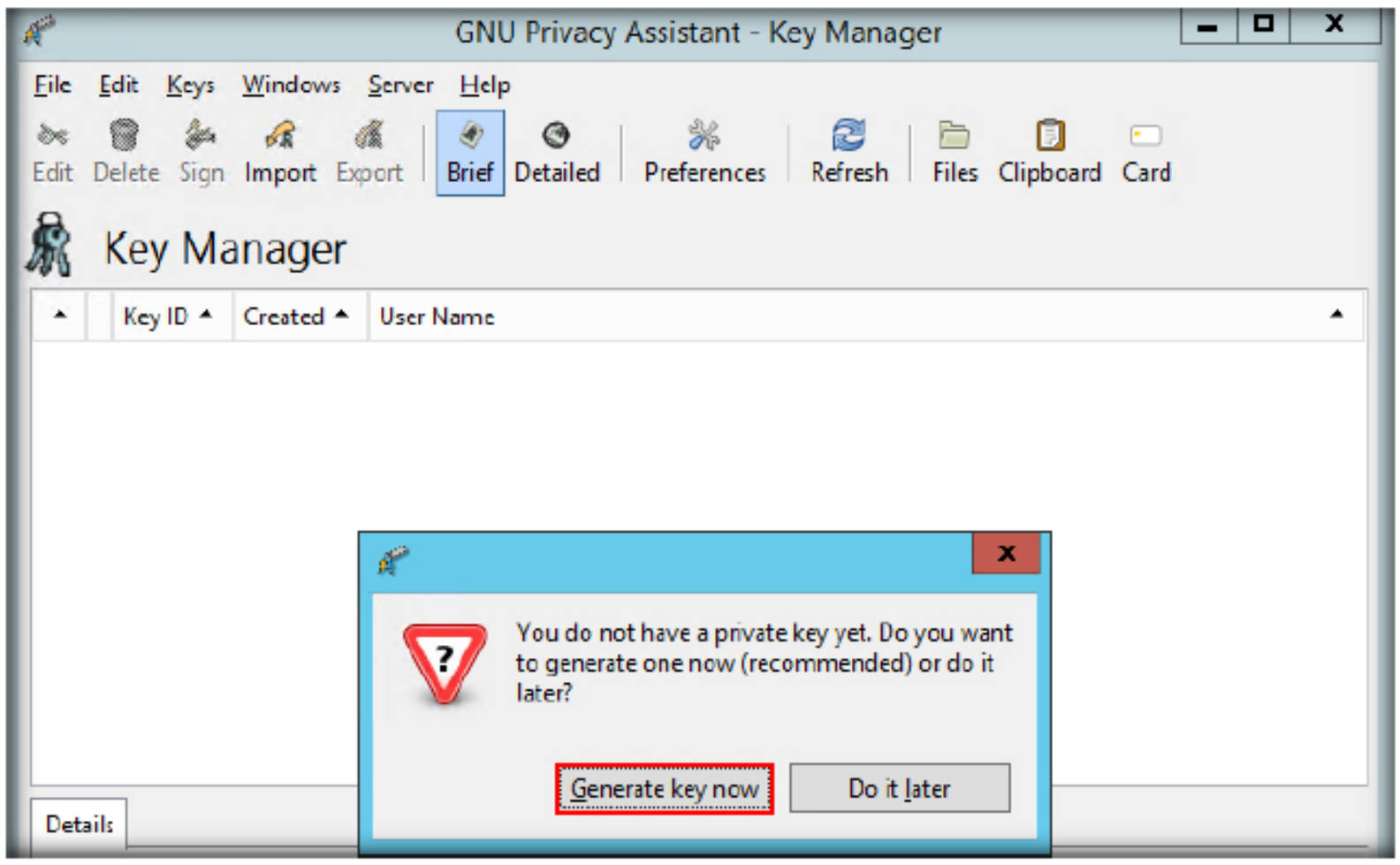

FIGURE 3.7: Generating Key

13. The **GNU Privacy Assistant - Generate key** window appears. Enter **CND user1** in the **Your Name:** field and click **Forward**.

FIGURE 3.8: Providing Key Name

14. Enter your first email account's ID (in this lab, we have used **cnduser1@gmail.com**) in the **Your Email Address:** field and click **Forward**.

FIGURE 3.9: Providing Key Mail Details

15. Ensure the **Create backup copy** radio button is selected. Next, click **Forward**.

FIGURE 3.10: creating Back Up

16. The **pinentry** window will pop up. Enter **qwerty@123** in the **Passphrase** field then click **OK**.

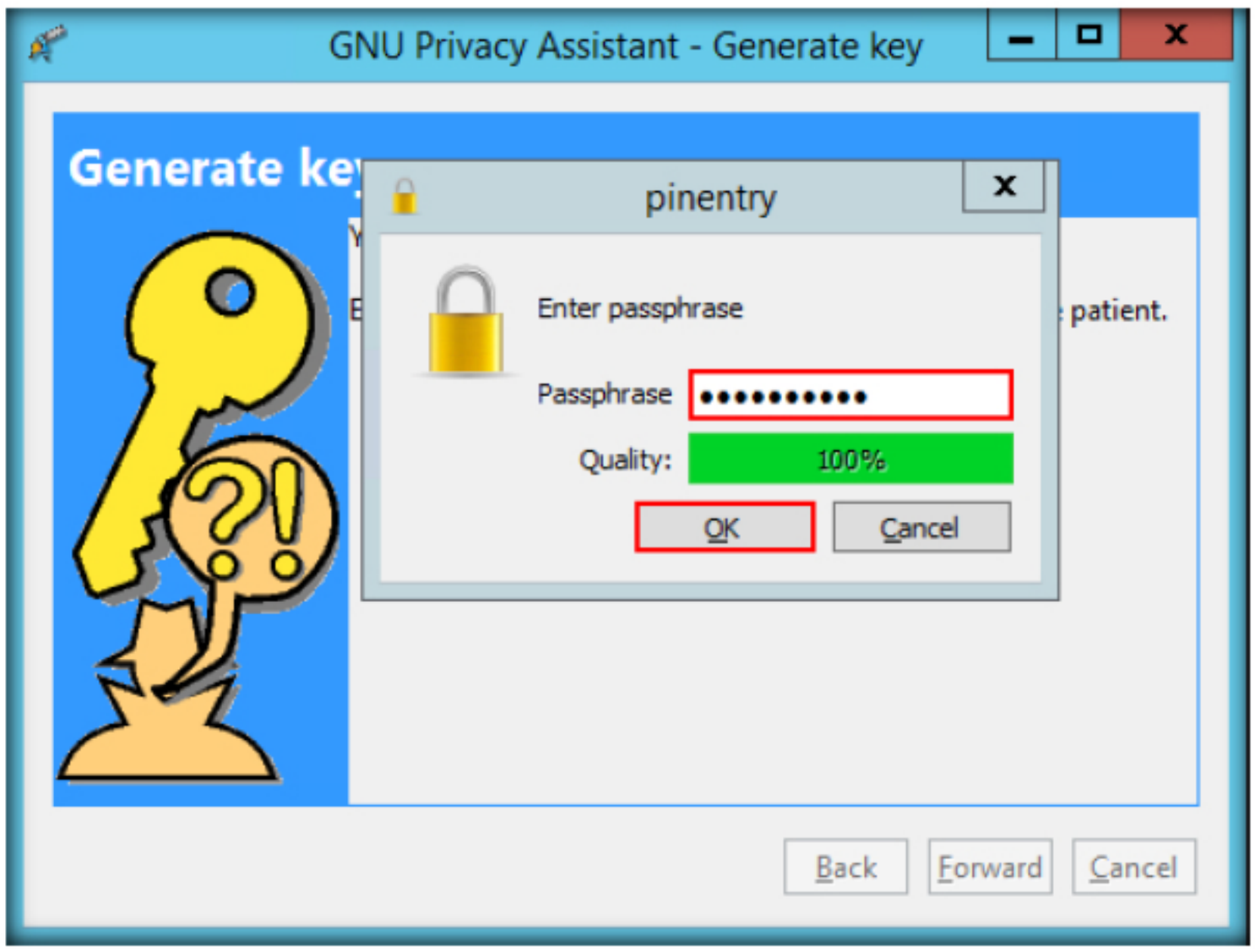

FIGURE 3.11: Creating Passphrase

17. Re-enter the Passphrase (qwerty@123) in the **Passphrase** field and click **OK**.

FIGURE 3.12: Reentering the Passphrase

18. The **Backup key to file** window appears. Select the **PGP keys** folder on the Desktop (created in step 2) then click **Save**.

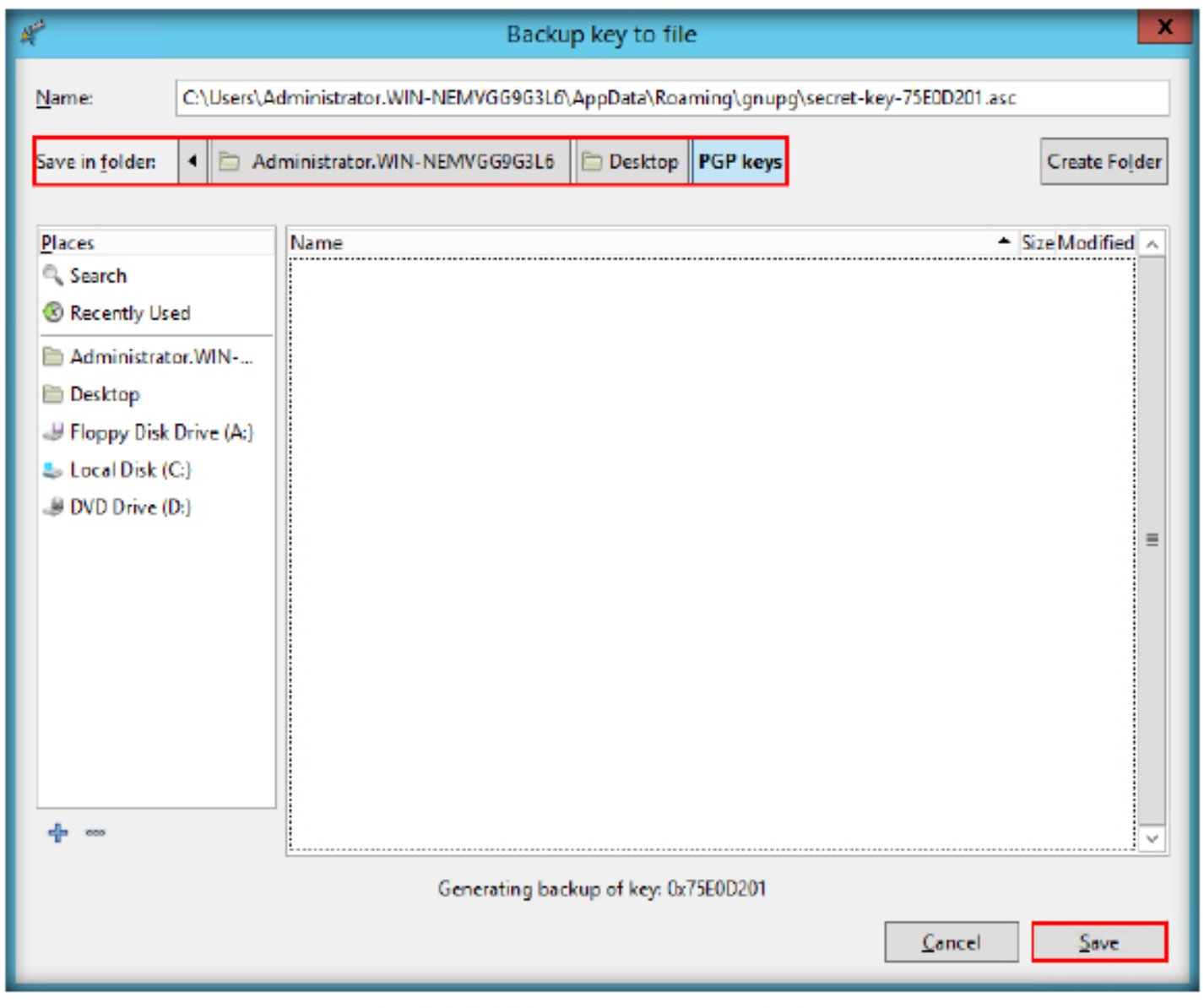

FIGURE 3.13: Saving a Backup of Keys

19. A pop up window will appear. Click **Close**.

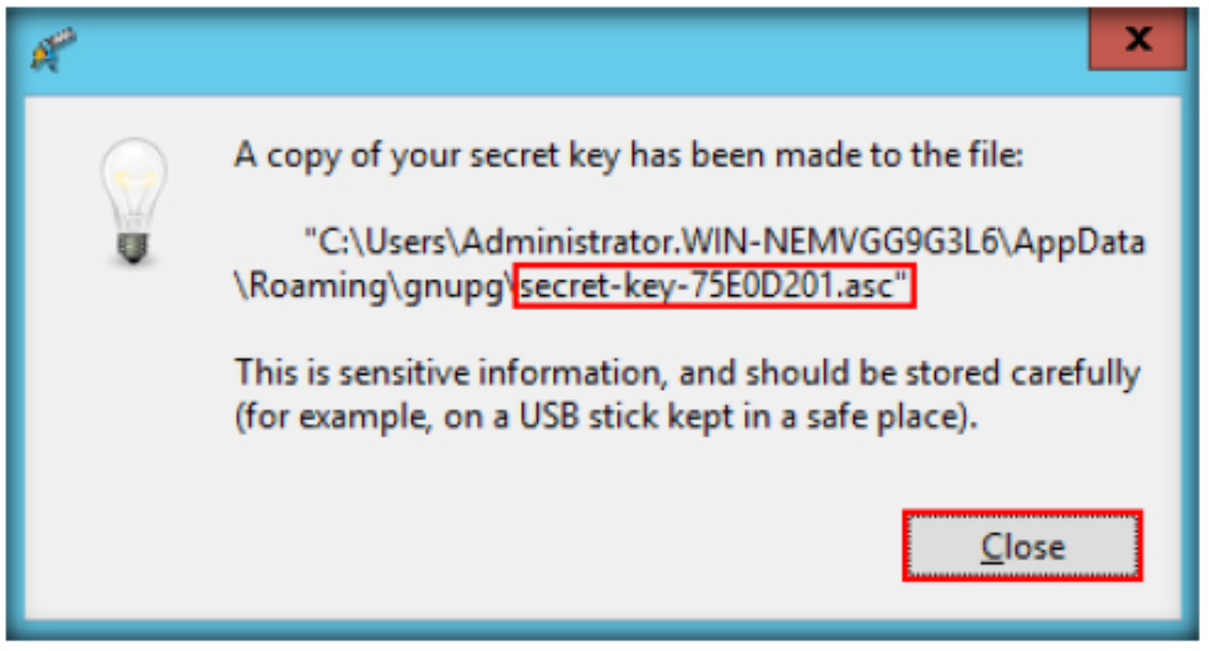

FIGURE 3.14: Secret Keys Saved

Note: Due to some restrictions, sometimes the PGP key is stored only in the default location, and the location given in these steps is ignored. Make a note of the location to retrieve the key when needed. You can see the new key appear on the GPG window.

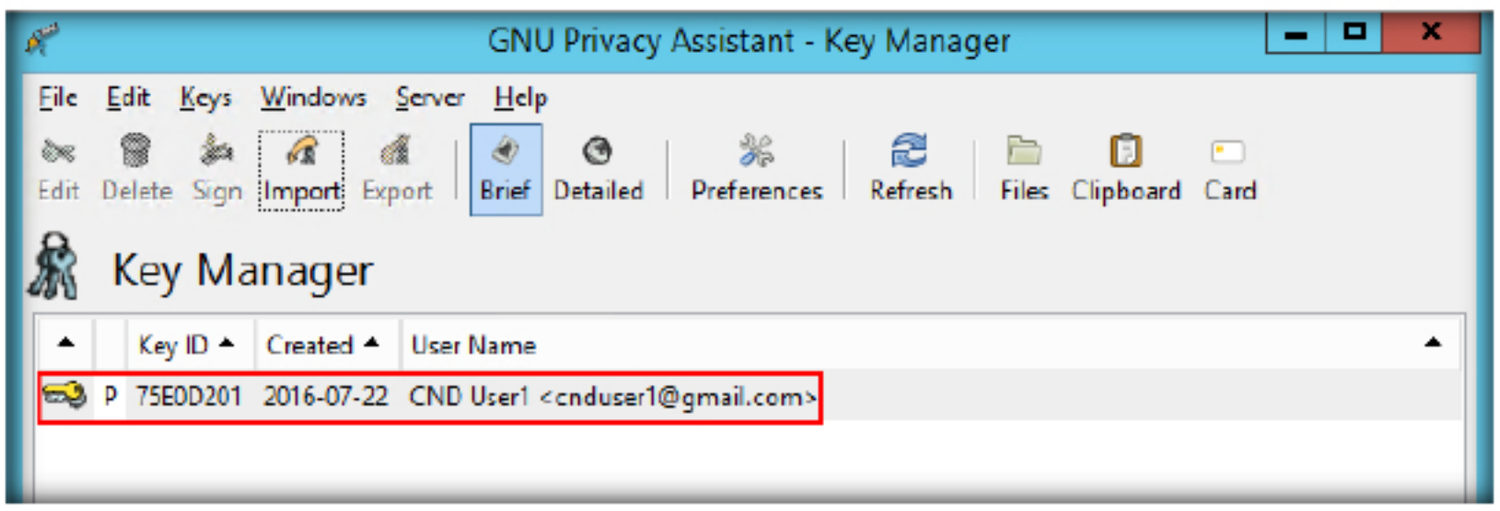

FIGURE 3.15: CND User 1 Key Created

20. Launch the **Public keys** folder on the desktop (created in step 3) and create a text document with name **cnd user1 public keys** and launch it.

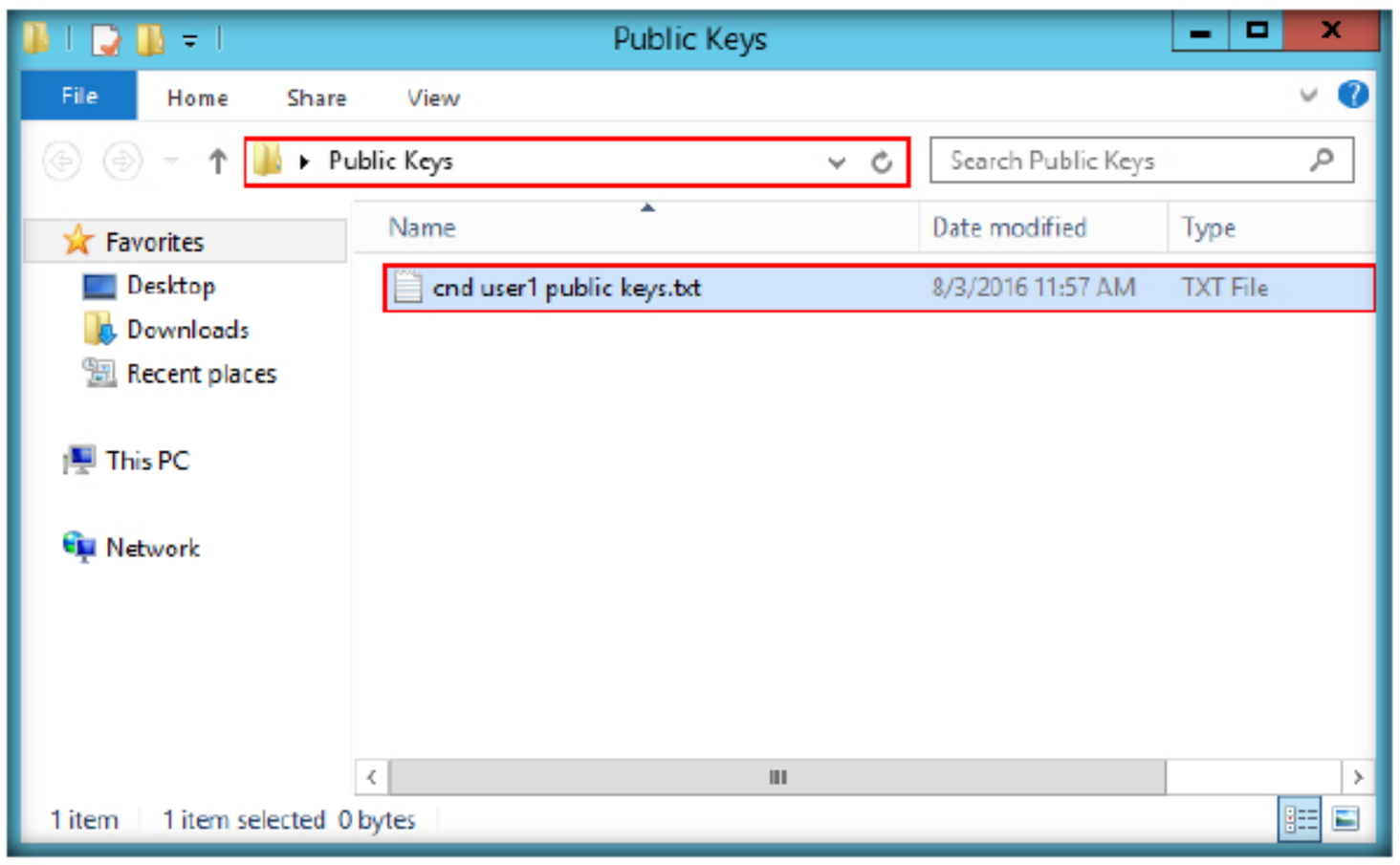

FIGURE 3.16: CND User 1 Public Key File Created

21. Switch back to the **GNU Privacy Assistant - Key Manager** window and right click on **cnd user1** then select **Copy** from the context menu.

