


Cryptography


Module 20


Cryptography


Cryptography is the study and art of hiding meaningful information in an unreadable format.

ICON KEY

 Valuable information

 Test your knowledge

 Web exercise

 Workbook review


Lab Scenario

Data security is critical to online business and privacy of communication. Today's information-based organizations extensively use Internet for e-commerce, market research, customer support, and a variety of other activities.

With this increasing adoption of Internet–World Wide Web use for business and personal communication, securing sensitive information such as credit-card numbers, personal identifiable information, bank account numbers, secret messages, and so on is becoming increasingly more important.

The ability to protect and secure information is vital to the growth of electronic commerce and to the growth of the Internet itself. Many people need or want to use communications and maintain data security. The encryption of data plays a major role in doing so. For example, banks all over the world use encryption methods to process financial transactions involving the transfer of huge amounts of money. They also use encryption methods to protect their customers' ID numbers at bank automated teller machines. Many companies and even shopping malls sell anything from flowers to wine over the Internet, and these transactions are made through credit cards and secure Internet browsers that include encryption. And it becomes vital to assure their Internet customers that their credit-card information and other financial details will remain private and secure. But this can only be accomplished by the use of strong and impenetrable encryption methods.

As part of a security assessment, you have to suggest to your target organization how it can use proper encryption techniques to protect data, both in storage and during transmission. The labs in this module demonstrate the use of encryption to protect information systems.

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

Lab Objectives

This lab will show you how to use encryption tools to encrypt data. It will teach you how to:

- Use encrypting/decrypting techniques
- Generate Hashes and checksum files

Lab Environment

To complete this lab, you will need:

- A computer running Window Server 2016
- A computer running Windows 10 in virtual machine

Module 20 - Cryptography

- A computer running Windows Server 2012 in virtual machine
- A computer running Kali Linux in virtual machine
- A Web browser with Internet access
- Administrative privileges to run the tool

Lab Duration

Time: 85 Minutes

Overview of Cryptography

Cryptography is the practice and study of hiding information. Before the modern age Cryptology, almost synonymous with encryption, was the conversion of information from a readable state to one which was apparently illegible. Modern cryptography intersects the disciplines of mathematics, computer science, and electrical engineering and helps in securing data from being intercepted and compromised during online transmissions. This module provides a comprehensive understanding of different crypto systems and algorithms, one-way hash functions, public-key infrastructure (PKI), and the different ways in which cryptography can help in ensuring privacy and security of online communication. The module also covers the various cryptography tools used to encrypt sensitive data.

Lab Tasks

TASK 1

Overview

Recommended labs to assist you in cryptography are:

- Calculating One-way Hashes using **HashCalc**
- Calculating MD5 Hashes using **MD5 Calculator**
- Understanding File and Text Encryption using **CryptoForge**
- Basic Data Encryption using **Advanced Encryption Package**
- Encrypting and Decrypting the Data using **BCTextEncoder**
- Creating and using **Self-Signed Certificates**
- Basic Disk Encryption using **VeraCrypt**
- Basic Data Encrypting using **Rohos Disk Encryption**
- Basic Data Encryption using **CrypTool**

Lab Analysis

Analyze and document the results related to this lab exercise. Provide your opinion on your target's security posture and exposure.


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





Calculating One-way Hashes using HashCalc

HashCalc enables you to compute multiple hashes, checksums and HMACs for files, text and hex strings. It supports MD2, MD4, MD5, SHA1, SHA2 (SHA256, SHA384, SHA512), RIPEMD160, PANAMA, TIGER, CRC32, ADLER32, and the hash used in eDonkey and eMule tools.

ICON KEY

 Valuable information

 Test your knowledge

 Web exercise

 Workbook review

Lab Scenario

Message digests or one-way hash functions distill the information contained within a file (small or large) into a single fixed-length number, typically between 128 and 256 bits in length. If any given bit of the function's input is changed, every output bit has a 50% chance of changing. Given an input file and its corresponding message digest, it should be nearly impossible to find another file with the same message digest value, as it is computationally unfeasible to have two files with the same message digest value.

Hash algorithms are widely used in a wide variety of cryptographic applications, and are useful for digital signature applications, file integrity checking, and storing passwords.

Lab Objectives


This lab will show you how to encrypt data and how to use it. It will teach you how to:

- Use encrypting/decrypting command
- Generate Hashes and checksum files

Lab Environment

To complete this lab, you will need:

- HashCalc located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\MD5 Hash Calculators\HashCalc**

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

Module 20 - Cryptography

- You can also download the latest version of HashCalc from the link <http://www.slavasoft.com/hashcalc/>
- If you decide to download the latest version, then screenshots shown in the lab might differ
- Follow the wizard driven installation instructions
- Run this tool in Windows Server 2016
- Administrative privileges to run tools

Lab Duration

Time: 5 Minutes

Overview of Hash

HashCalc is a fast and easy-to-use calculator that allows computing **message digests**, **checksums**, and **HMACs for files**, as well as for **text and hex strings**. It offers a choice of 13 of the most popular hash and checksum algorithms for calculations.