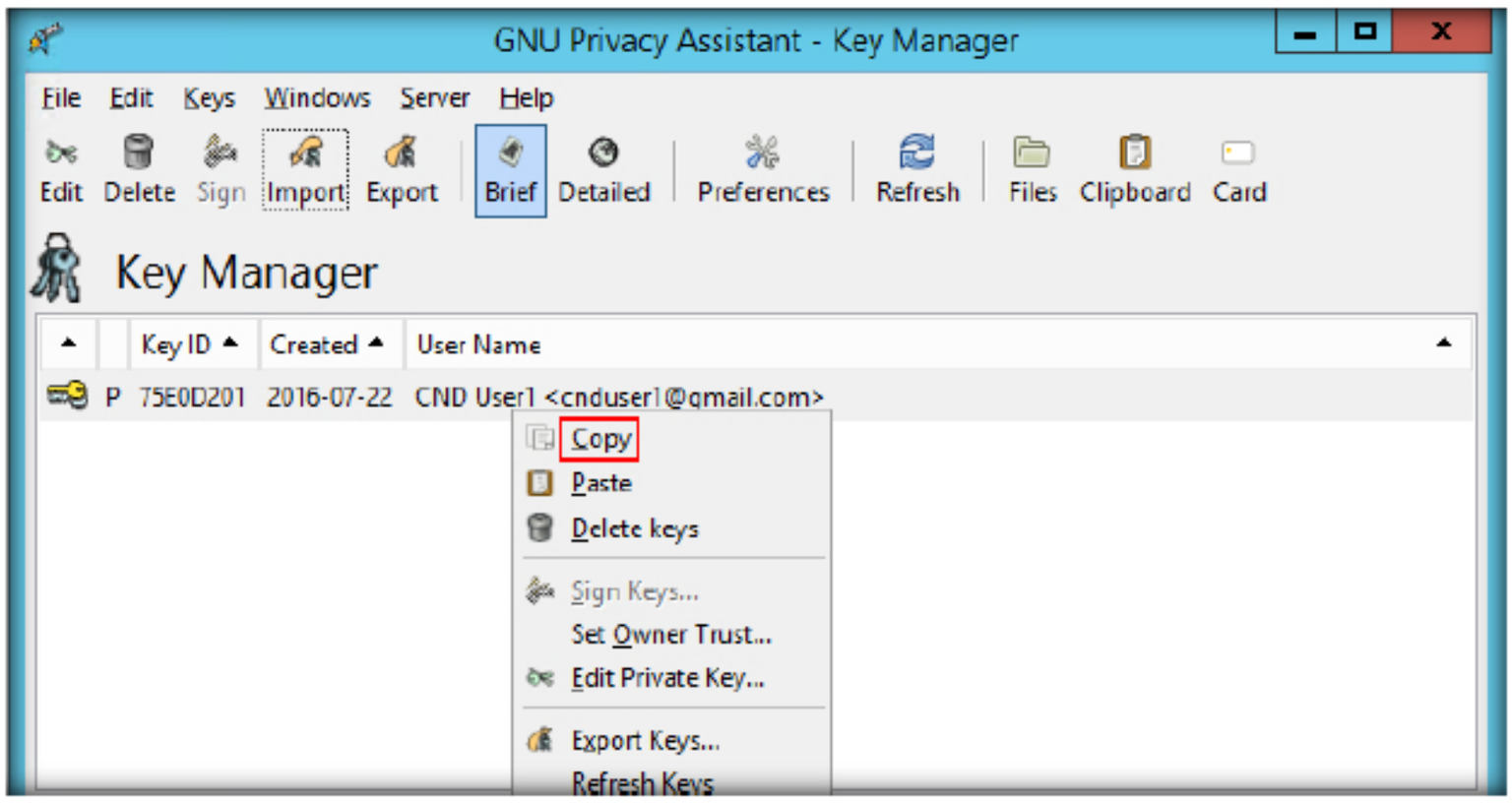

FIGURE 3.17: Copying CND User 1 Public Keys

22. Another Pop up window will appear. Click **Close**.

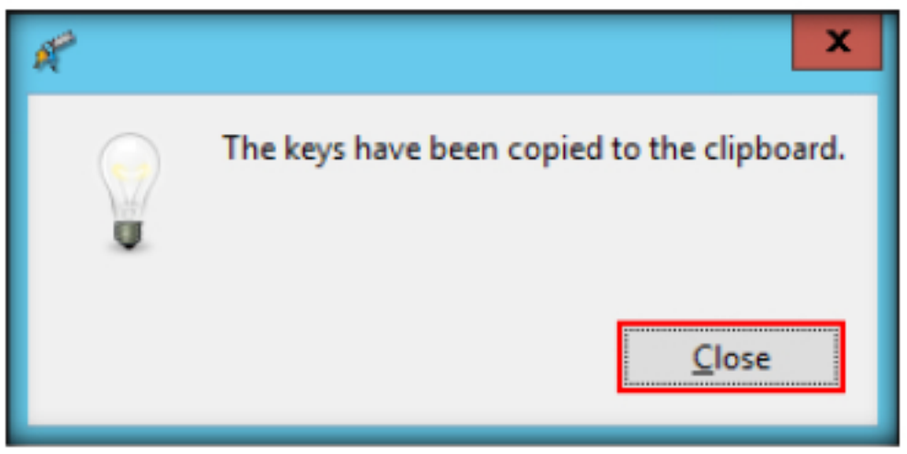

FIGURE 3.18: Keys Copied Successfully

23. Switch back to the **cnd user1 public keys** text file, right click and select **Paste**.

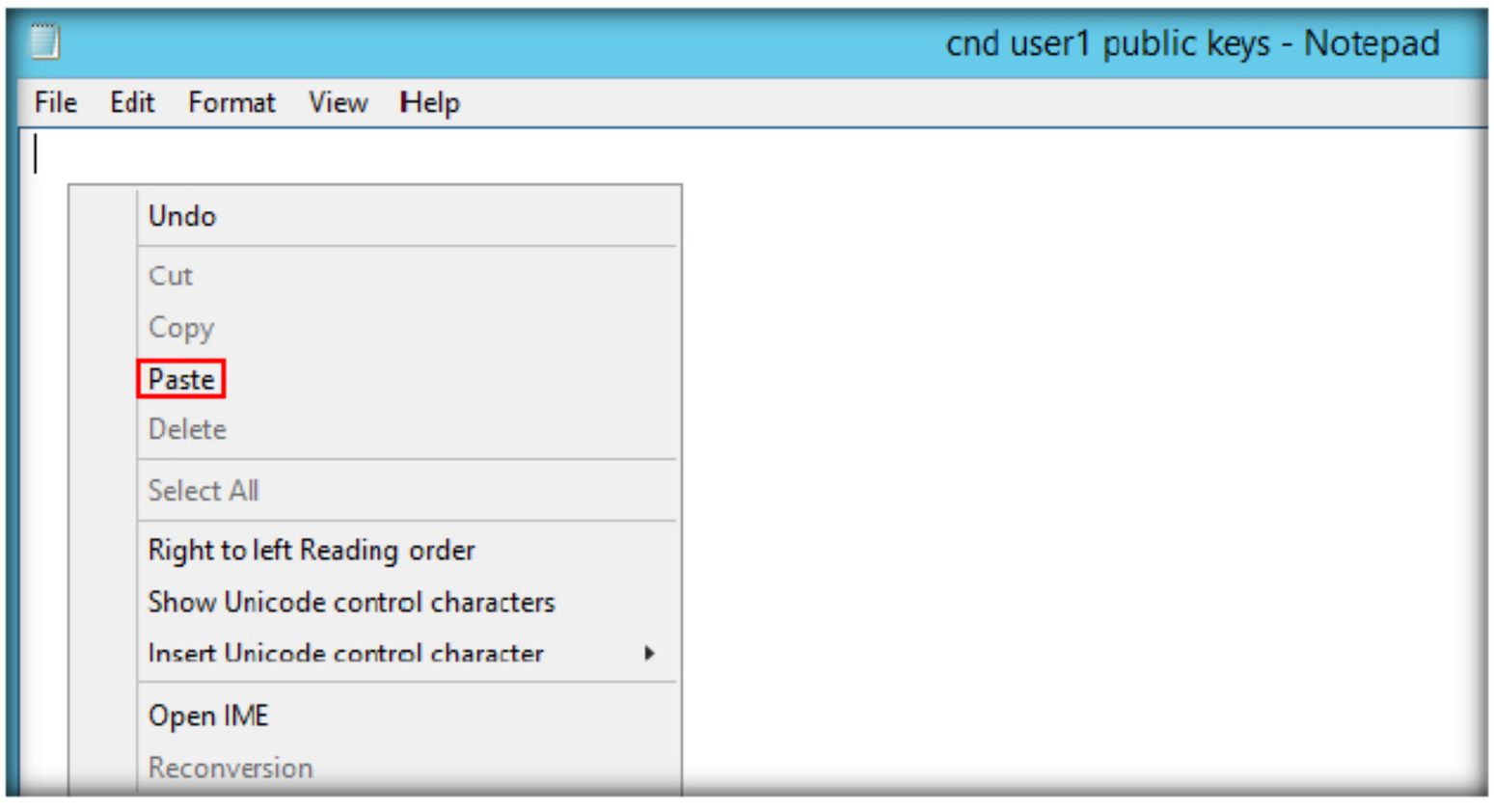

FIGURE 3.19: Pasting the Keys

24. You will see text in the file. It is the public key for **cnd user1**.

FIGURE 3.20: Cnd User1 Public Keys

TASK 3

Share the Folder

25. **Save** the text file and close it. Right-click on the **Public keys** folder and select **Properties**.