Lab Tasks

TASK 1

Calculate the Hash

1. Launch **HashCalc** application from **Apps** list.

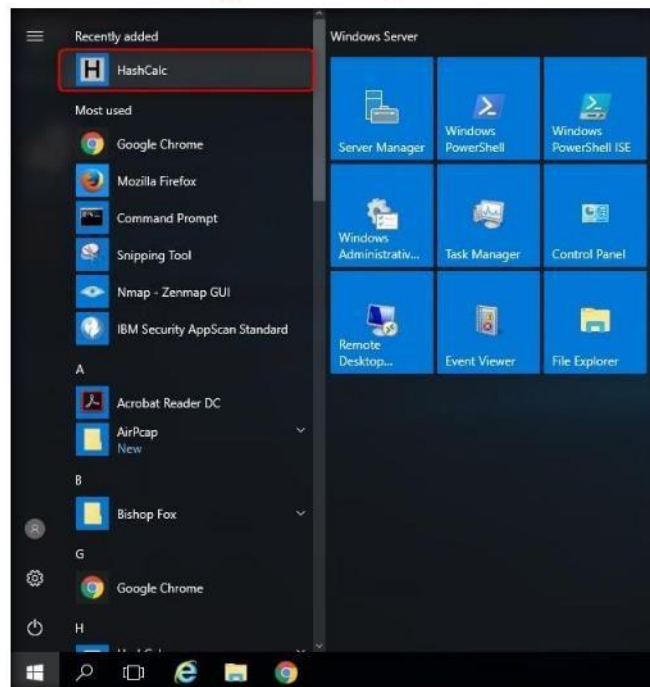


FIGURE 1.1: Launching HashCalc application

HashCalc simple dialog-size interface dispenses with glitz to plainly list input and results.

Hash algorithms support three input data formats: file, text string, and hexadecimal string.

2. The main window of **HashCalc** appears; select the type of **Data Format** (here, **Text string**) from dropdown list.

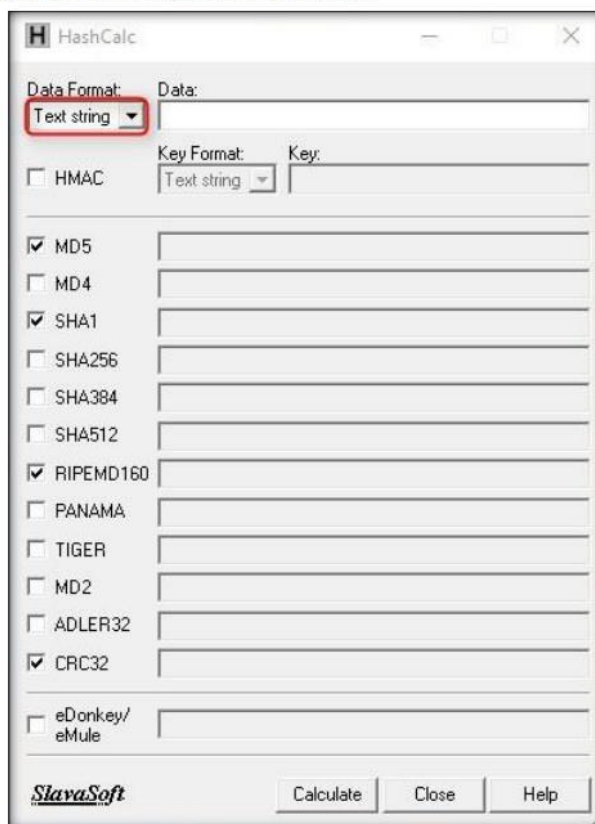


FIGURE 1.2: HashCalc main window

3. As you are specifying the data format as **Text string**, the application accepts text strings and converts them to their respective hashes.

Module 20 - Cryptography

4. Enter data which you would like to **calculate**.
5. Choose the appropriate **Hash algorithms** by selecting their respective checkboxes.
6. In this lab, **MD5**, **SHA1**, **RIPEND160** and **CRC32** hash algorithms have been selected.
7. Now, click **Calculate**.

 You can also download HashCalc from <http://www.slavasoft.com>.

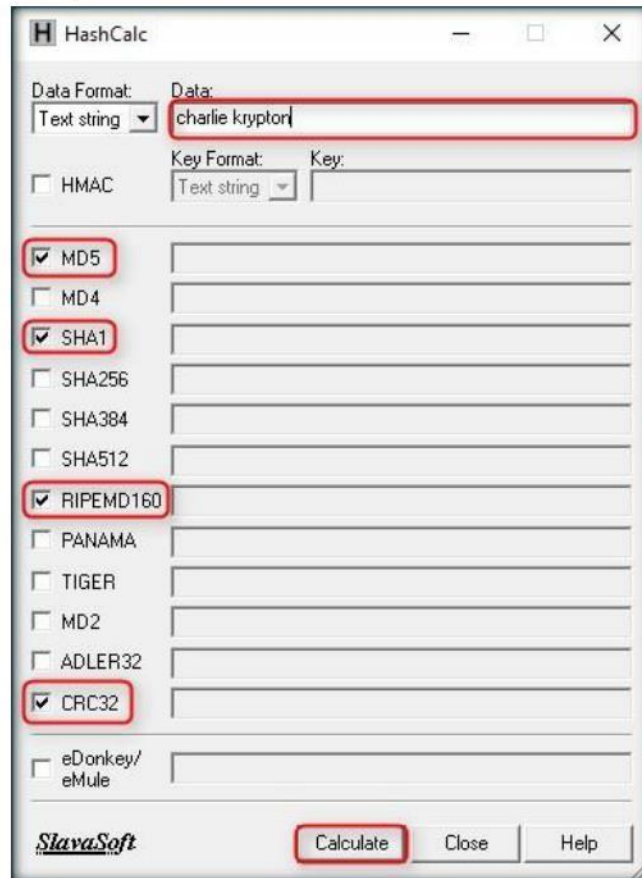



FIGURE 1.3: Calculating the hashes

Module 20 - Cryptography

8. The application calculates the hashes and displays them, as shown in the screenshot:

 HashCalc is used to generate crypting text.