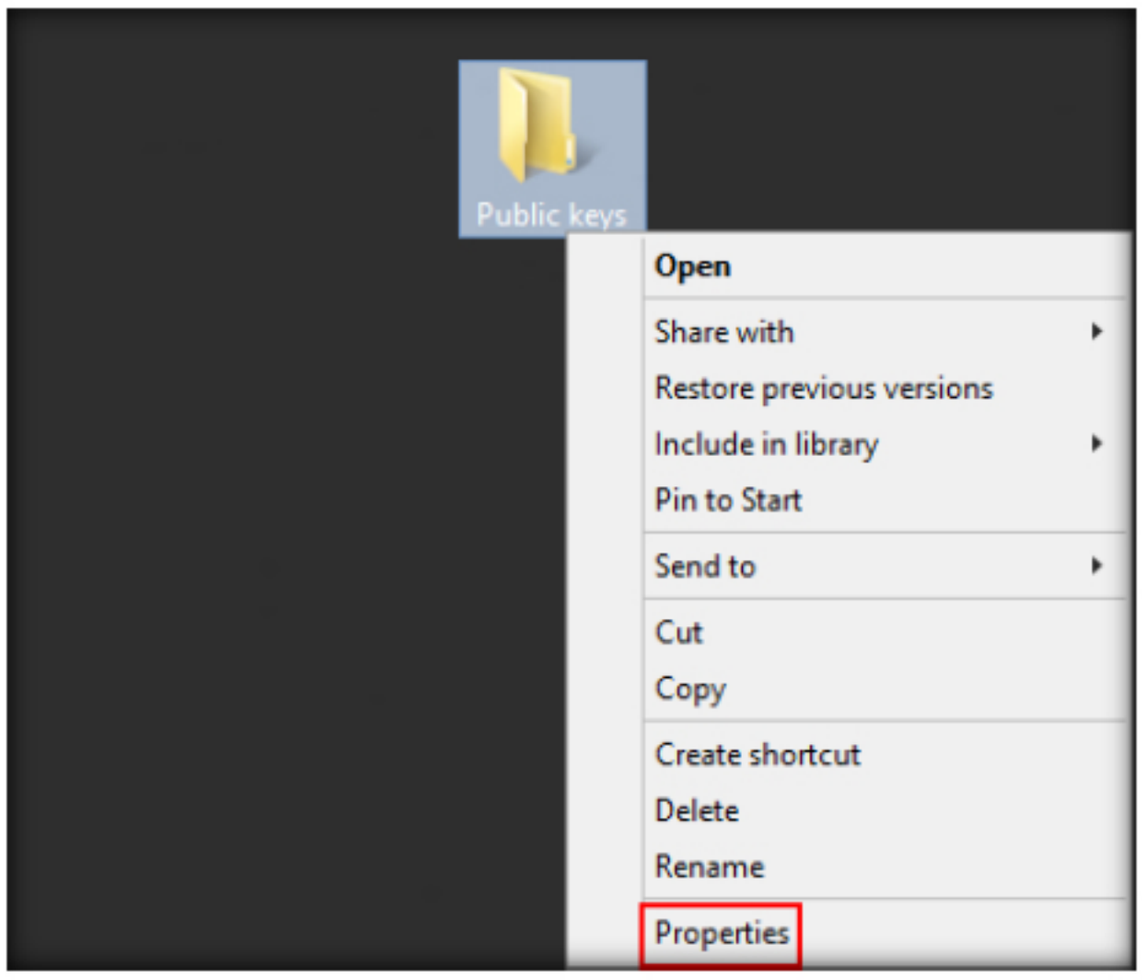

FIGURE 3.21: Navigating to Pubic Keys Properties

26. The **Public keys Properties** window appears. Click the **Sharing** tab.

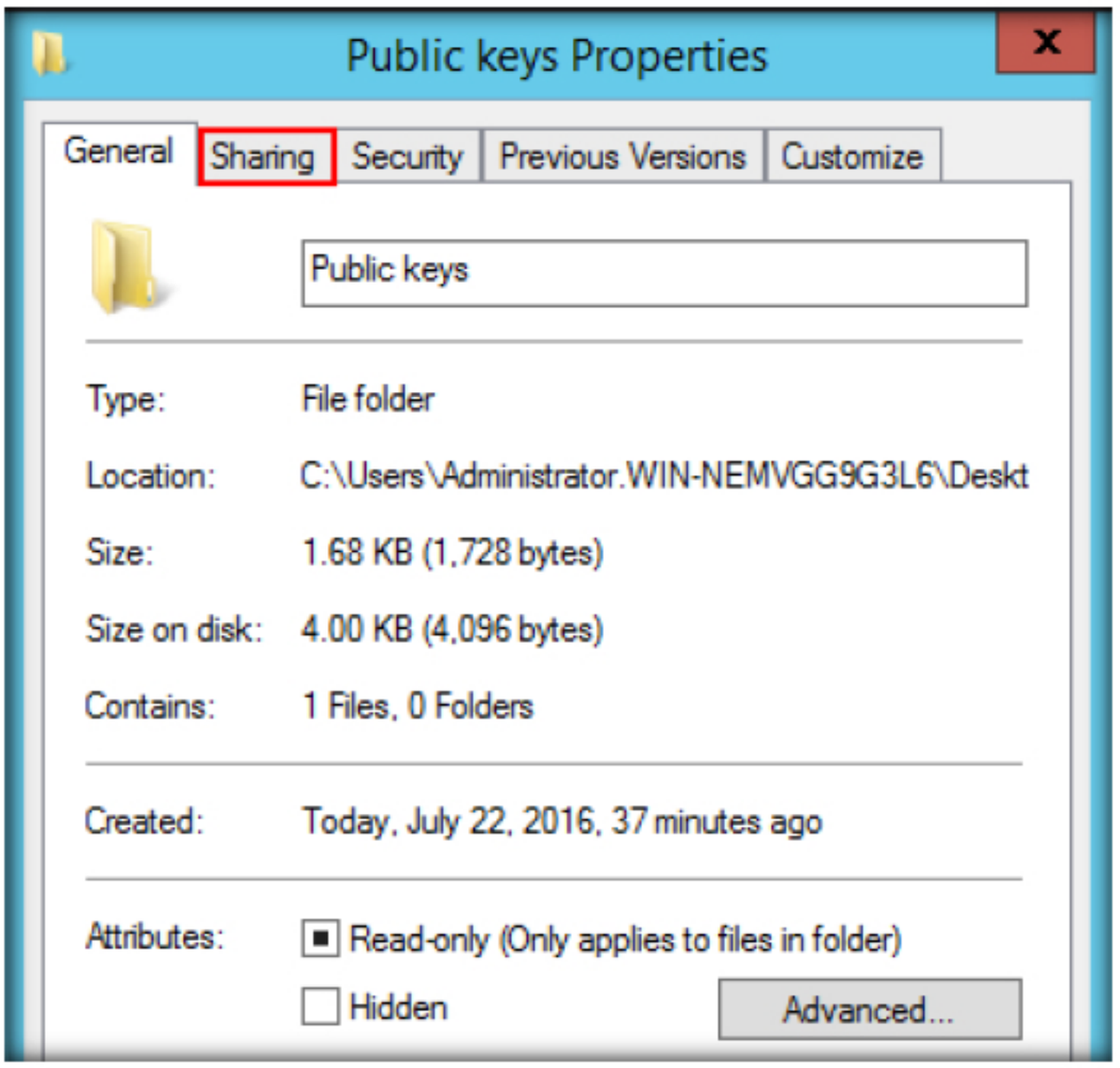

FIGURE 3.22: Navigating to Sharing Tab

27. The **Sharing** tab appears. Click **Share...**.

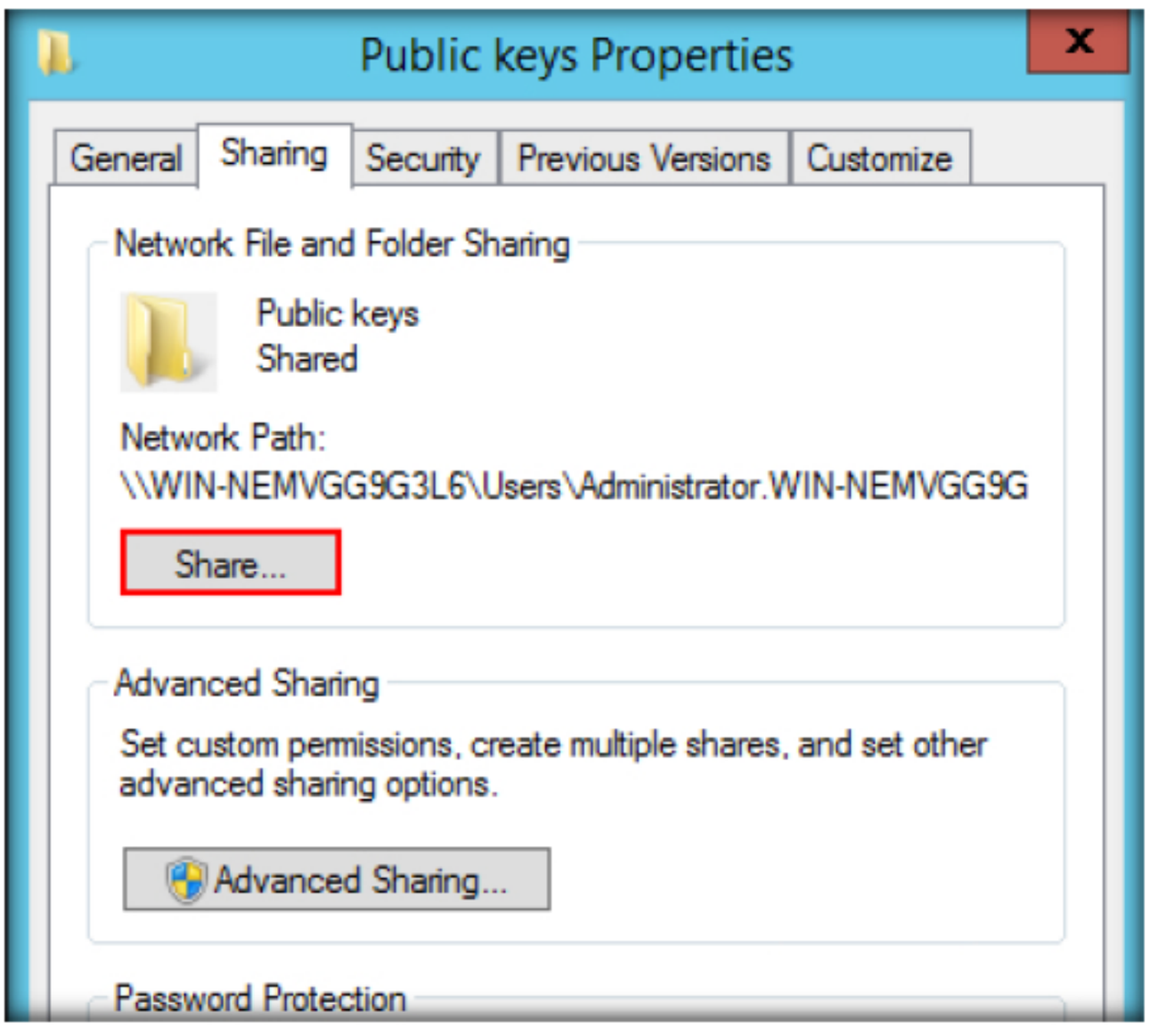

FIGURE 3.23: Navigating to Share Option

28. The **File Sharing** window appears. Click **Administrator** then click **Share**.

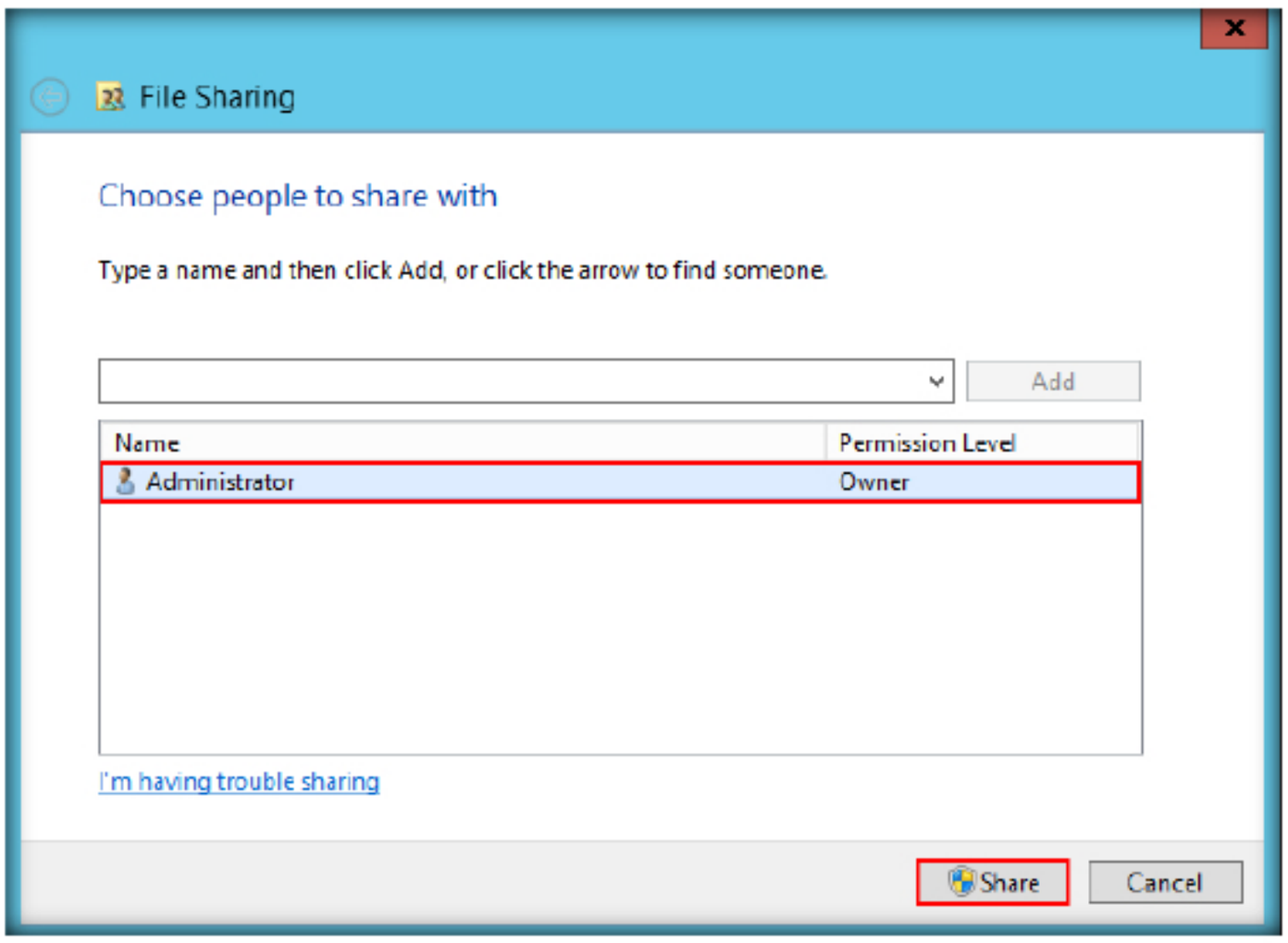

FIGURE 3.24: Sharing with Administrator

Note: The account name may vary in your environment.

If a **Network discovery and file sharing** pop-up appears, select **Yes, turn on network discovery and file sharing for all public networks**.

29. Click **Done**.

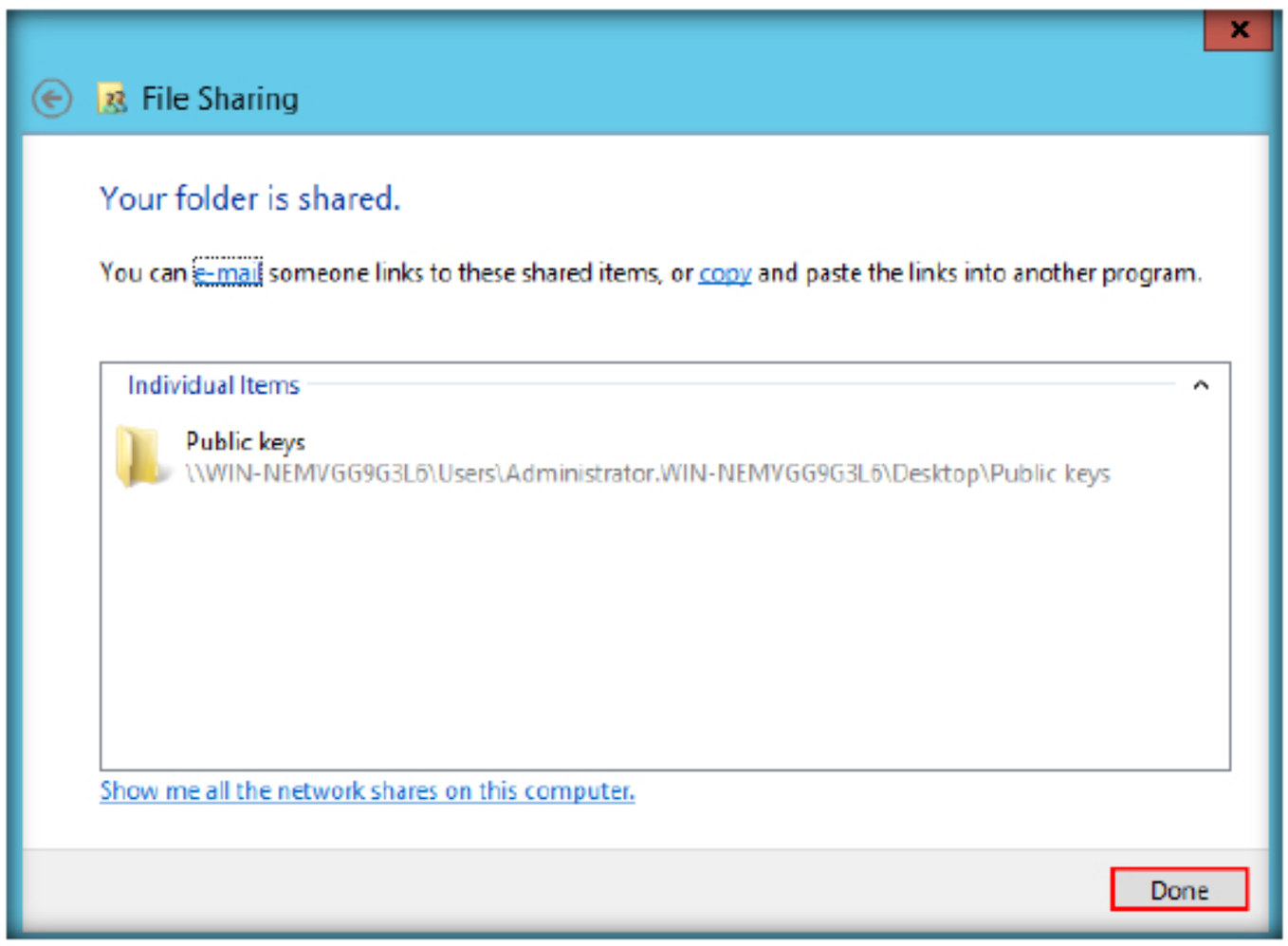

FIGURE 3.25: File Shared Successfully

30. Close the **Public Keys** properties window and login as the **Local Administrator** to the **Windows 10** machine.

TASK 4

Install Gpg4win

31. Create two folders on the desktop with the names of **PGP keys** and **Public keys**.
32. Navigate to **Z:\CND-Tools\CND Module 03 Network Security Controls, Protocols, and Devices\Data Encryption Tools\Gpg4win**, double-click **gpg4win-2.3.2.exe** and follow steps **4-10** to complete the installation of the **GPA** tool.

Note: If a **User Account Control** pop-up appears, click **Yes**.

33. Navigate to **Start → All apps → Gpg4win** and click **GPA**.

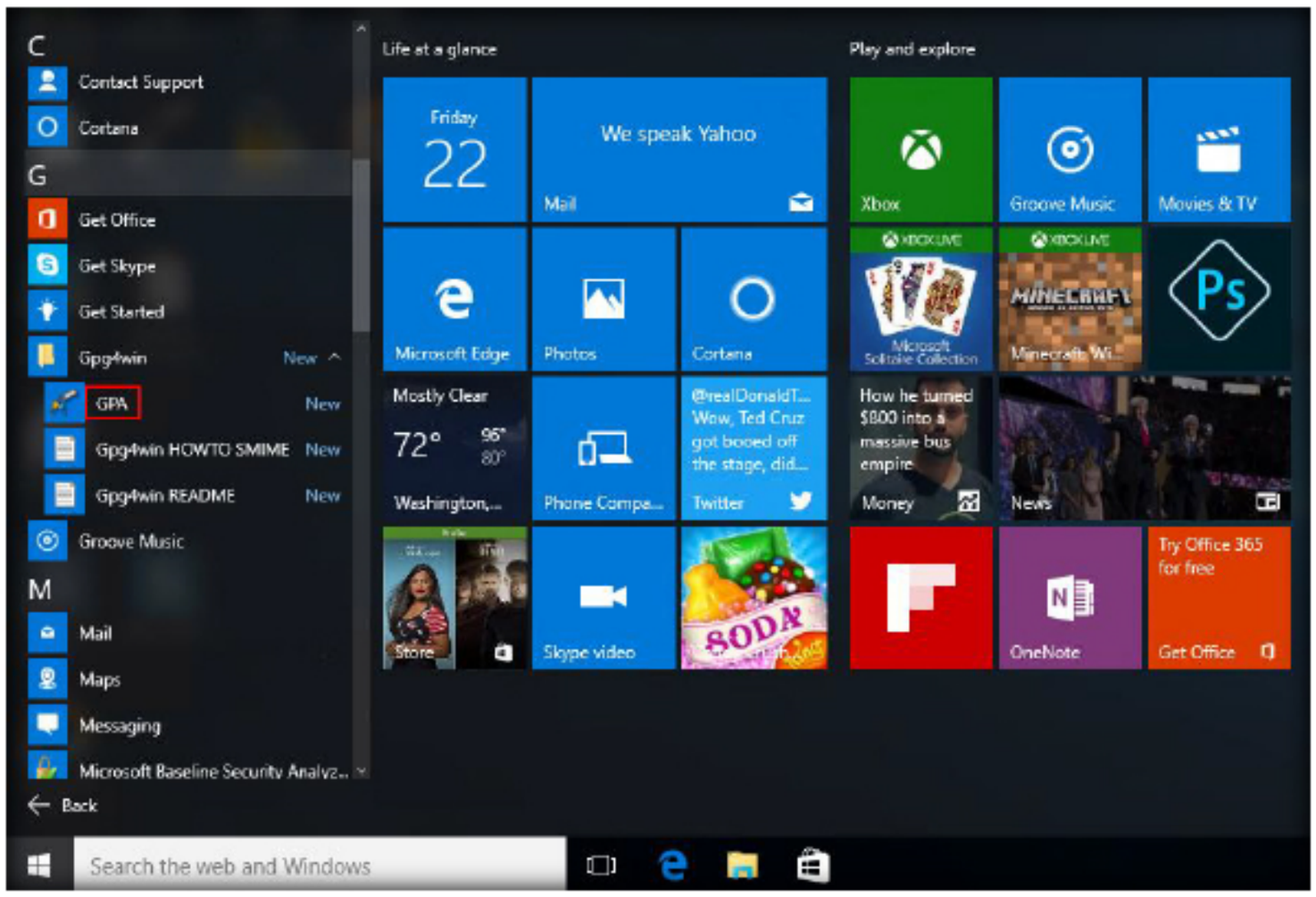

FIGURE 3.26: Launching GPA Tool

34. The **GNU Privacy Assistant - Key Manager window** appears. Click **Generate key now**.

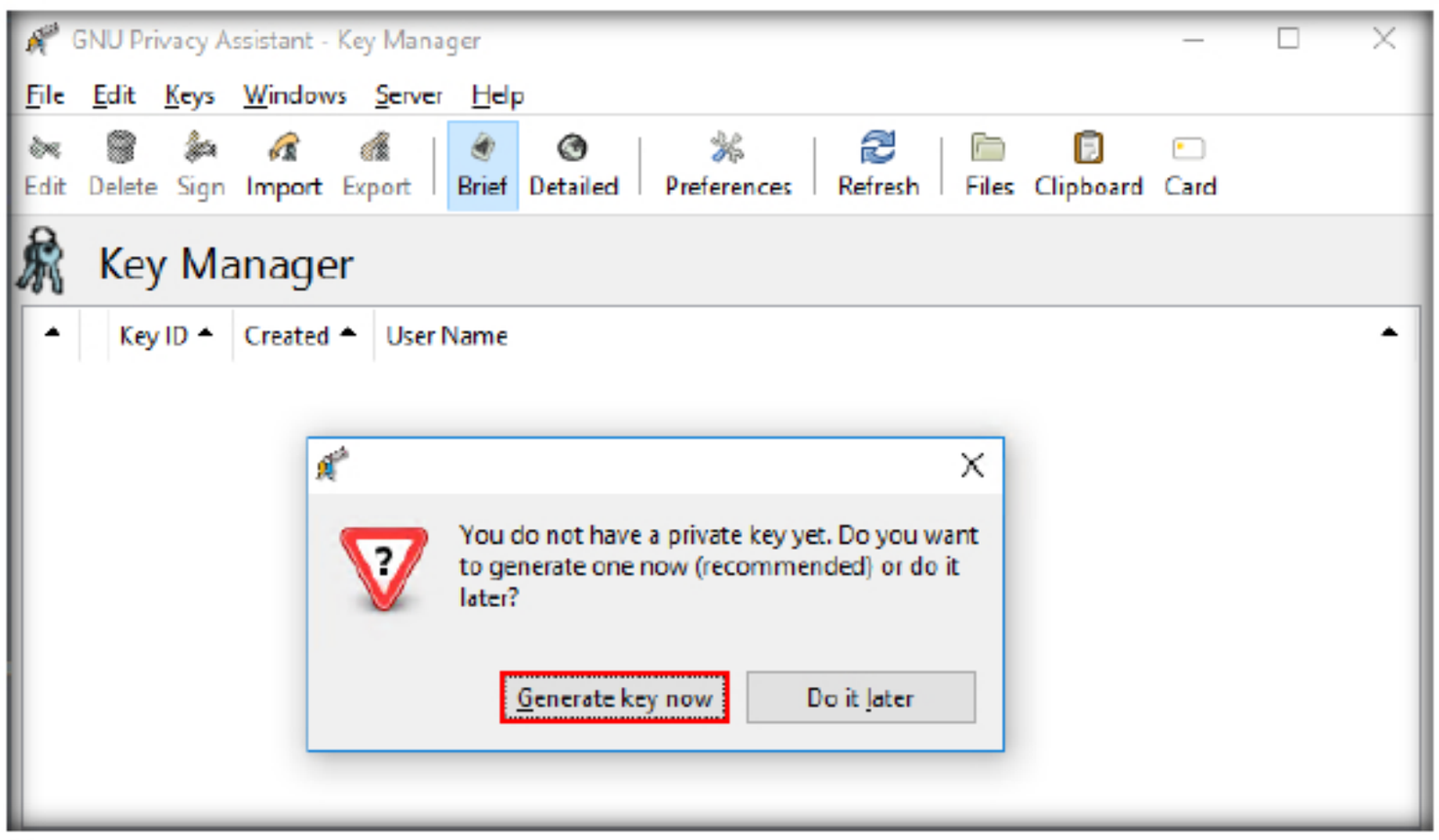

FIGURE 3.27: navigating to Key Generation

TASK 5

Create Public Key

35. The **GNU Privacy Assistant - Generate key** window appears. Enter **CND user2** in the **Your Name:** field then click **Forward**.

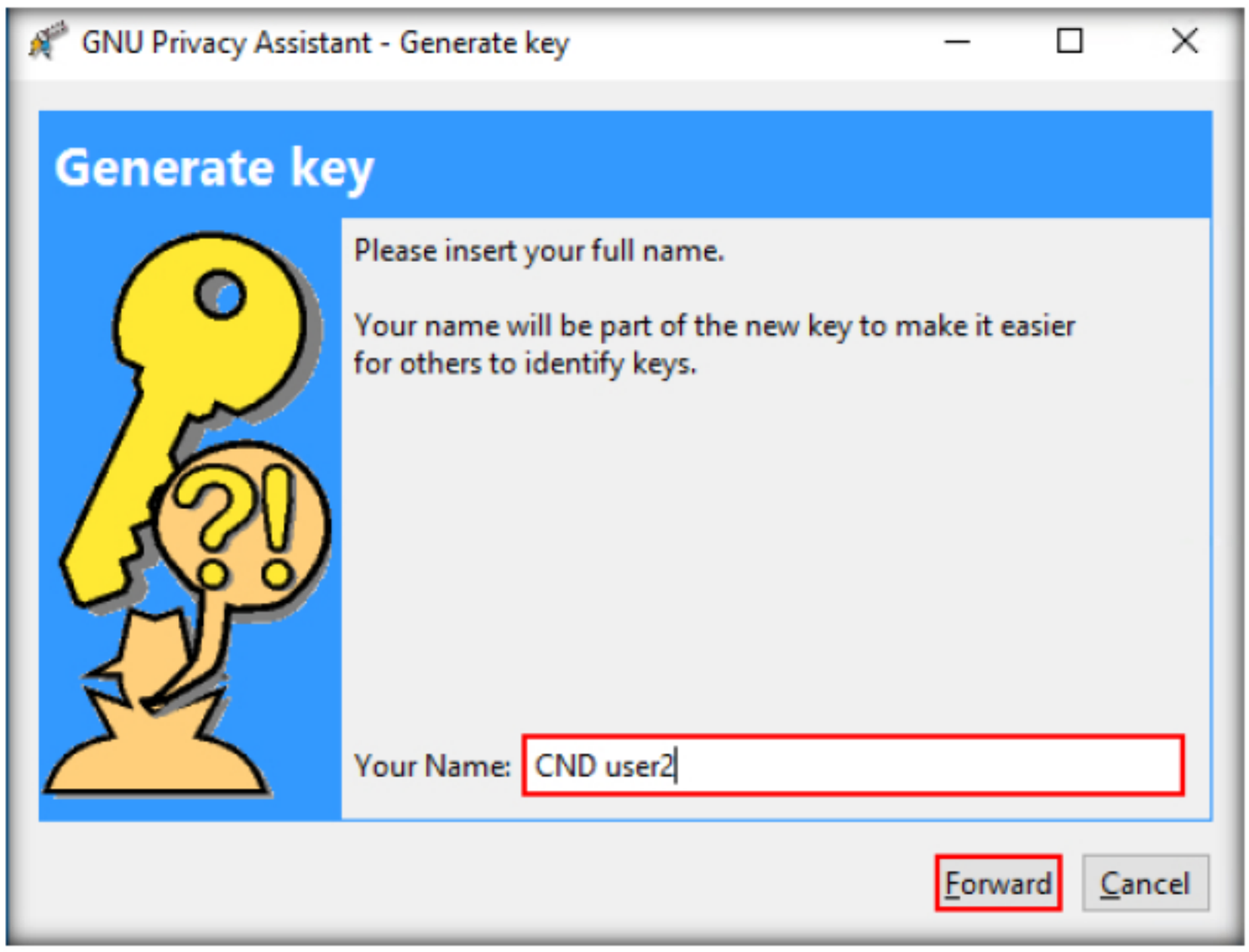

FIGURE 3.28: CND User2 Name Creation

36. Enter your second email account's ID (in this lab, we have used **cnduser2@gmail.com**) in the **Your Email Address:** field then click **Forward**.