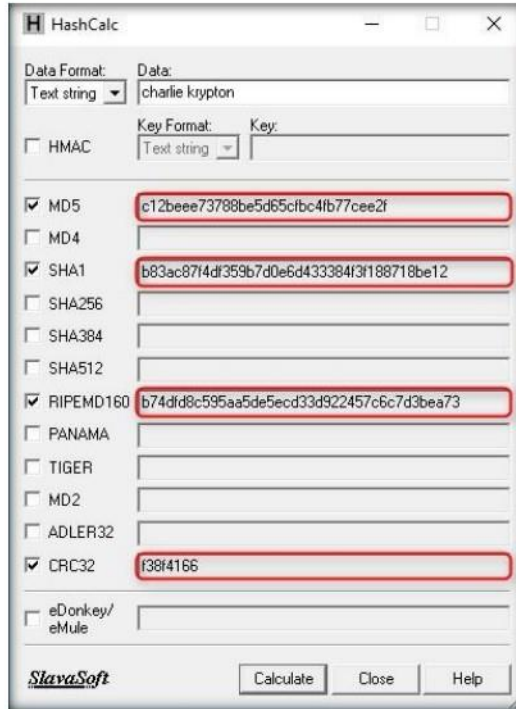


FIGURE 1.4: Hash is generated for chosen hash string

9. Hash calculation is mainly performed to check data integrity.

Lab Analysis

Document all Hash, MD5, and CRC values for further references.

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





Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs



Calculating MD5 Hashes using MD5 Calculator

MD5 Calculator is a simple application that calculates the MD5 hash of a given file. It can be used with big files (e.g., multiple gigabytes). It features a progress counter and a text field from which the final MD5 hash can be easily copied to the clipboard.

ICON KEY

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

Lab Scenario

There has been a need to protect information from “prying eyes.” In the electronic age, information that could otherwise benefit or educate a group or individual can also be used against such groups or individuals. Industrial espionage among highly competitive businesses often requires extensive security measures to be put into place. And those who wish to exercise their personal freedom, outside oppressive governments, may also wish to encrypt certain information to avoid suffering the penalties of going against the wishes of those who attempt to control it. Still, the methods of data encryption and decryption are relatively straightforward; algorithms are used to encrypt the data and store system information files safely, away from prying eyes. To be an Expert Ethical Hacker and Penetration Tester, you must understand data encryption using encrypting algorithms.

Lab Objectives

This lab will give you experience on encrypting data and show you how to do it. It will teach you how to:

- Use encrypting/decrypting command
- Calculate the MD5 value of the selected file

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

Lab Environment

To complete this lab, you will need:

- MD5 Calculator located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\MD5 Hash Calculators\MD5 Calculator**
- You can also download the latest version of MD5 Calculator from the link **<http://www.bulzip.com/products/md5/info.php>**
- If you decide to download the latest version, then screenshots shown in the lab might differ
- Run this tool in Windows Server 2016
- Administrative privileges to run tools

Lab Duration

Time: 5 Minutes

Overview of MD5 Calculator

MD5 Calculator is a bare-bones program for calculating and comparing MD5 files. While its layout leaves something to be desired, its results are fast and simple.

Lab Tasks

TASK 1

Calculate MD5 Checksum

1. Navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\MD5 Hash Calculators\MD5 Calculator**, double-click **md5calc(1.0.0.0).msi** and follow the installation steps to install MD5 Calculator.
2. To find MD5 Hash of any file, right-click on the specific file (here, **md5calc(1.0.0.0).msi**), and Select **"MD5 Calculator"** from the context menu.

MD5 checksum is used to generate MD5 hash.

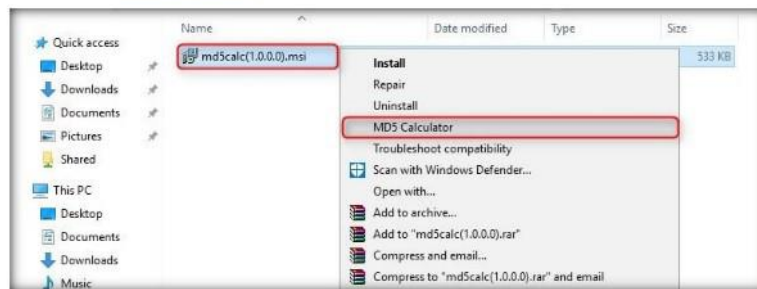



FIGURE 2.1: MD5 Calculator option in context menu

Module 20 - Cryptography

3. **MD5 Calculator** shows the MD5 digest of the selected file.

Note: Alternatively, you can browse any file to calculate the MD5 hash and click on the **Calculate** button to calculate the MD5 hash of the file.

 MD5 hash (or checksum) functions as a compact digital fingerprint of a file.

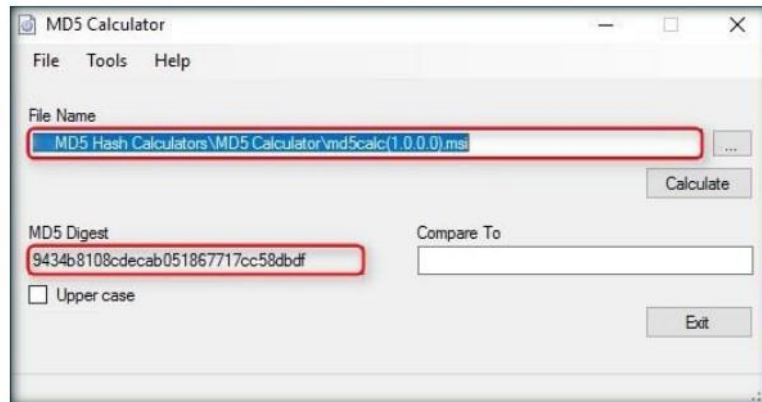


FIGURE 2.2: MD5 is generate for the chosen file

4. MD5 calculator is used to check the integrity of a file.
5. If a person wants to send a file to another person via a medium, he/she will calculate its hashes and send the file (along with the hash value) to the intended person. When the person on the other side receives the mail, he/she will download the file and calculate its value using MD5 Calculator.
6. Then, the person compares the generated hash value with the hash value that was sent through mail. If both the hash values tally, it is evident that the person obtained the file without any modifications by a third person.

Lab Analysis

Analyze and document the results related to this lab exercise.

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





Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs




Understanding File and Text Encryption using CryptoForge

CryptoForge is a file encryption software for personal and professional data security. It allows you to protect the privacy of sensitive files, folders, or email messages, by encrypting them with strong encryption algorithms.

ICON KEY

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

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

Lab Scenario

CryptoForge allows you to protect the privacy of sensitive files, folders, or email messages, by encrypting them with up to four strong encryption algorithms. Once the information has been encrypted, it can be stored on insecure media or transmitted on an insecure network—such as the Internet—and remain a secret. Later, the information can be decrypted into its original form.

Lab Objectives

This lab will show you how to encrypt files and text.

Lab Environment

To complete this lab, you will need:

- CryptoForge located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\CryptoForge**
- You can also download the latest version of CryptoForge from the link **<http://www.cryptoforge.com/download>**
- If you decide to download the latest version, then screenshots shown in the lab might differ
- Follow the wizard driven installation instructions
- Windows Server 2016 running as a virtual machine
- Windows 10 running as a virtual machine
- Administrative privileges to run tools

Lab Duration

Time: 10 Minutes

Overview of the Lab

The lab demonstrates basic encryption methodology used to encrypt files and text messages and share them with the intended person/people.

Lab Tasks

TASK 1

Encrypt a File

1. In the Windows Server 2016 machine, navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\CryptoForge**, double-click **CryptoForge.exe** and follow the steps to install the application.
2. Once done with the installation, log in to Windows 10 virtual machine, navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\CryptoForge**, double-click **CryptoForge.exe** and follow the steps to install the application.

Note: If a **User Account Control** pop-up appears, click **Yes**. If a **Windows Security dialog**-box appears, enter the credentials of Windows Server 2016 machine, and click **OK**.

3. Now, switch to **Windows 10** machine, navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\CryptoForge**, right-click **Confidential.txt**, and select **Encrypt** from the context menu.

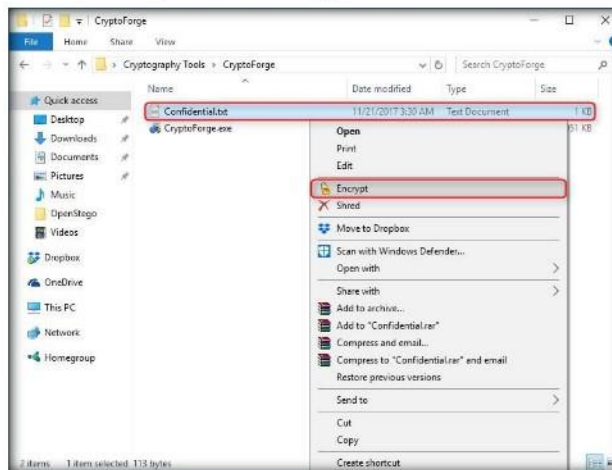


FIGURE 3.1: Encrypting a File

Module 20 - Cryptography

- The **Enter Passphrase - CryptoForge Files** dialog-box appears; type a password in the **Passphrase** field, retype it in the **Confirm** field, and click **OK**. The password used in this lab is **qwerty@123**.

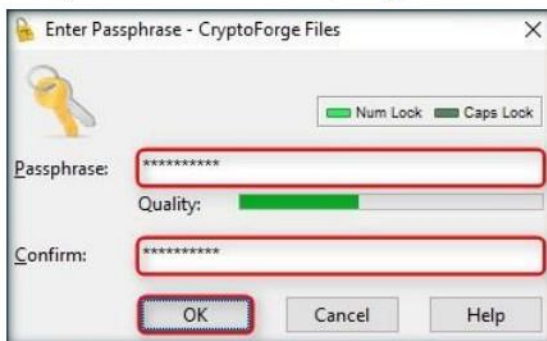


FIGURE 3.2: Enter Passphrase - CryptoForge Files Dialog-Box

- Now, the file will be encrypted in the same location, and the old file will be deleted automatically, as shown in the screenshot:

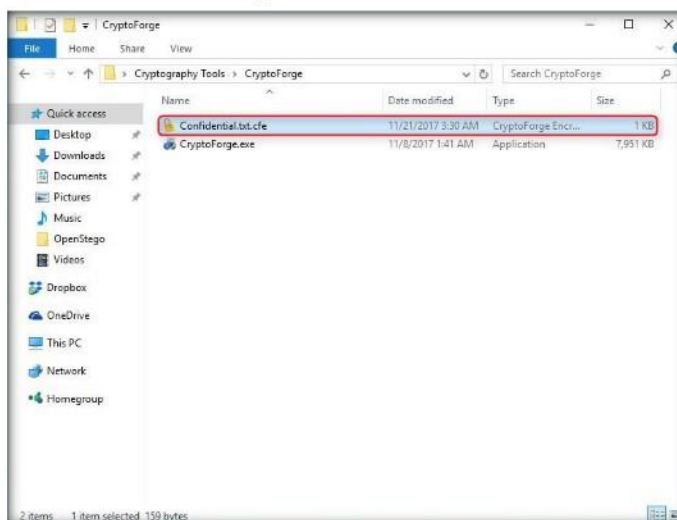


FIGURE 3.3: File Encrypted

TASK 2

Decrypt the Encrypted File

- No one can access this file unless he/she provides the password for the encrypted file. You will have to share the password with him/her through message, mail, or any other means.
- Let us assume that you shared this file through shared network drive.

Module 20 - Cryptography

- Now, switch to Windows Server 2016 virtual machine, navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\CryptoForge**. You will observe the encrypted file in this location.

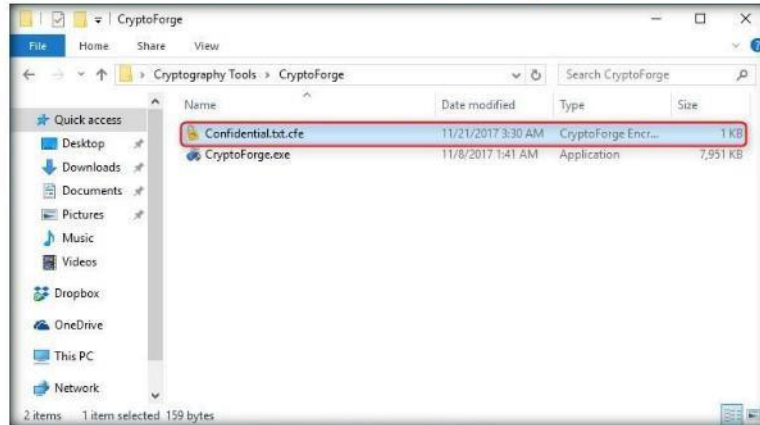


FIGURE 3.4: Viewing the Encrypted File

- Now, double-click the encrypted file to decrypt it and view its contents.

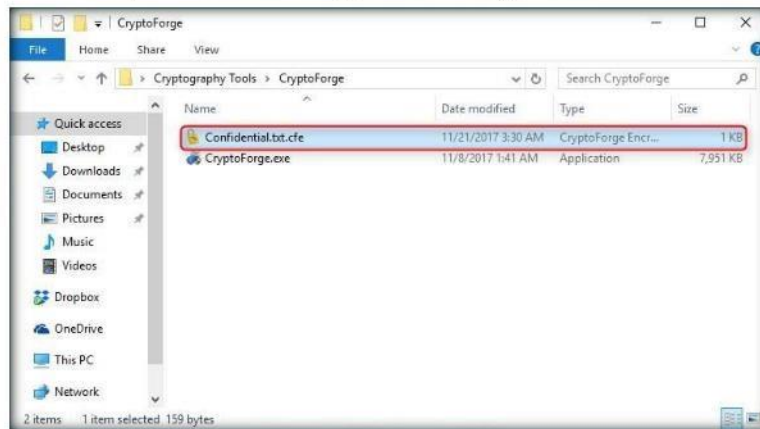


FIGURE 3.5: Decrypted the Encrypted File

Module 20 - Cryptography

10. The **Enter Passphrase - CryptoForge Files** dialog-box appears; enter the password that you have provided to encrypt the file, and click **OK**.



FIGURE 3.6: Enter Passphrase - CryptoForge Files Dialog-Box

11. On entering the password, the file will be successfully decrypted. You may now double-click the file to view its contents.

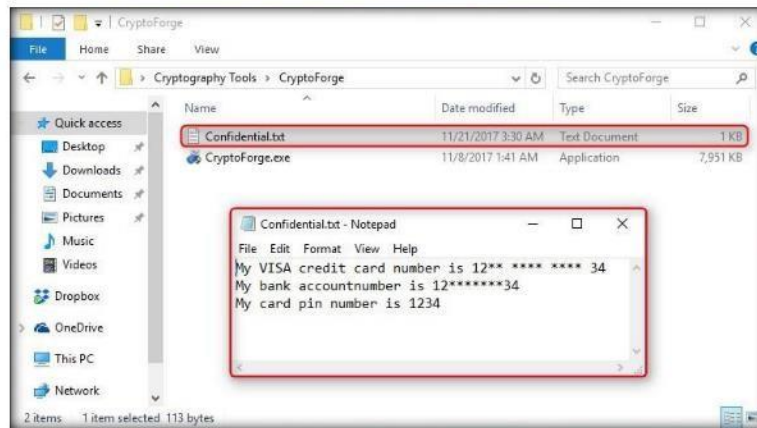


FIGURE 3.7: File Decrypted Successfully

12. So far, you have seen how to encrypt a file and share it with the intended user. Now, you will learn how to share an encrypted message with a user.
13. Switch to **Windows Server 2016** machine, go to the **Apps** screen, and click **CryptoForge Text** to launch the application.

TASK 3

Encrypt a Message

Module 20 - Cryptography

14. **CryptoForge Text** window appears, type a message, and click **Encrypt** from the toolbar.