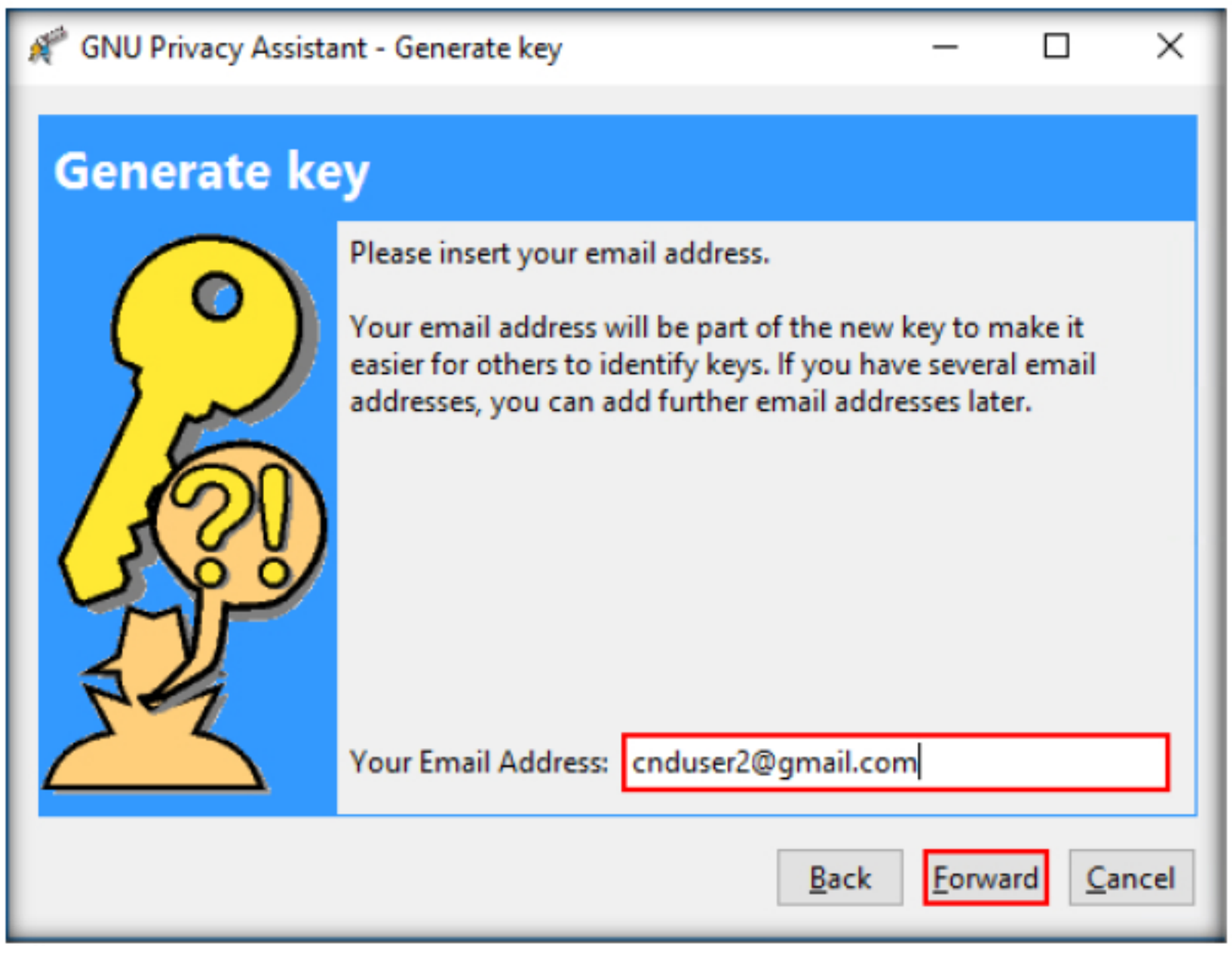

FIGURE 3.29: CND User2 Mail Creation

37. Ensure the **Create backup copy** radio button is clicked. Then click **Forward**.

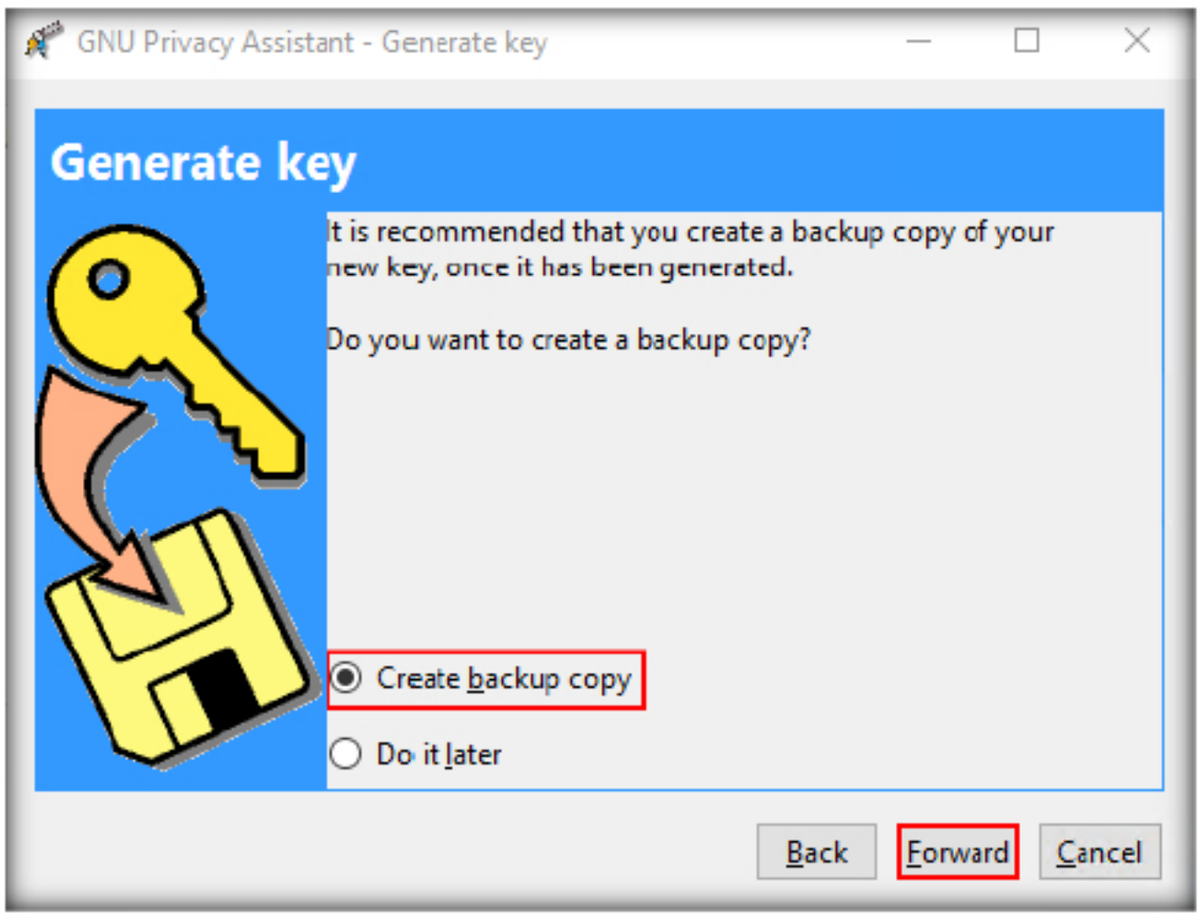

FIGURE 3.30: Creating Back Up of the Key

38. The **pinentry** window pops up. Enter **qwerty@321** in the Passphrase field and click **OK**.

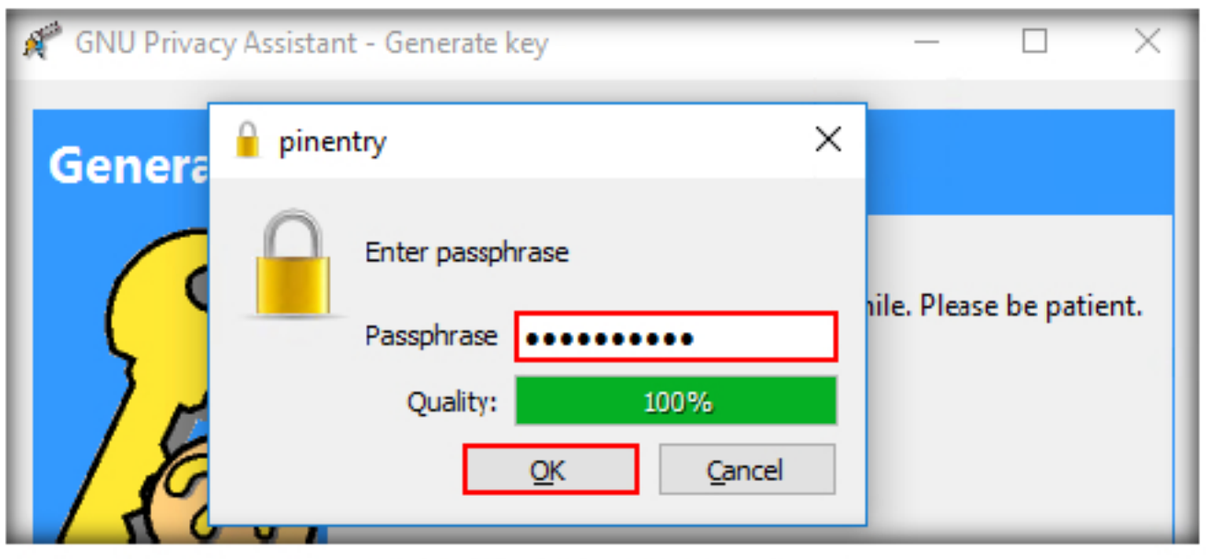

FIGURE 3.31: Creating Passphrase

39. Re-enter the Passphrase (qwerty@321) in the **Passphrase** field and click **OK**.

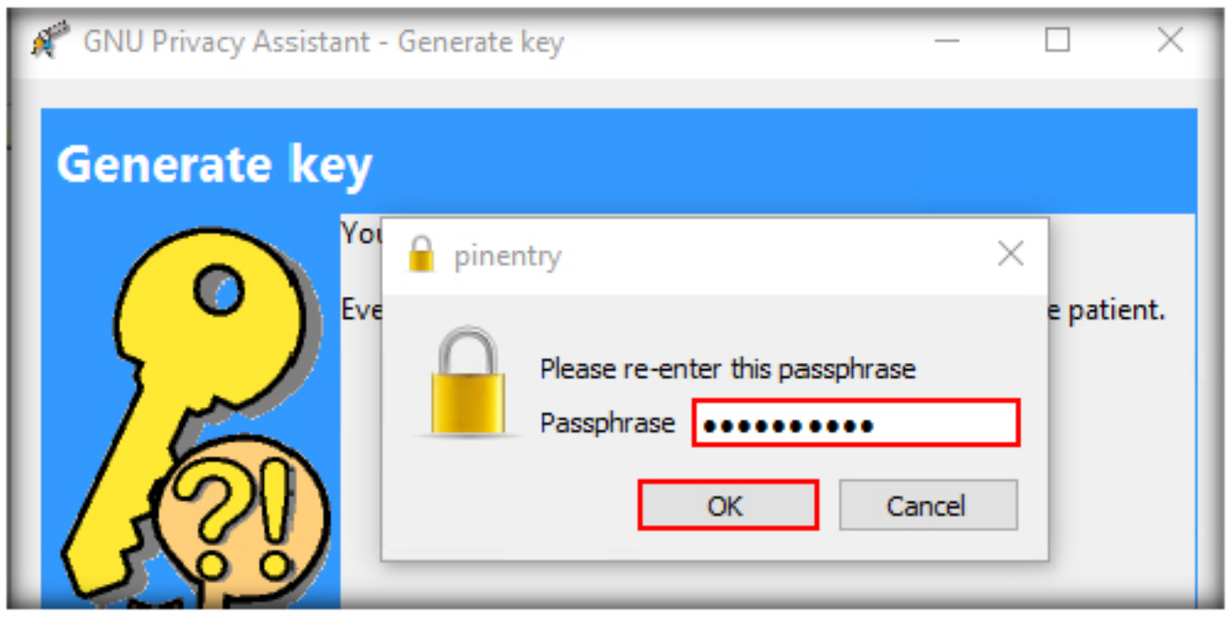

FIGURE 3.32: Re-Entering Passphrase

40. The **Backup key to file** window appears. Select the **PGP keys** folder on the Desktop (created in step 31) and click **Save**.
41. A Pop up window appears. Click **Close**.

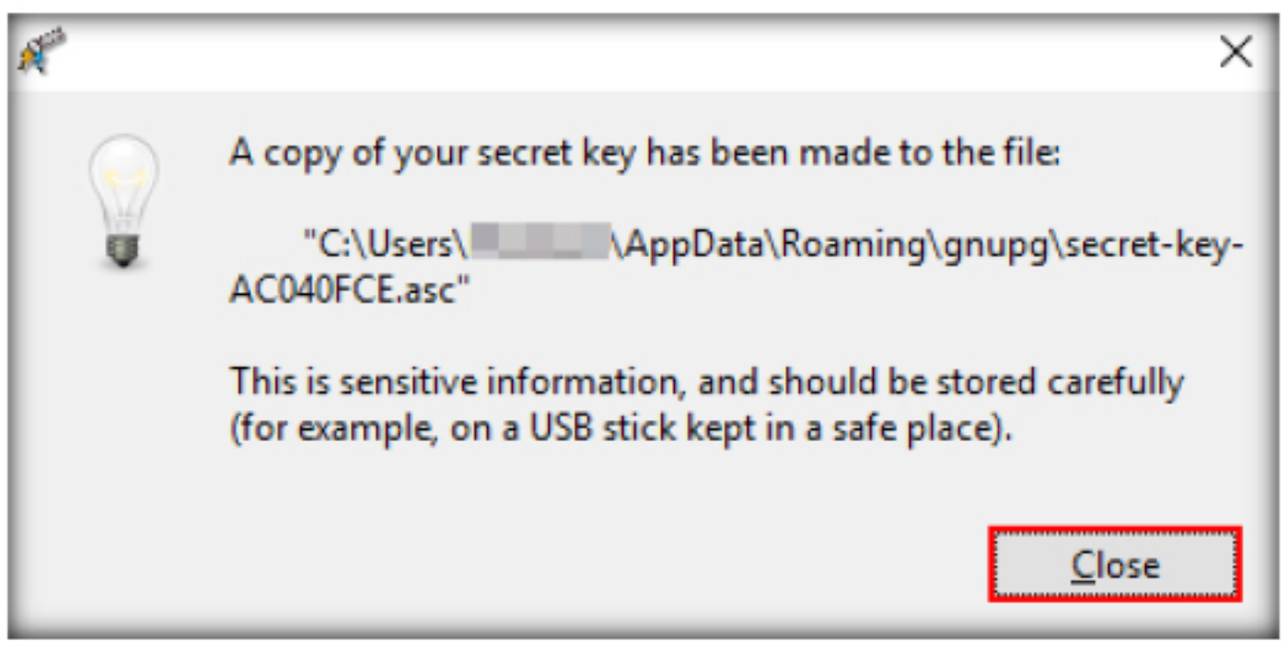

FIGURE 3.33: Secret Key Copied

Note: Due to some restrictions, sometimes the PGP key is stored in the default location, and the location given in these steps is ignored. Make a note of the location to retrieve the key when needed.

42. You can see the new key appear in the GPG window.

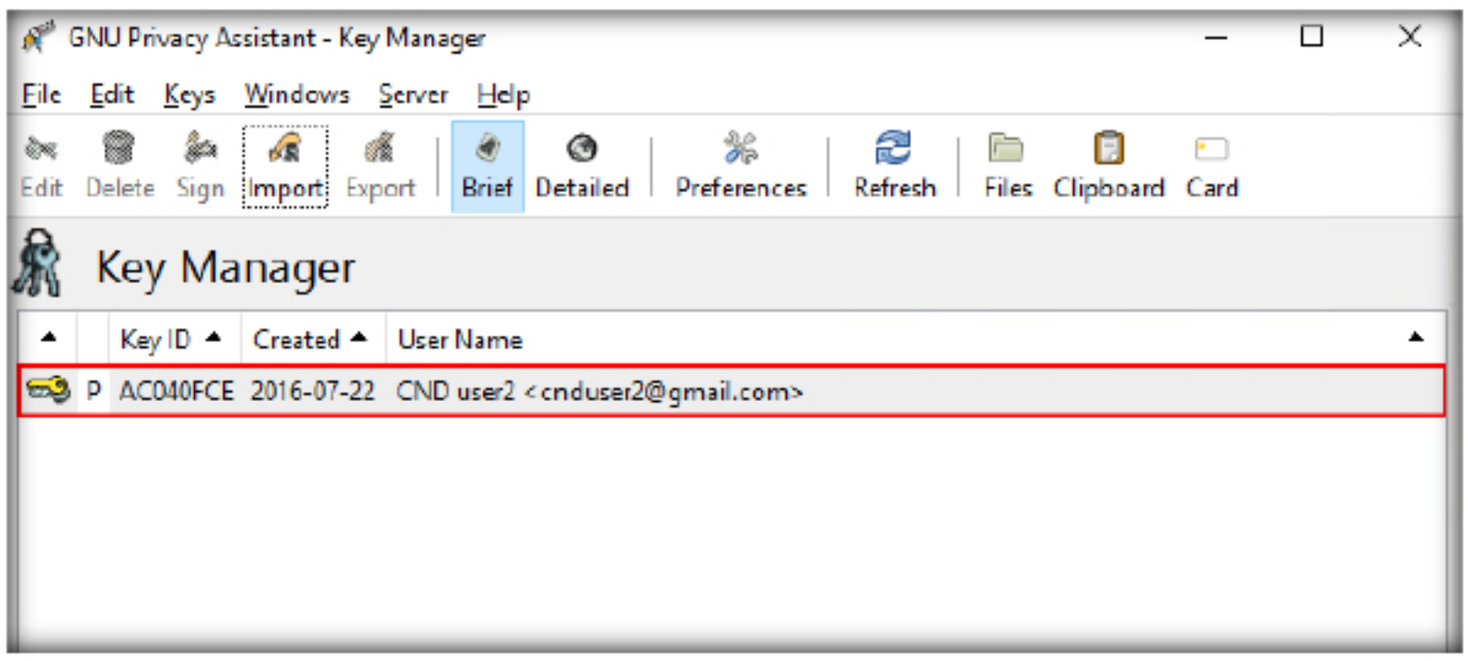

FIGURE 3.34: CND User2 Key Created

TASK 6

Paste the Public Key

43. Minimize the window and press **Windows + R** keys. The **Run** window appears. Enter **\\10.10.10.12** in the **Open:** field and click **OK**.

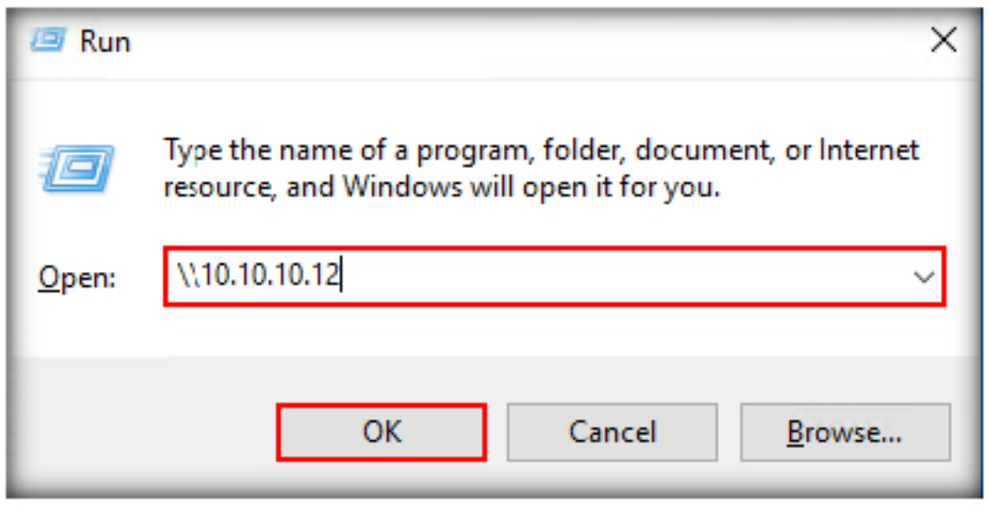

FIGURE 3.35: Launching Run

44. The **Windows Security** appears. Enter the credentials for the **Windows Server 2012** and click **OK**. The credentials are **Administrator/Pa$$w0rd**.