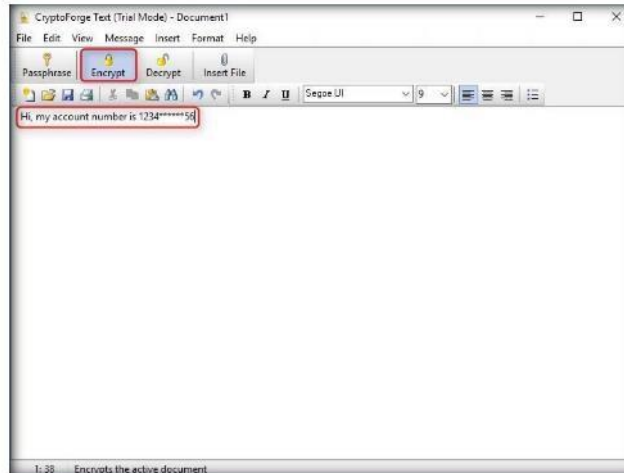


FIGURE 3.8: Encrypting a Text Message

15. The **Enter Passphrase - CryptoForge Text** dialog-box appears; type a password in the **Passphrase** field, retype it in the **Confirm** field, and click **OK**. The password used in this lab is **test@123**.

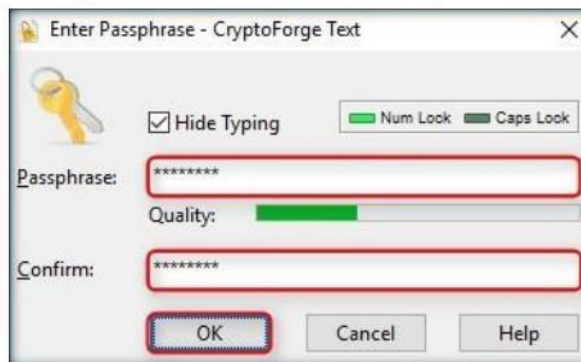


FIGURE 3.9: Enter Passphrase - CryptoForge Text Dialog-Box

Module 20 - Cryptography

16. The message you type will be encrypted, as shown in the screenshot:

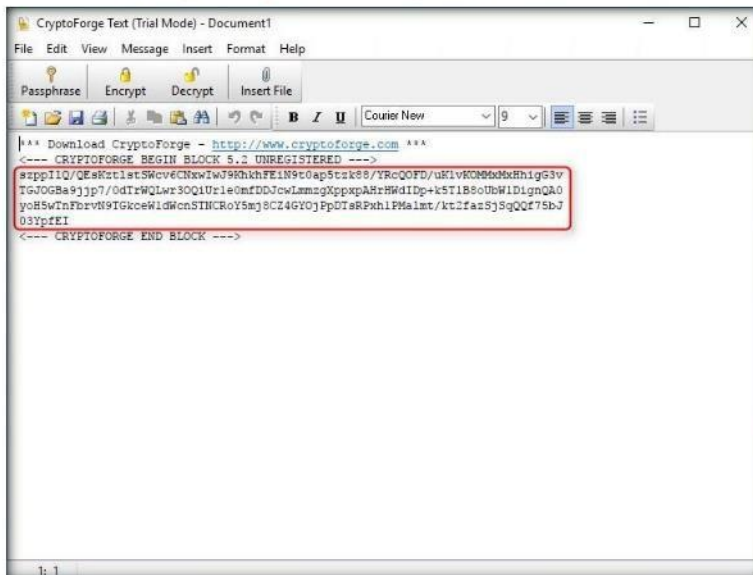


FIGURE 3.10: Message Encrypted

17. Now, you need to save the file. Click **File** in the menu bar, and click **Save**.

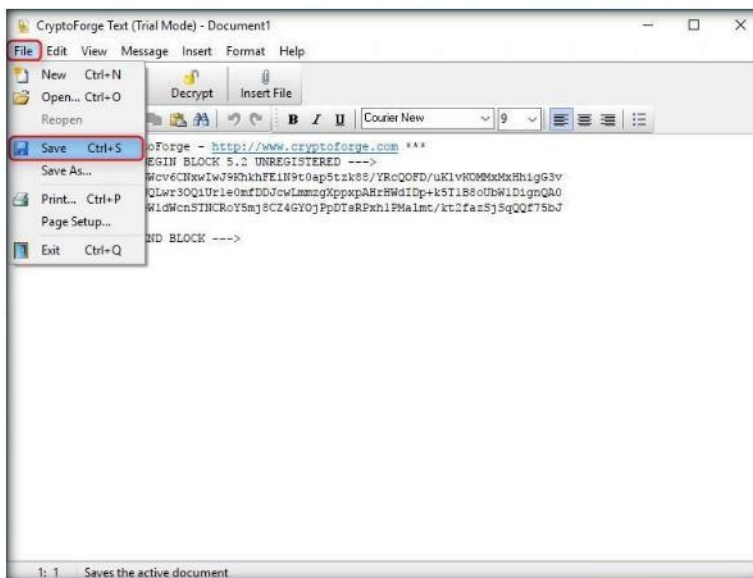


FIGURE 3.11: Saving the File

Module 20 - Cryptography

18. The **Save As** window appears; navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography Tools\CryptoForge**, specify the file name as **Credentials.cfd** and click **Save**.