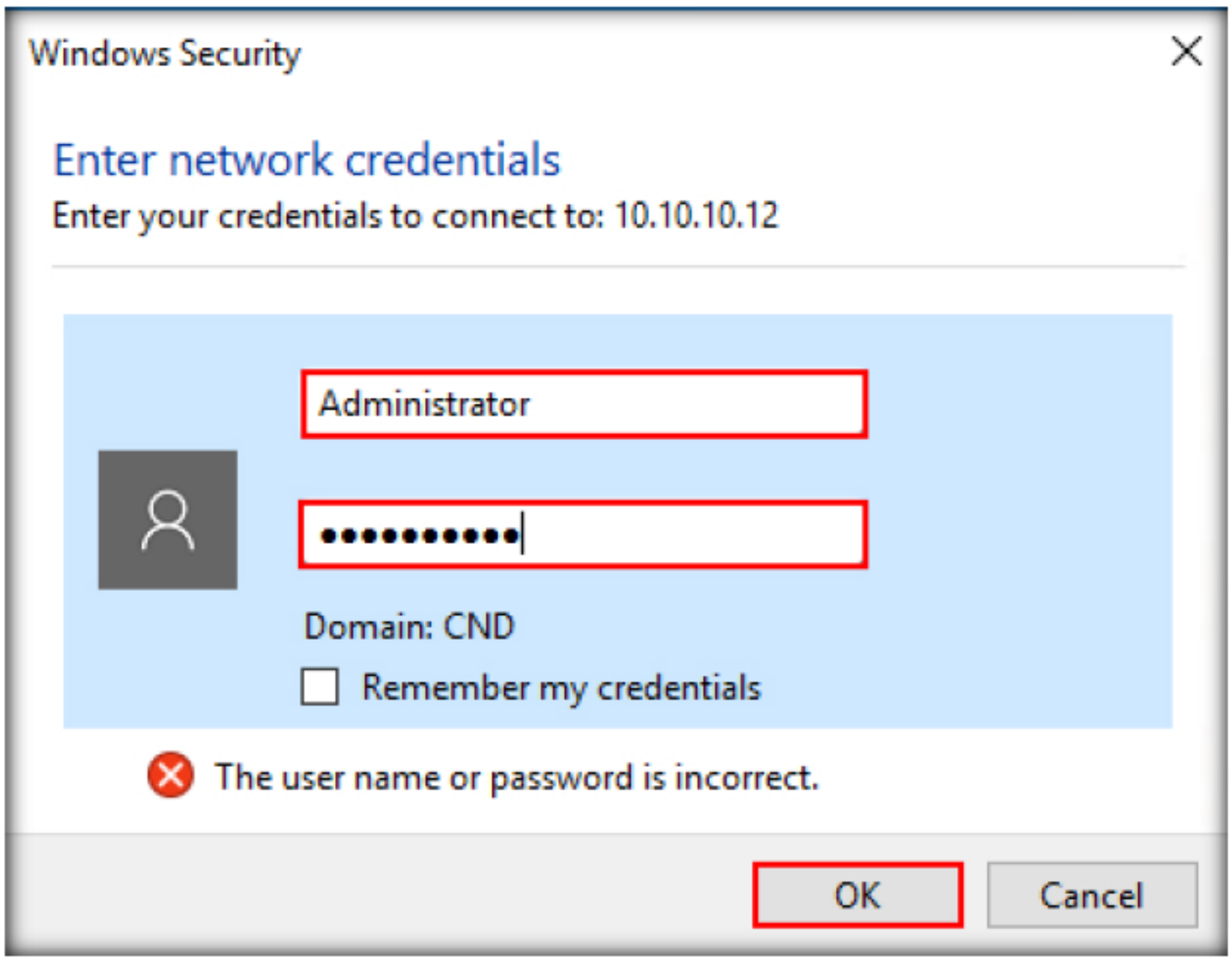

FIGURE 3.36: Entering Windows Server 2012 Credentials

45. Navigate to the Windows Server 2012 machine's **Desktop**, open the **Public Keys** folder and launch the **cnd user1 public keys.txt** file.

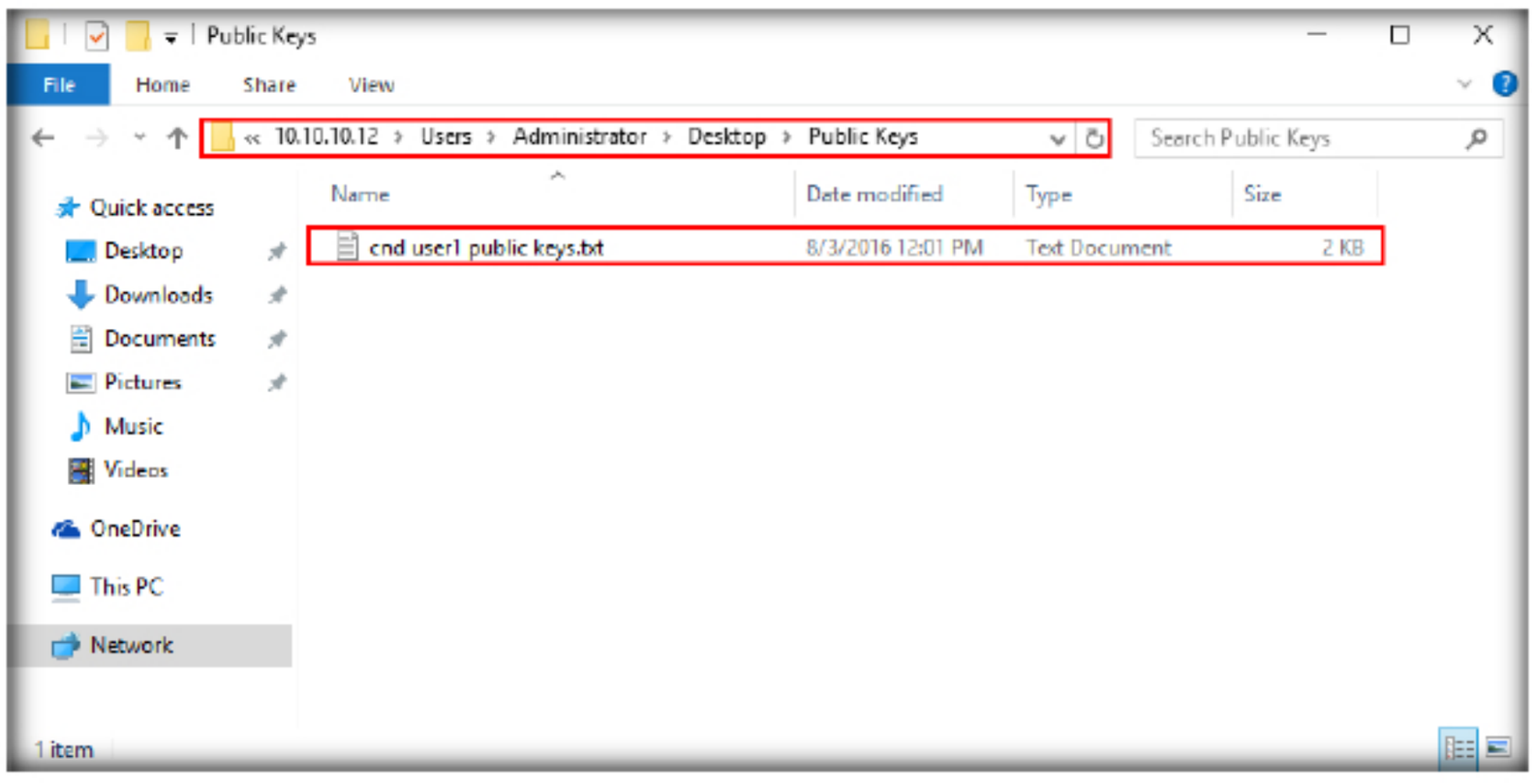

FIGURE 3.37: Navigating to User1 Publiv Keys

46. Hit the **Ctrl + A** keys to select all of the contents in the file. Right click and select **Copy** from the context menu.

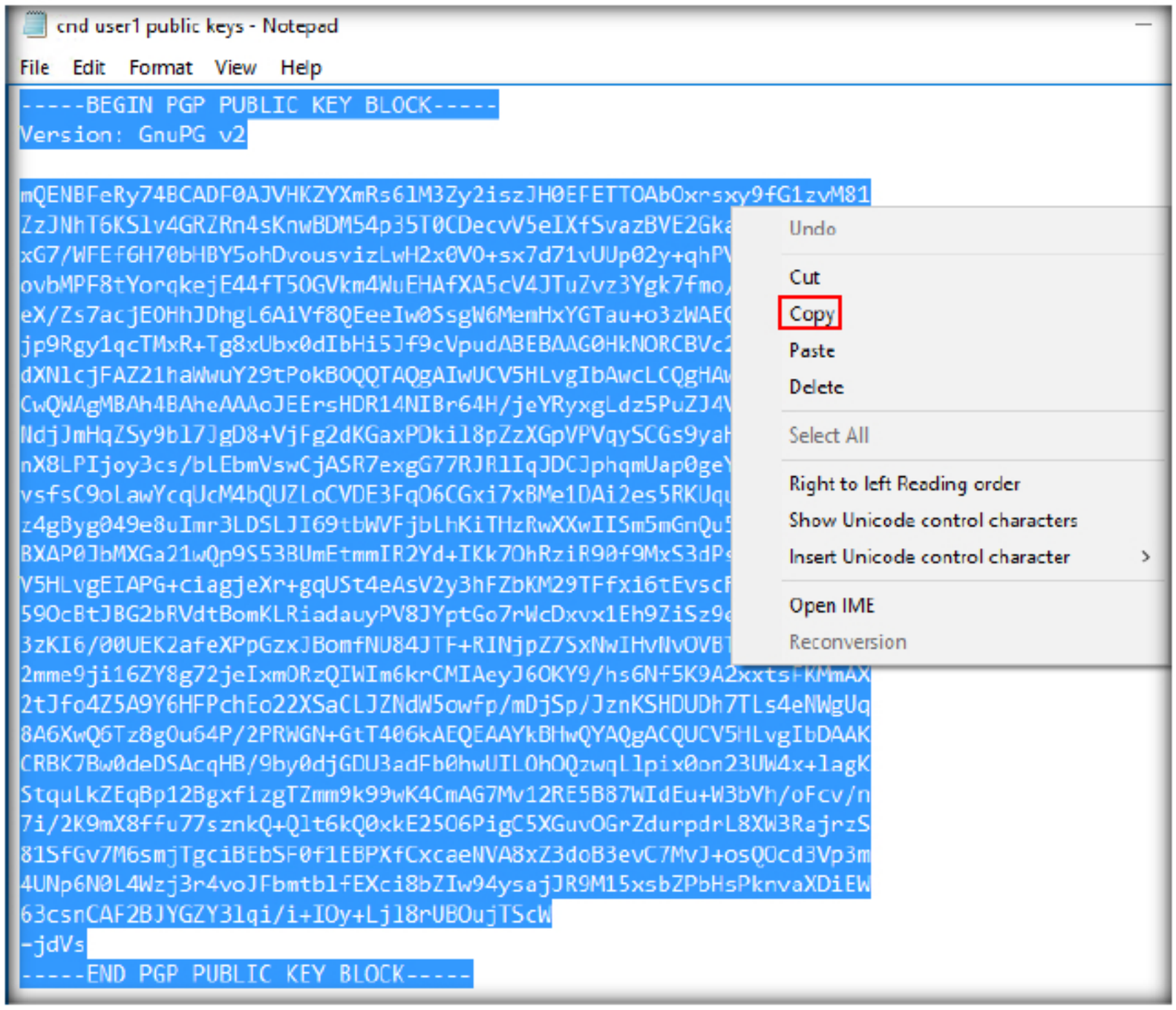

FIGURE 3.38: Copying User1 Public Keys

47. Switch back to the GNU privacy assistant window.

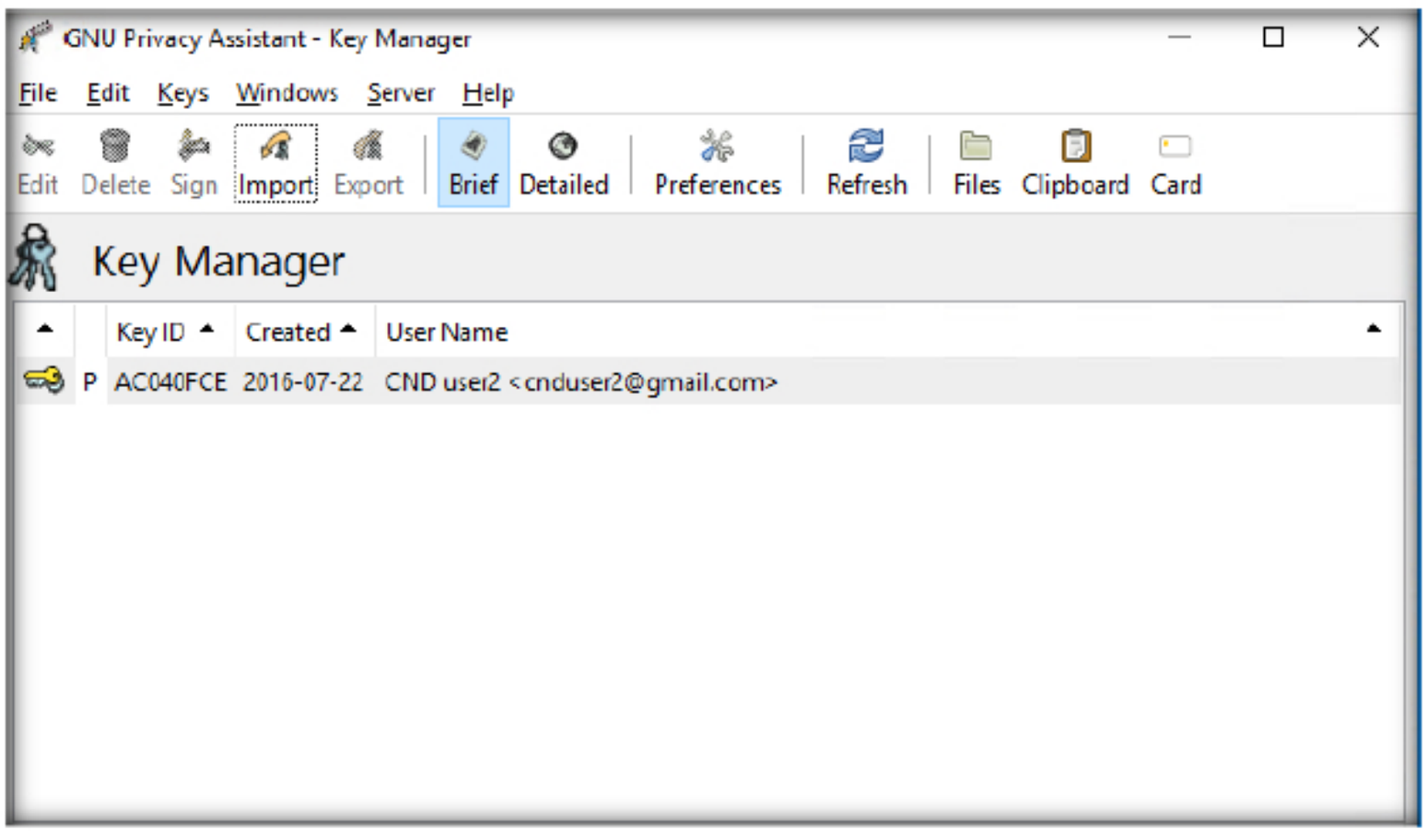

FIGURE 3.39: GNU Privacy Assistant Window

48. Hit the **Ctrl + V** keys to paste the copied content. The CND user1's public keys appear. You also get a pop up window. Click **Close**.

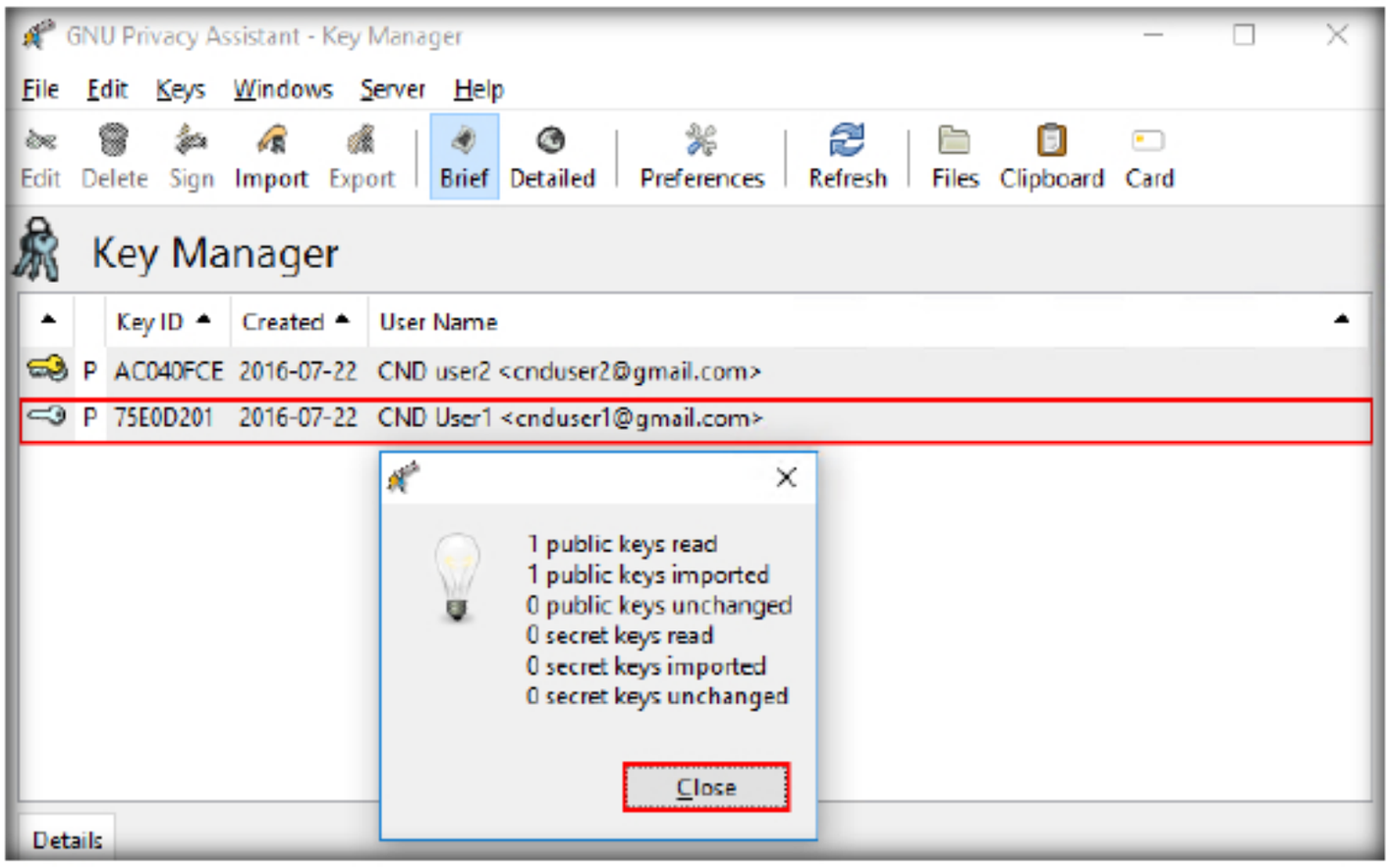

FIGURE 3.40: User1 Public Keys Imported

49. Next, is to send some data from the Windows 10 machine to the Windows Server 2012 machine. The data will be encrypted using Gmail.

50. Click **Clipboard**.

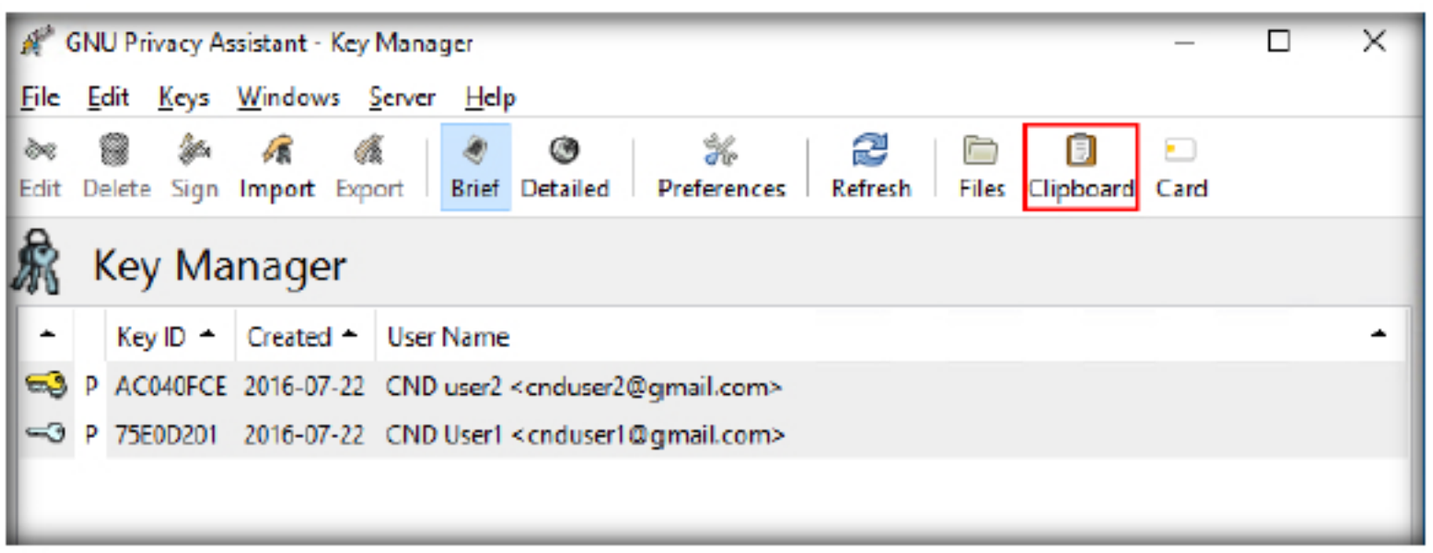

FIGURE 3.41: Navigating to Clipboard

TASK 7

Encrypt the Email Communication

51. The Clipboard window appears. Enter random data.

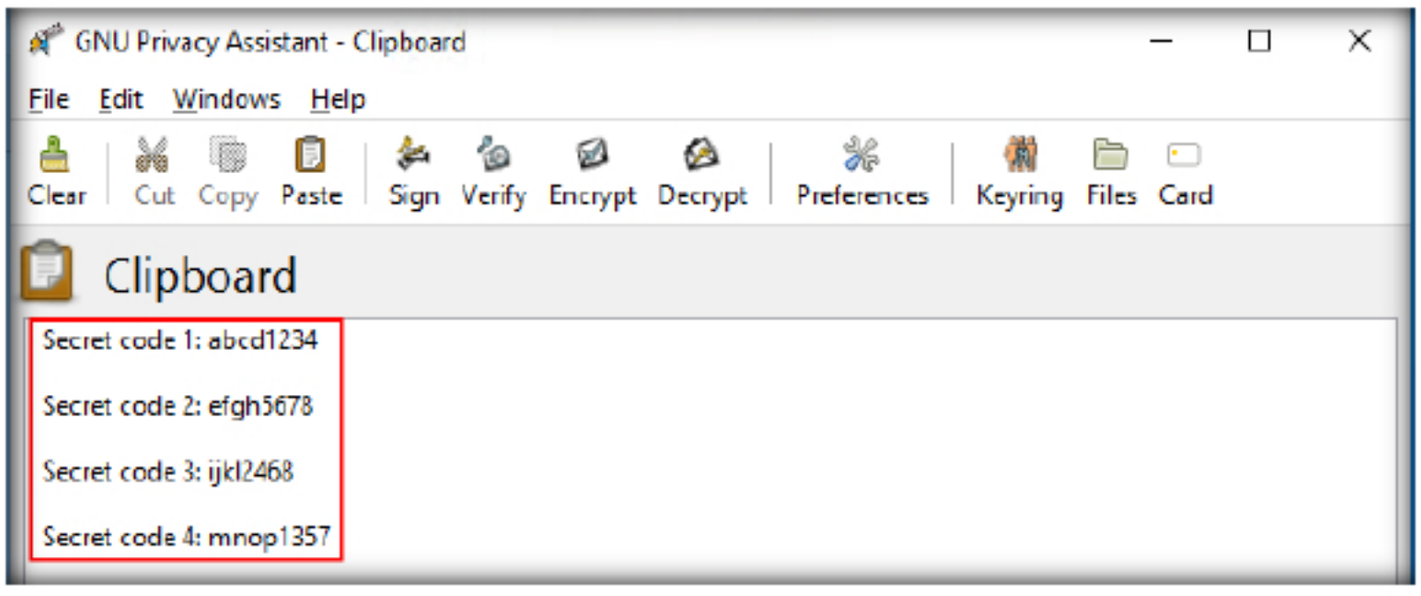

FIGURE 3.42: Entering Data in Clipboard

52. Click **Sign**.

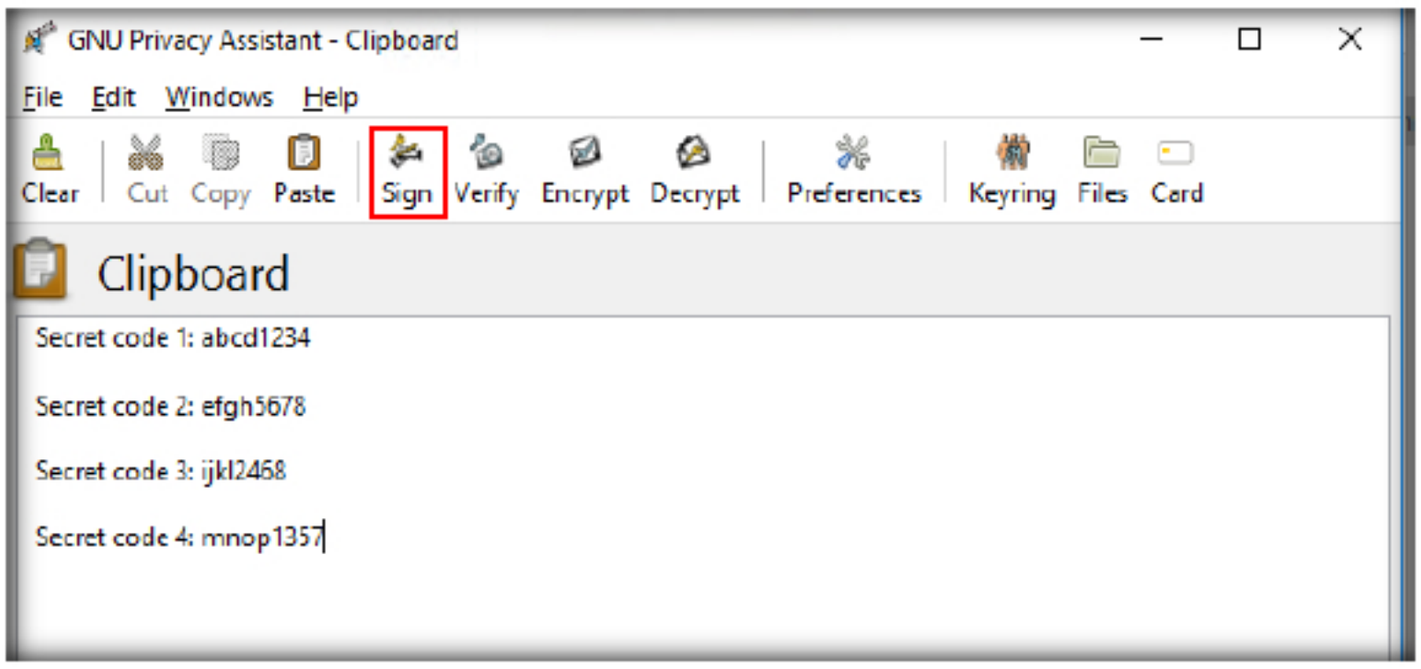

FIGURE 3.43: Signing the Data

53. The **Sign documents** window appears. Select **user 2** and click **OK**.

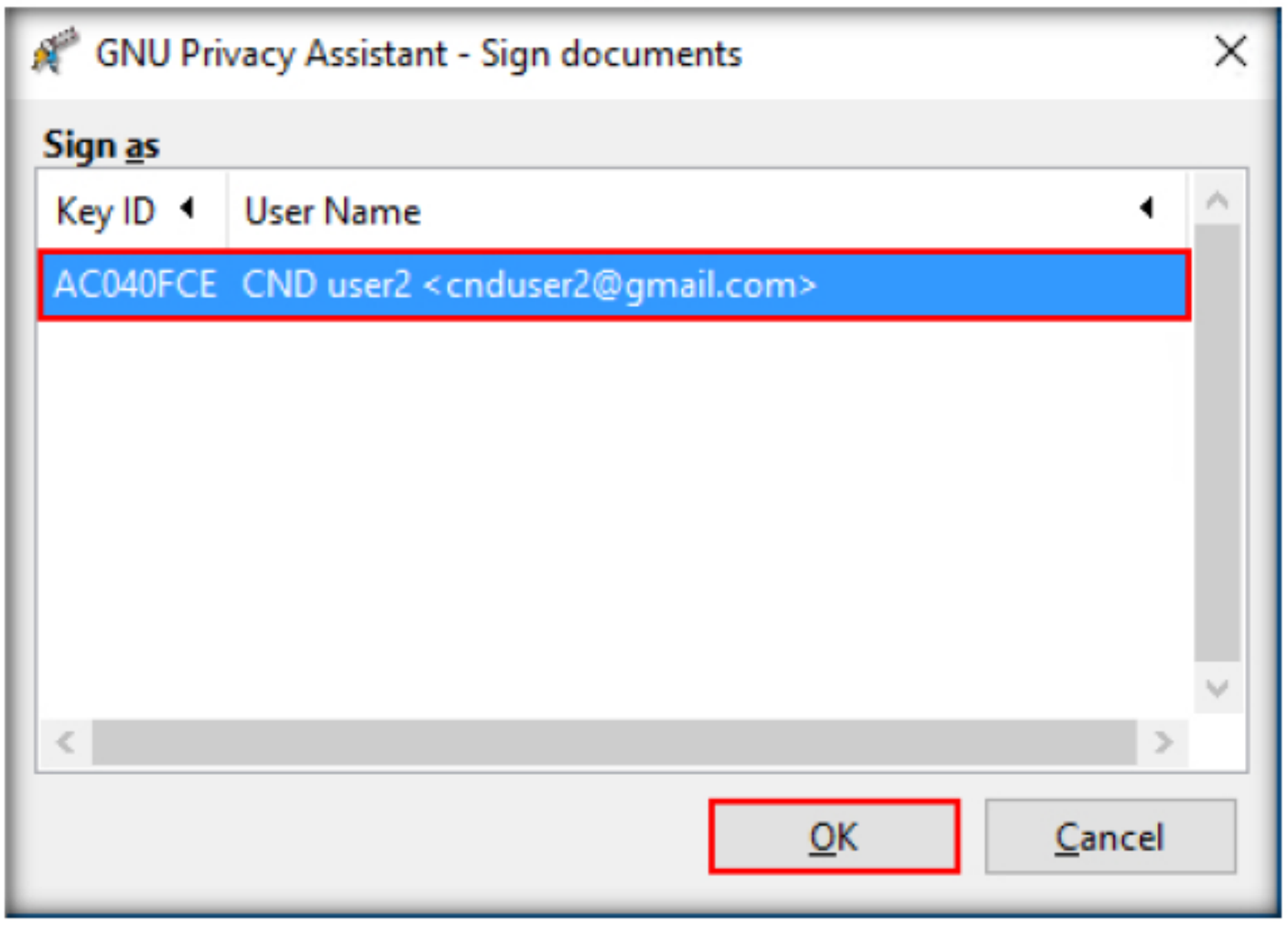

FIGURE 3.44: Signing the Document

54. The **pinentry** window appears. Enter the passphrase for **user2** (qwerty@321) and Click **OK**.

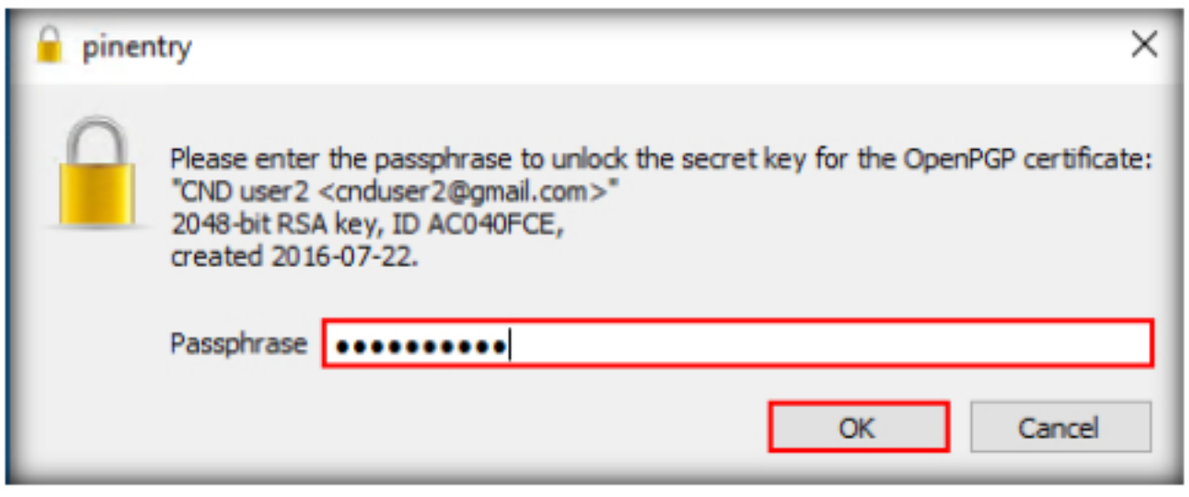

FIGURE 3.45: Entering Passphrase of User2

55. You will see the signature appear at the bottom of the message in an encrypted form.

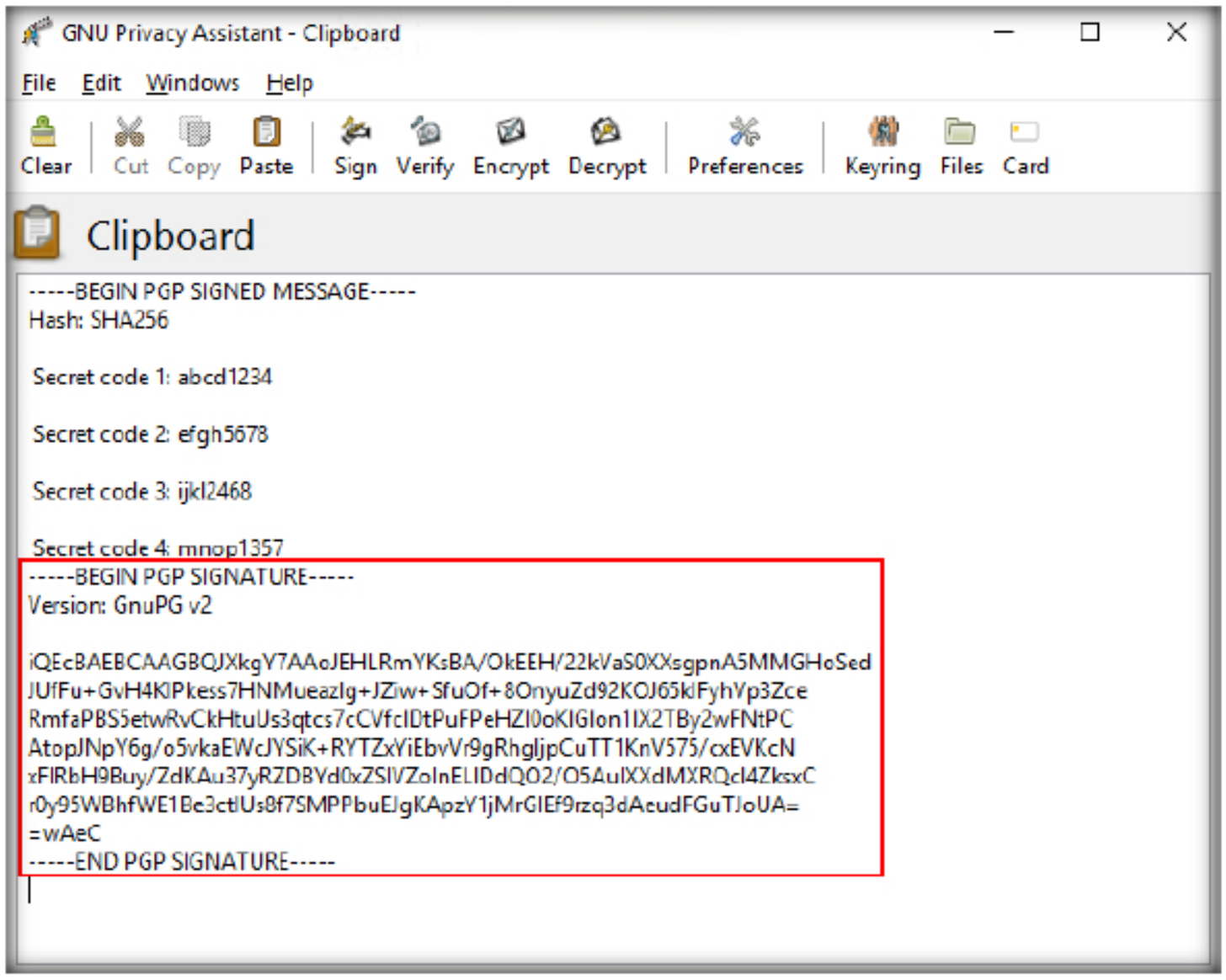

FIGURE 3.46: Signature Created

56. Click **Encrypt**.

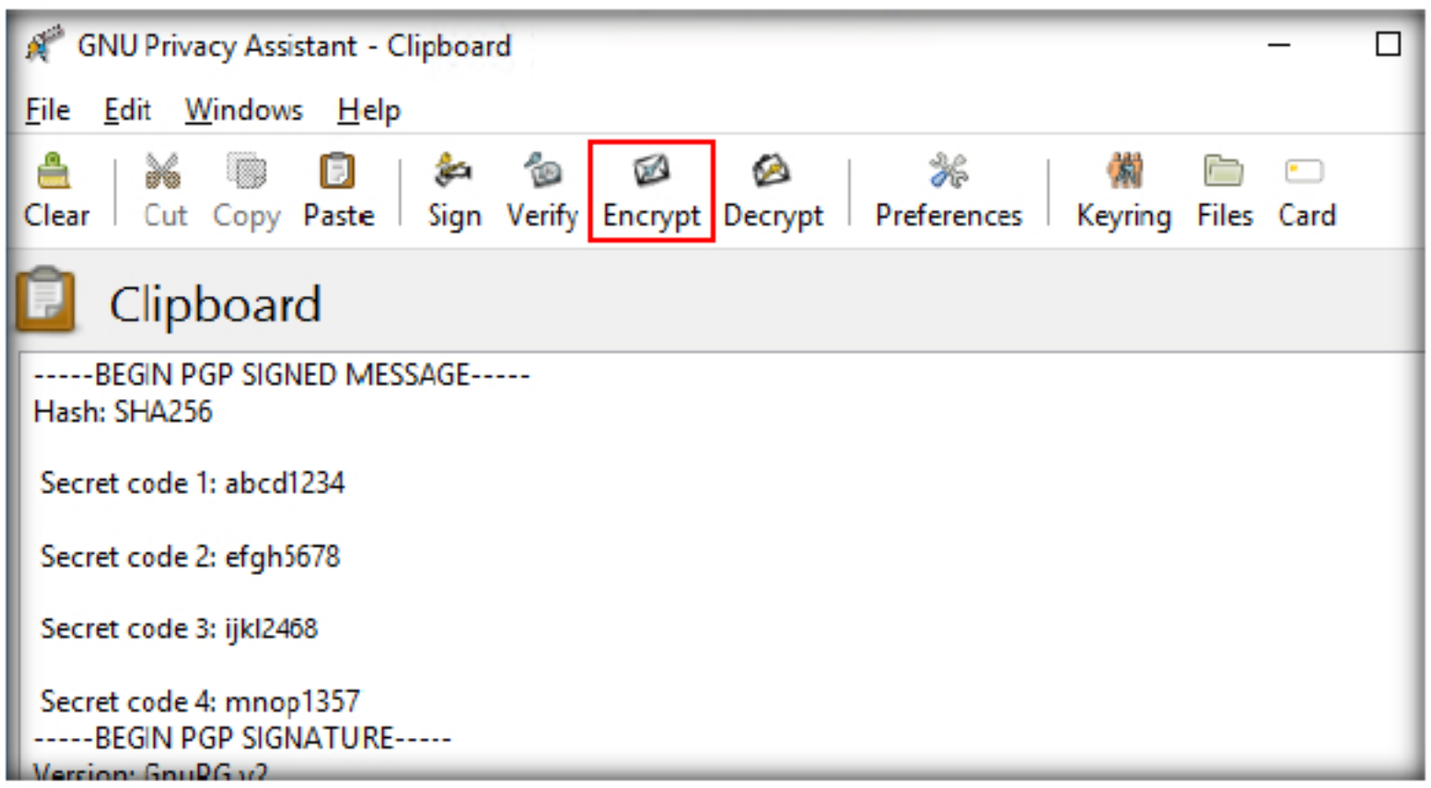

FIGURE 3.47: Navigating to Encrypting the data

57. The **Public keys** window appears. Select **CND user1** and click **OK**.