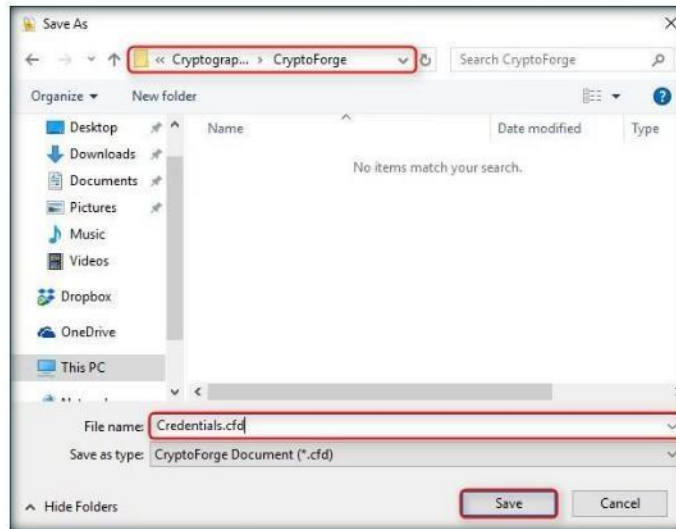


FIGURE 3.12: Saving the File

19. Close the **CryptoForge Text** window.
20. Now, let us assume that you shared the file through mapped network drive, and shared the password to decrypt the file in an email message or some other means.
21. Switch to Windows 10 virtual machine, and navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography Tools\CryptoForge**. Observe the encrypted file in this location; double-click.

TASK 4

Decrypt the Encrypted Message

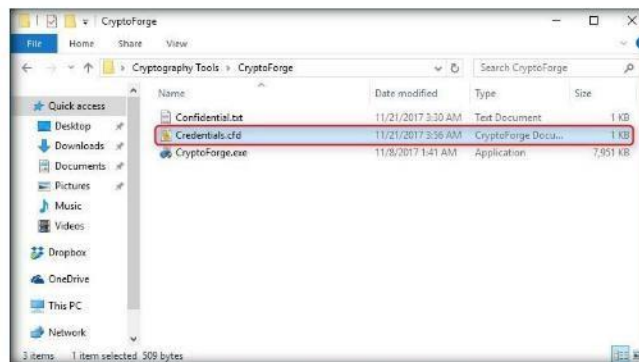


FIGURE 3.13: Viewing the Encrypted File

Module 20 - Cryptography

22. The **CryptoForge Text** window appears, displaying the message in encrypted format. Click **Decrypt** to decrypt it.

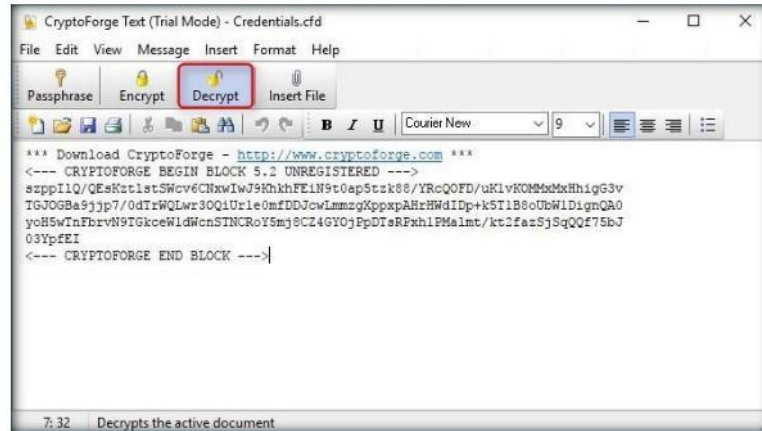


FIGURE 3.14: Decrypting the Encrypted File

23. The **Enter Passphrase - CryptoForge Text** dialog-box appears; enter the password you used to encrypt the message in the **Passphrase** field, and click **OK**.



FIGURE 3.15: Enter Passphrase - CryptoForge Text Dialog-Box

Module 20 - Cryptography

24. The **CryptoForge Text** window appears, displaying the message in plain-text format, as shown in the screenshot:

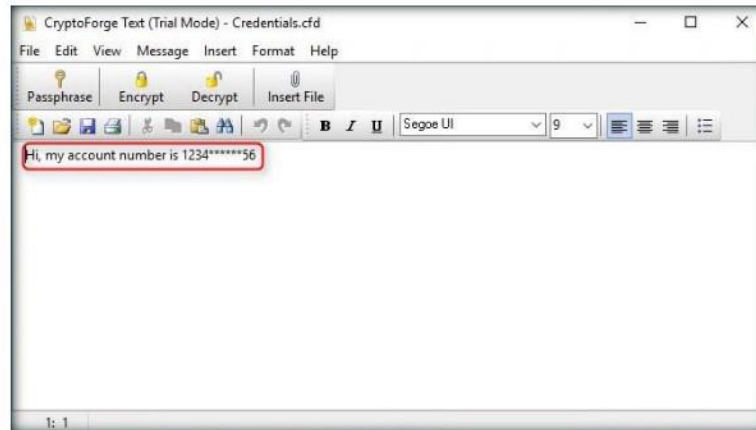


FIGURE 3.16: Message Decrypted Successfully

25. Thus, you have used CryptoForge tool to encrypt as well as share files and messages with the intended person.
26. In real time, you may share sensitive information through email by encrypting data using CryptoForge.

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	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs



Basic Data Encryption using Advanced Encryption Package

Advanced Encryption Package is most noteworthy for its flexibility; not only can you encrypt files for your own protection, but you can easily create "self-decrypting" versions of your files that others can run without needing this or any other software.