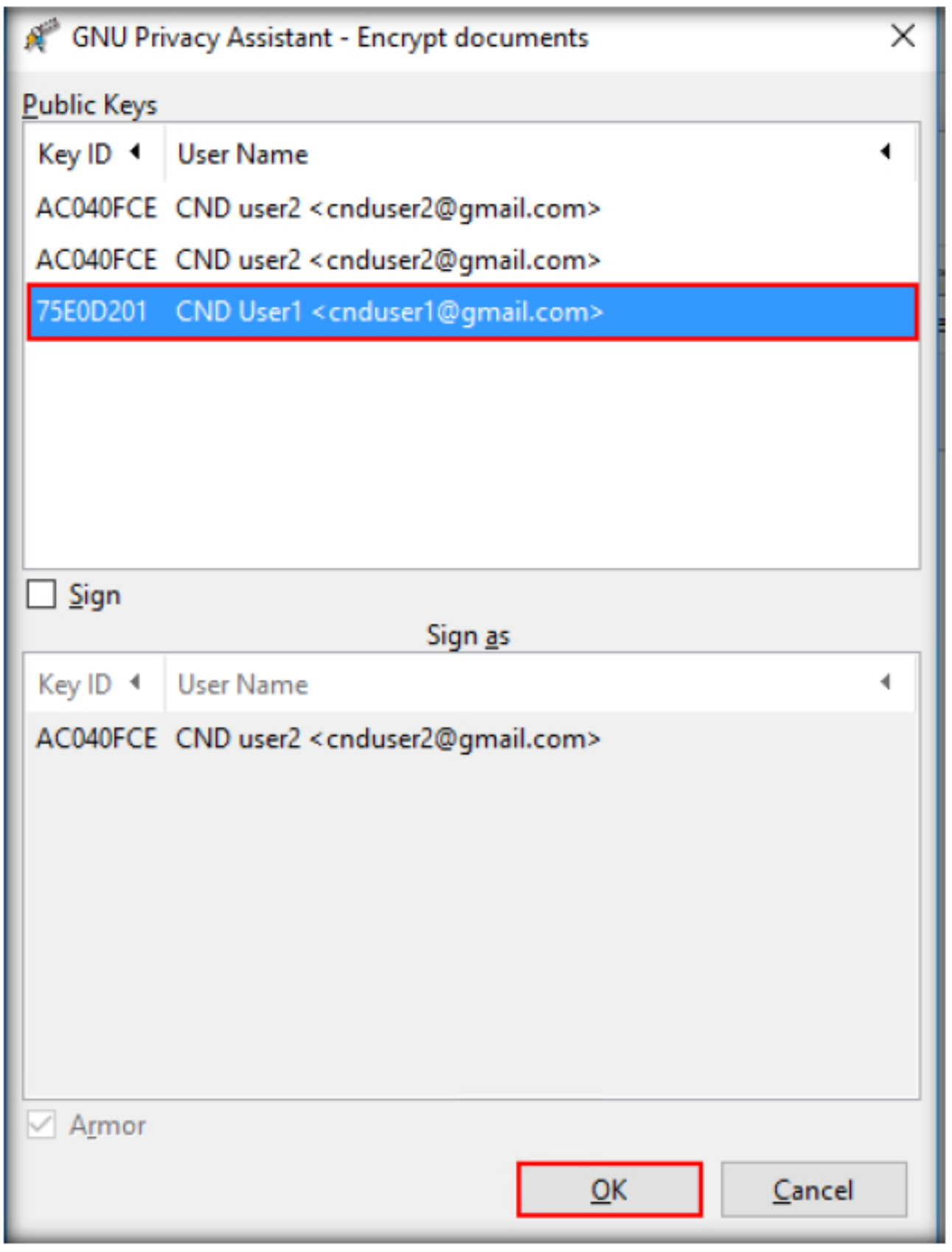

FIGURE 3.48: Encrypting with CND User1 Public Keys

58. When the **Unknown Key** window pops up. Click **Yes**.

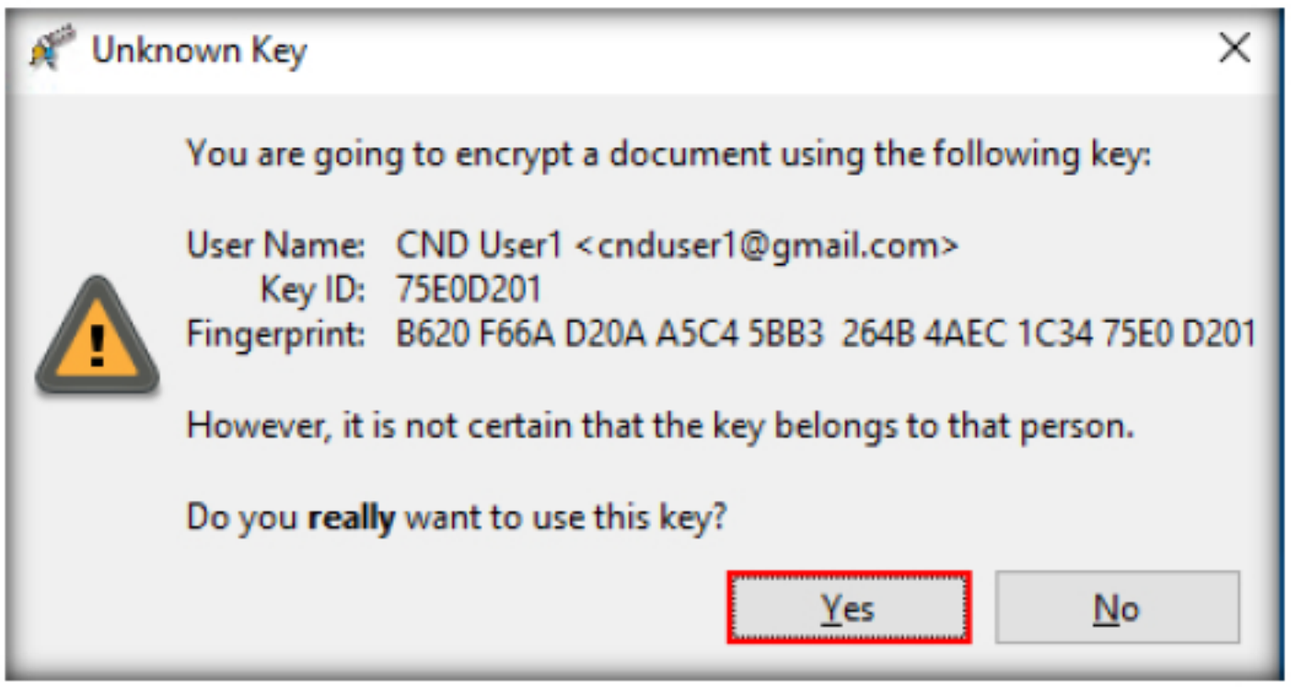

FIGURE 3.49: Unknown Key Window

59. You can see that the message was encrypted.

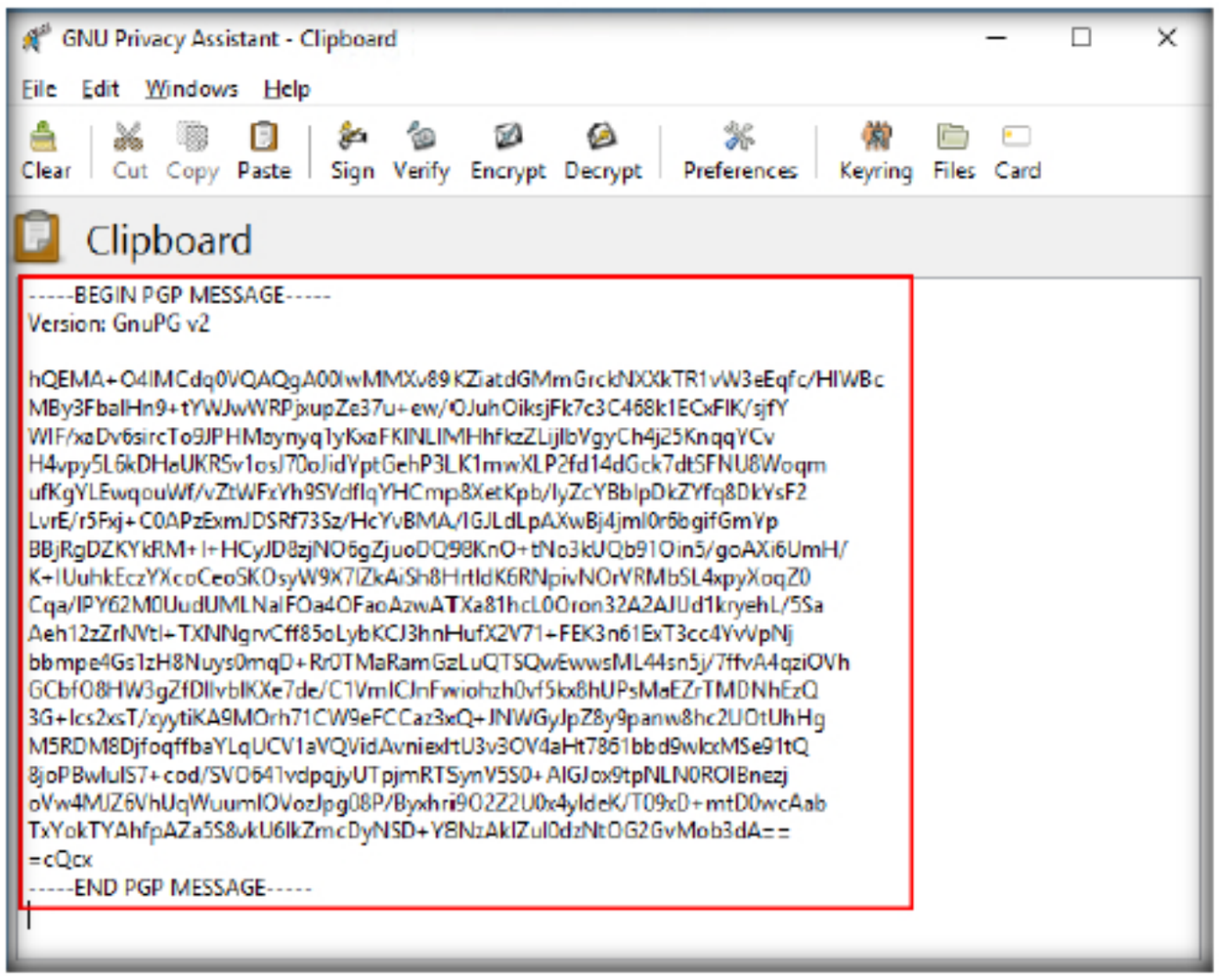

FIGURE 3.50: Message Encrypted

60. Next, transmit this message to **CND User1** using the Gmail account for **CND user 2** or any other insecure media over the Internet.

61. Copy the entire encrypted message, log into Gmail with your second email ID (here, we are using the **cnduser2** user account) and compose a mail to the first email ID (here, we are using **cnduser1@gmail.com**). Paste the encrypted message and click **Send**.

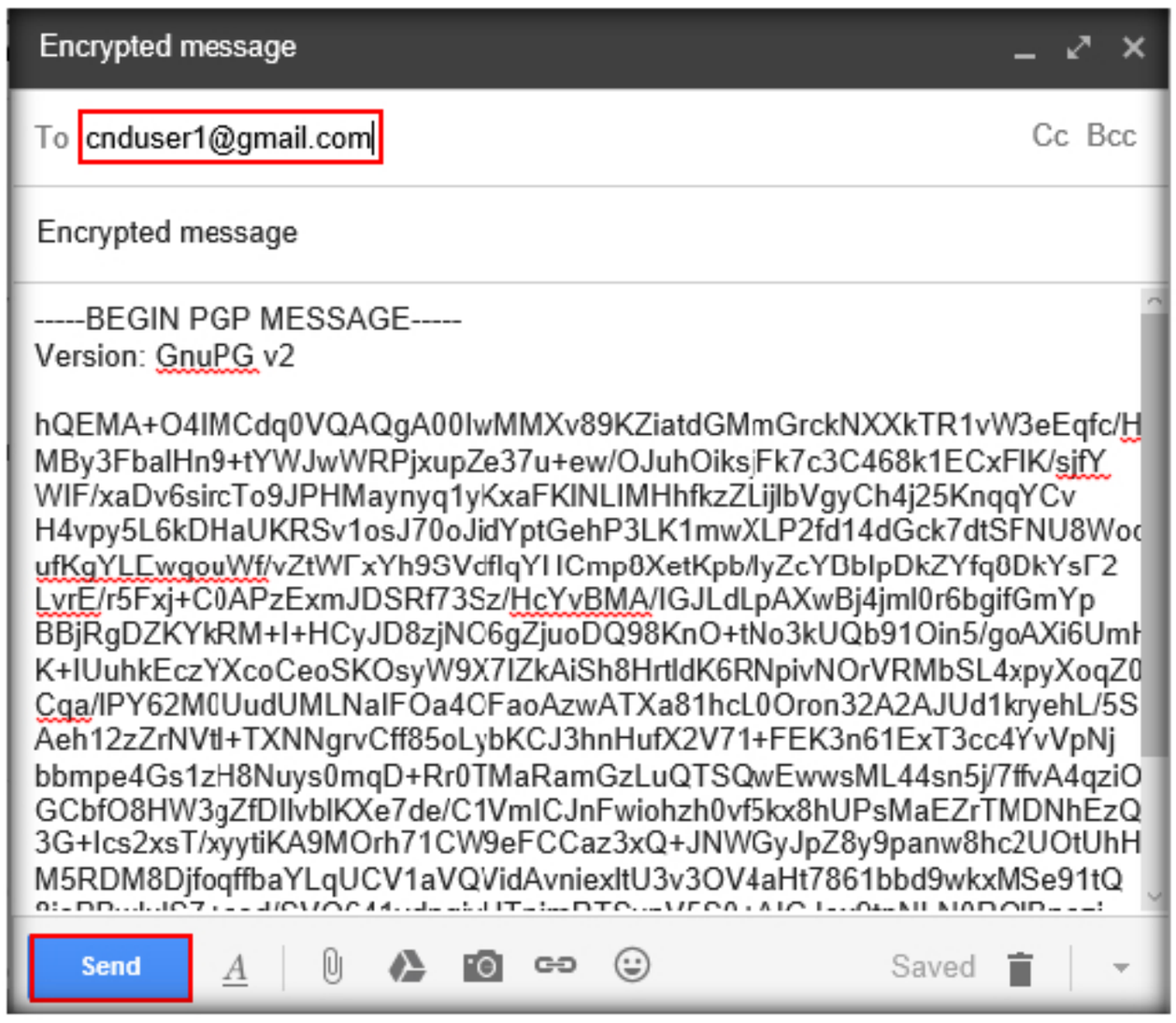

FIGURE 3.51: Mail to User1

62. Switch to the Windows Server 2012 machine and log into your first Gmail account (here, we are logging in to **cnduser1@gmail.com**). You can see the mail from your second email account (in this case, **cnduser2@gmail.com**). Next, open the mail.

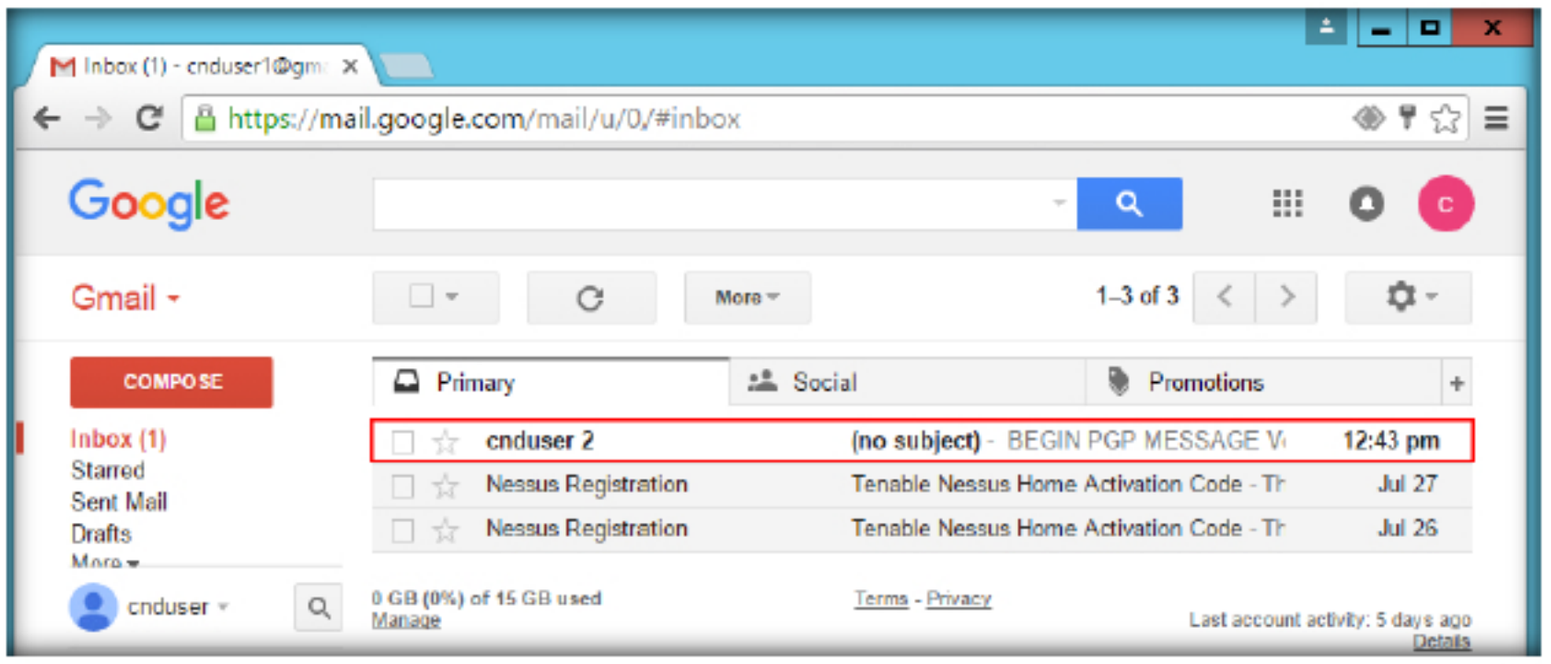

FIGURE 3.52: Mail from User2

TASK 8

Decrypt the Email Content

63. Select the entire message, right click and select **Copy** from the context menu.

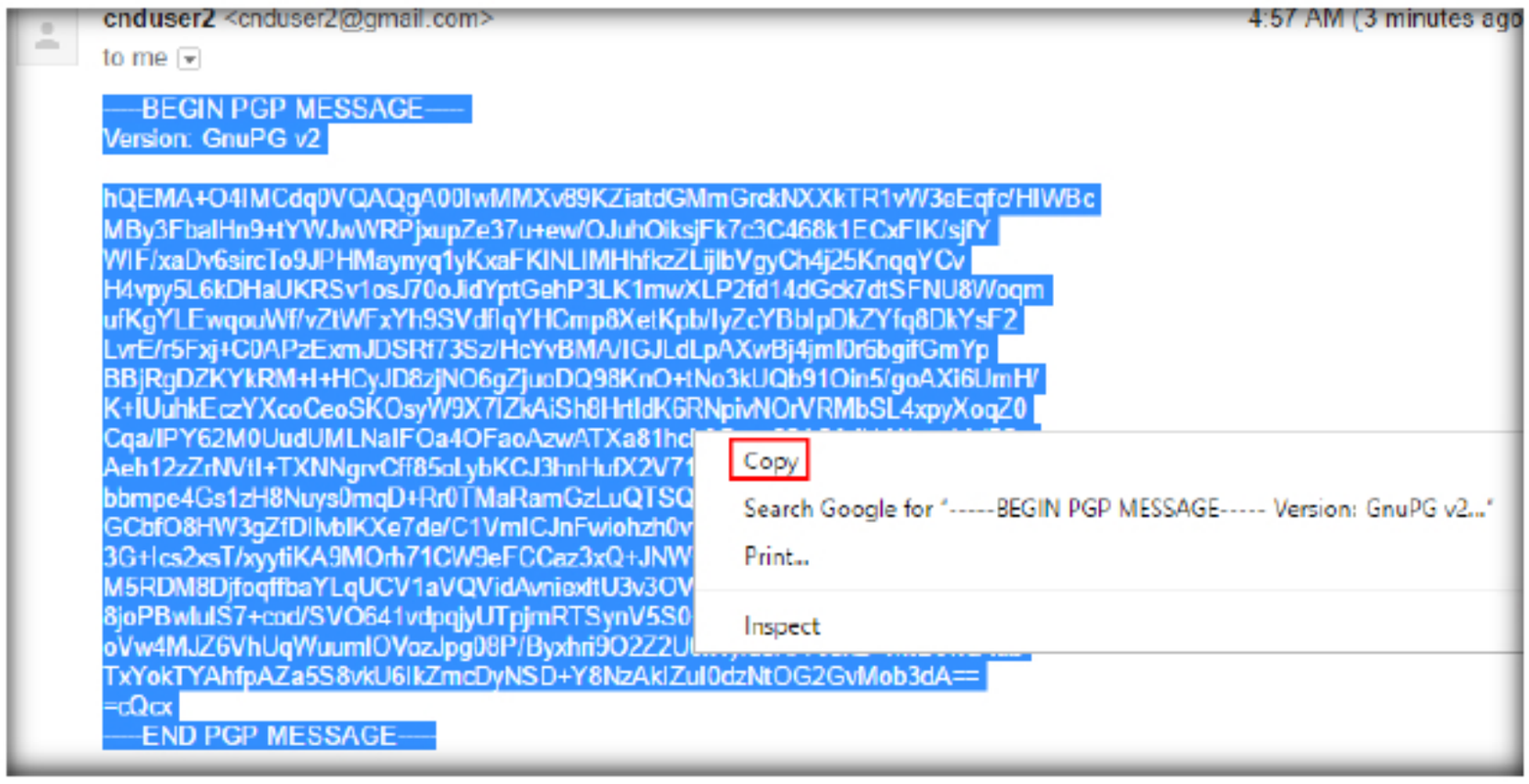

FIGURE 3.53: Copying the Message

64. Switch to the **GNU Privacy Assistant** window then click **Clipboard**.

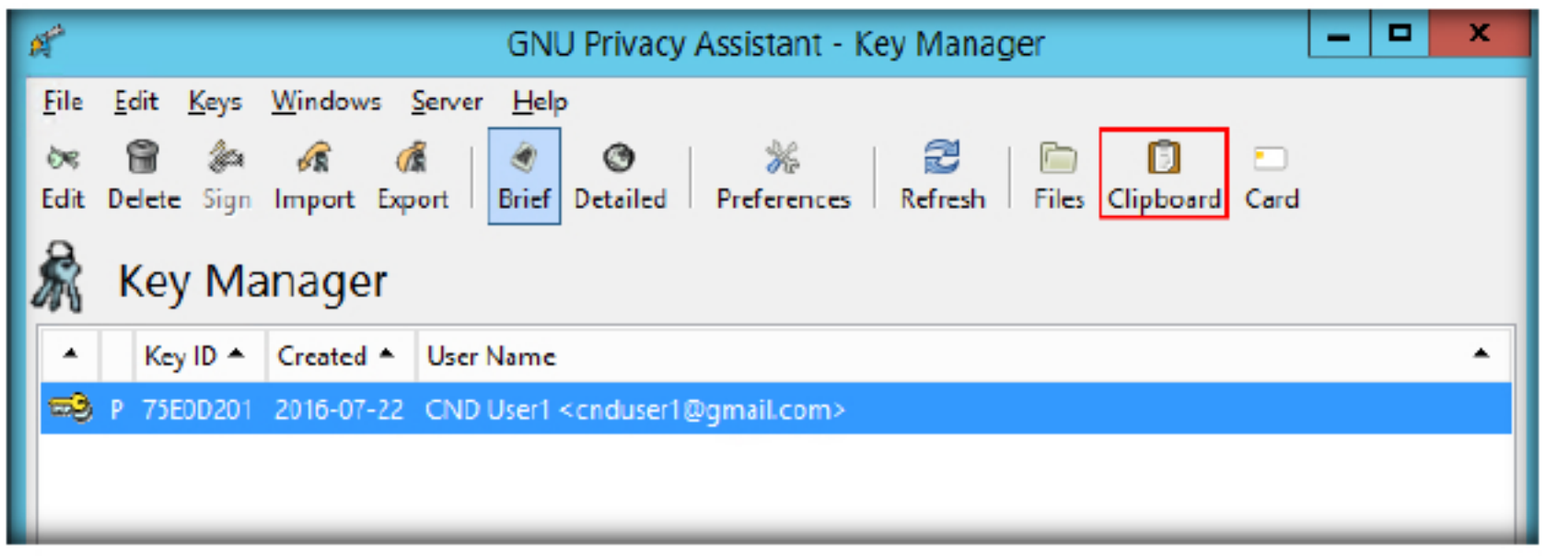

FIGURE 3.54: Navigating to User1's Clipboard

65. The **Clipboard** window appears. Right click and select **Paste**.

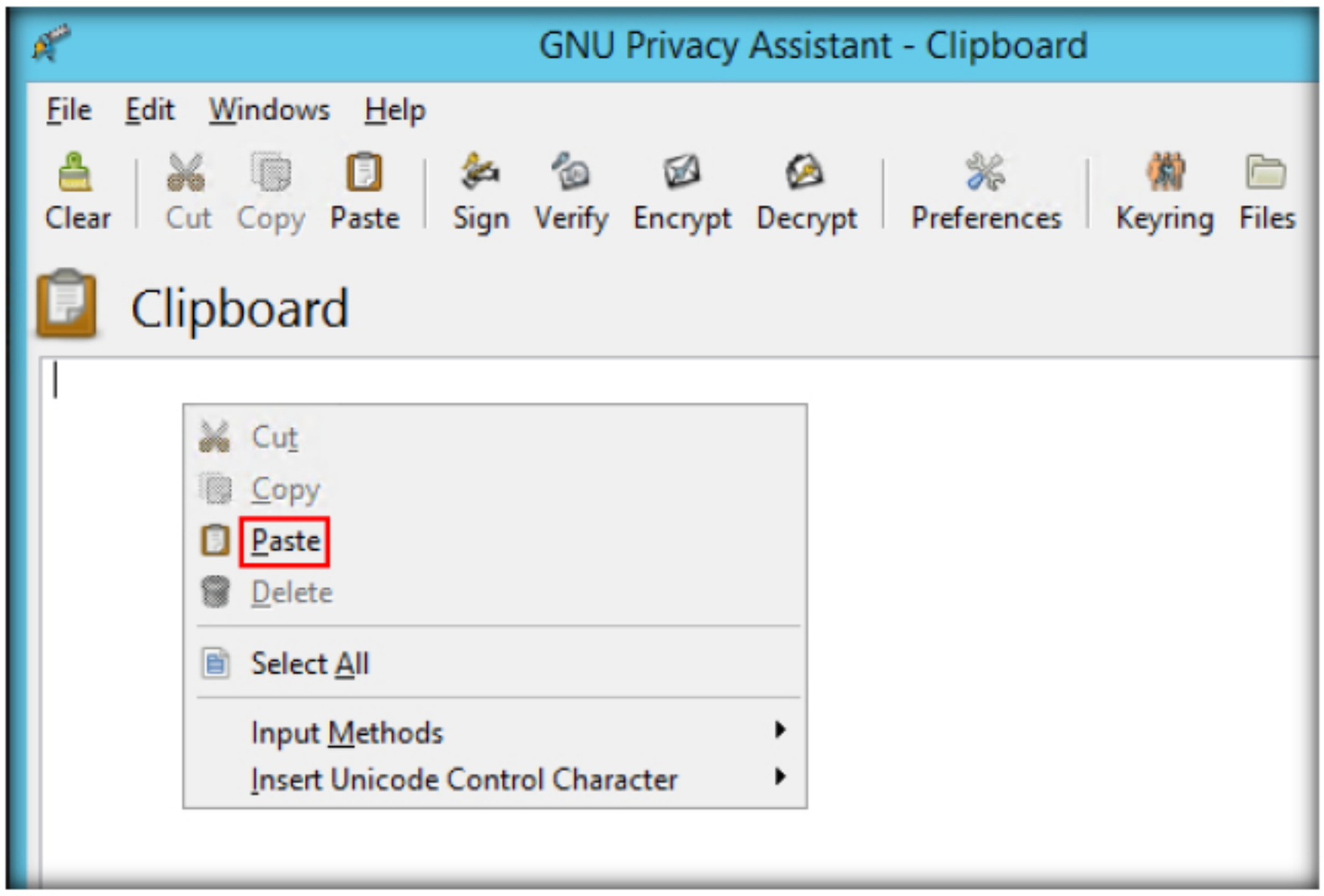

FIGURE 3.55: Pasting the Encrypted Message

66. The encrypted message is pasted. Click **Decrypt**.

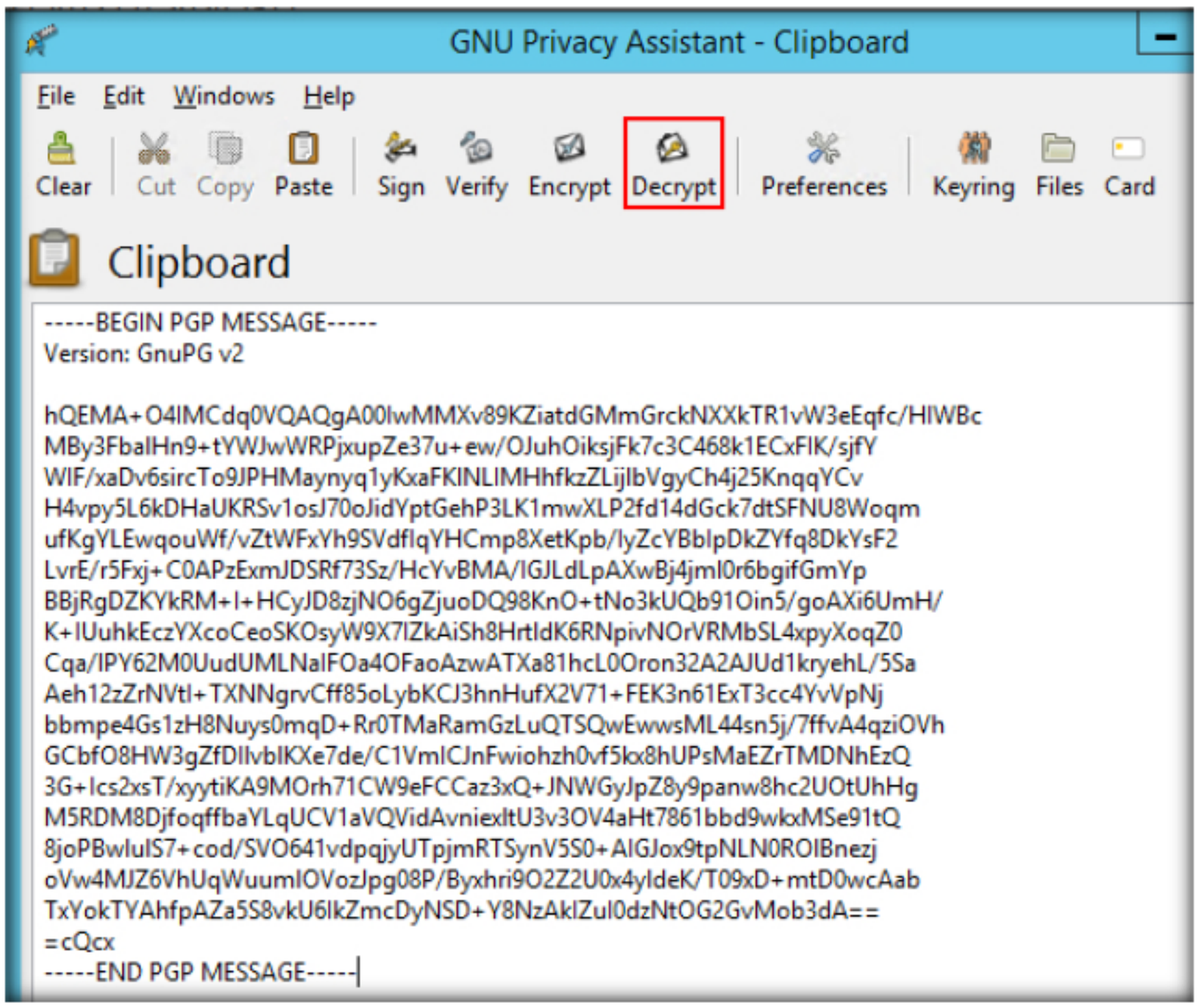

FIGURE 3.56: Decrypting the Message

67. The **pinentry** window appears. Enter the Passphrase for the **CND user1** (qwerty@123) then click **OK**.

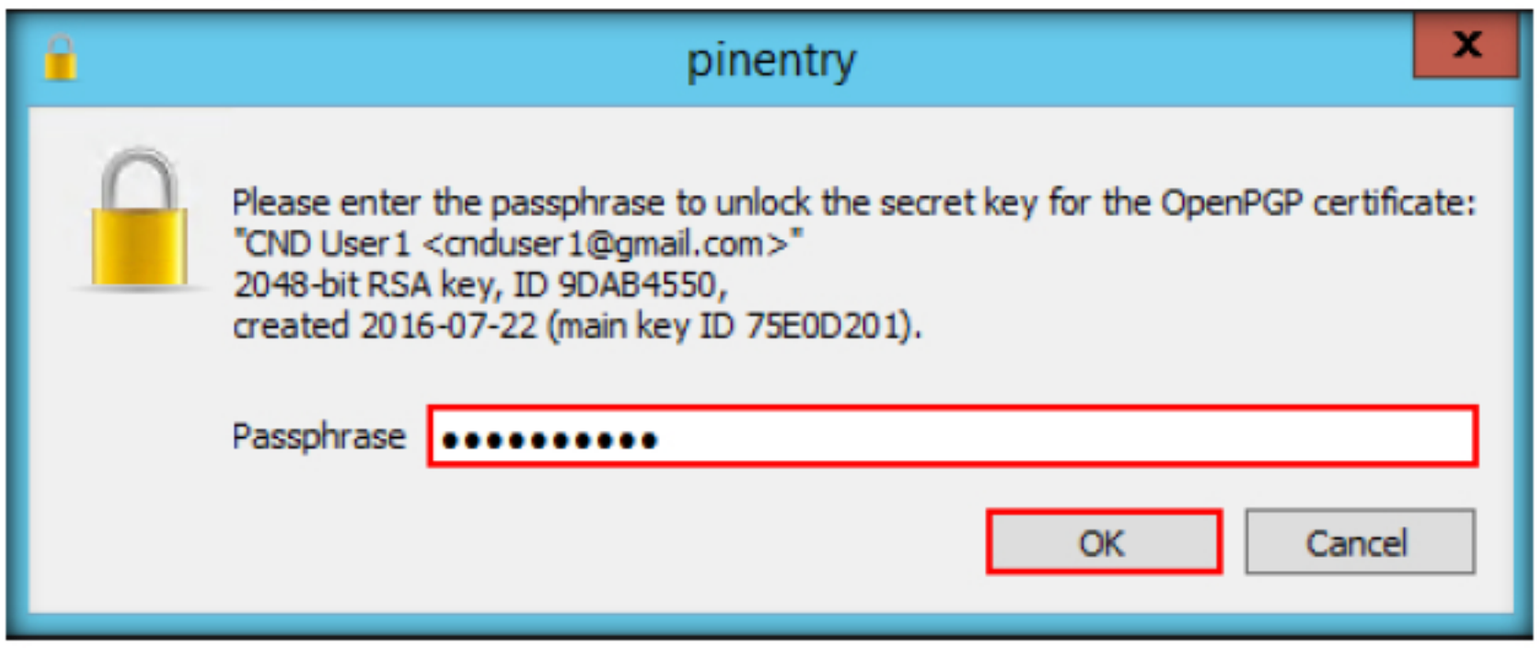

FIGURE 3.57: Entering User1 Passphrase

68. You can see the decrypted message appear on the clipboard.

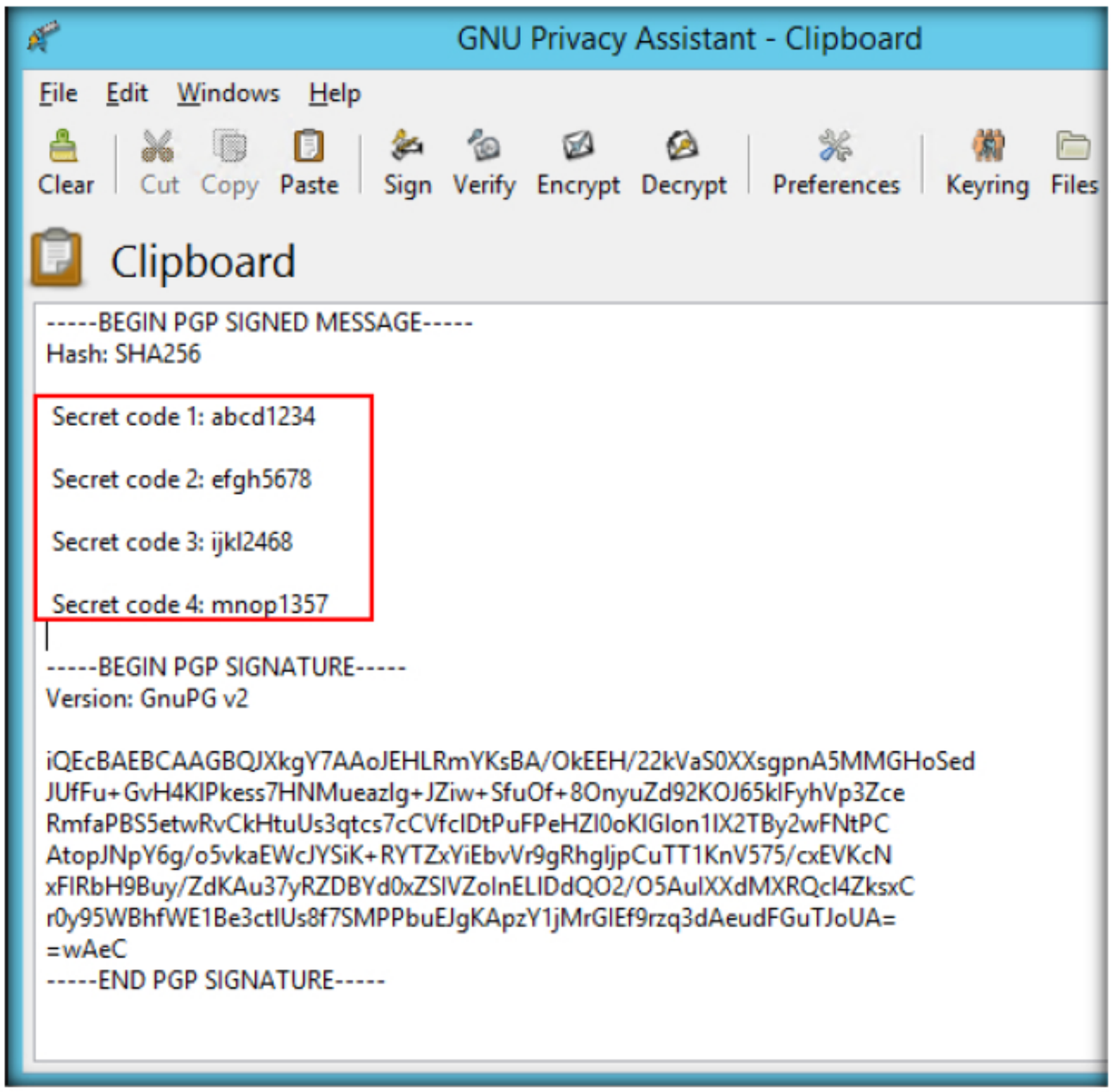

FIGURE 3.58: Message Decrypted

Note: CND user1 encrypted a message with CND user1's public key and sent it to the second user. If the message is encrypted with CND user1's public key it can only be decrypted with CND user1's private key (passphrase). Similarly, you can get CND user2's public key on Server 2012 and send an encrypted message using that public key and then navigate to the Windows 10 machine and decrypt it with CND user2's private key.

Lab Analysis

Analyze and document the results of the lab exercise. Give your opinion on your target's security posture and exposure through free public information.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS ABOUT THIS LAB.

Internet Connection Required	
☑ Yes	☐ No
Platform Supported	
☑ Classroom	☐ iLabs