ICON KEY

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


Lab Scenario

Data encryption and decryption operations require major security applications to secure data. Most systems use block ciphers, such as public AES standard. However, implementations of block ciphers such as AES, as well as other cryptographic algorithms, are subject to side-channel attacks. These attacks allow adversaries to extract secret keys from devices by passively monitoring the power consumption of other side channels. Counter measures are required for applications to which side-channel attacks are a threat. These include several military and aerospace applications in which program information, classified data, algorithms, and secret keys reside on assets that may not always be physically protected. To be an Expert Ethical Hacker and Penetration Tester, you must understand file data encryption.

Lab Objectives

This lab will give you experience regarding data encryption and show you the techniques to do it. It will teach you how to:

- Use encrypting/decrypting command
- Calculate the encrypted value of the selected file

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

Lab Environment

To complete this lab, you will need:

- Advanced Encryption Package located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\Advanced Encryption Package 2017**
- You can also download the latest version of Advanced Encryption Package from the link http://www.secureaction.com/encryption_pro/
- If you decide to download the latest version, then screenshots shown in the lab might differ
- Administrative privileges to run tools
- Run this tool in Windows Server 2016

Lab Duration

Time: 10 Minutes

Overview of Advanced Encryption Package

Advanced Encryption Package includes a file shredder that wipes out the contents of your original files. It also integrates nicely with Windows Explorer, allowing you to use Explorer's context menus and avoid having another window clutter your screen.

Lab Tasks




TASK 1

Encrypting a File

1. Navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\Advanced Encryption Package 2017**, double-click **aep.msi** and follow the steps to install the application.

Module 20 - Cryptography

2. On completing the installation, launch **Advanced Encryption Package** application from the **Apps** screen.

 Advanced Encryption Package is a symmetric-key encryption comprising three block ciphers, AES-128, AES-192 and AES-256.

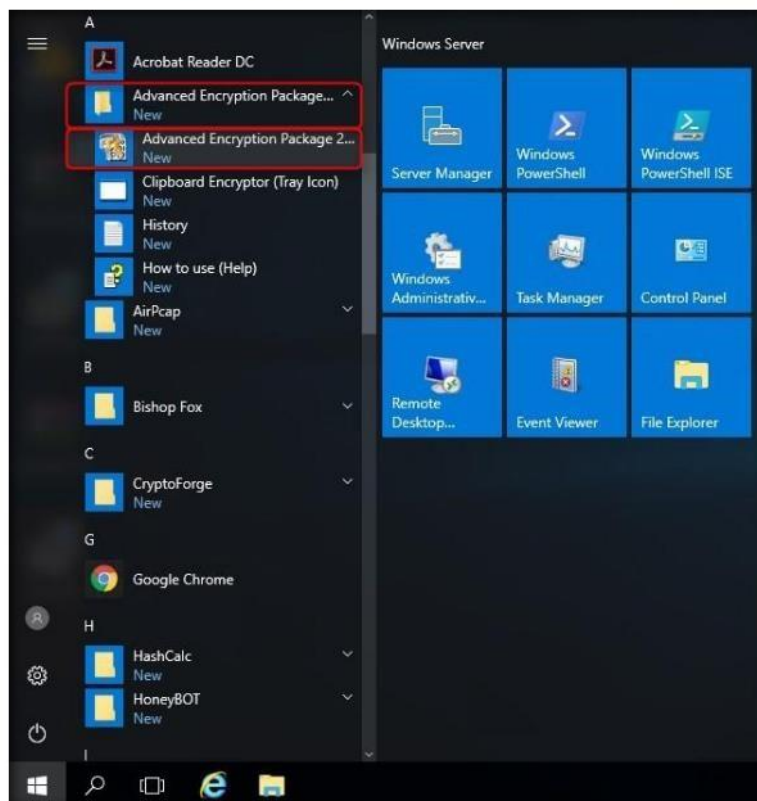


FIGURE 4.1: Launching Advanced Encryption Package application from the Apps screen.

Module 20 - Cryptography

3. The **Advanced Encryption Package 2017 - License Manager** window appears displaying the **License Manager** section. Select **Start free 30-day trial** radio button, and click **Next**.

You can also download Advance Encryption Package from <http://www.secureaction.com>

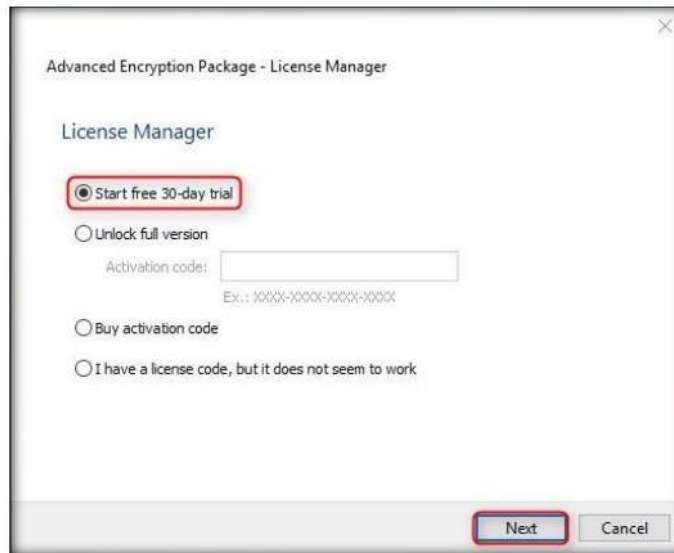


FIGURE 4.2: License Manager window

4. The **Activating** step appears; click **Next**.

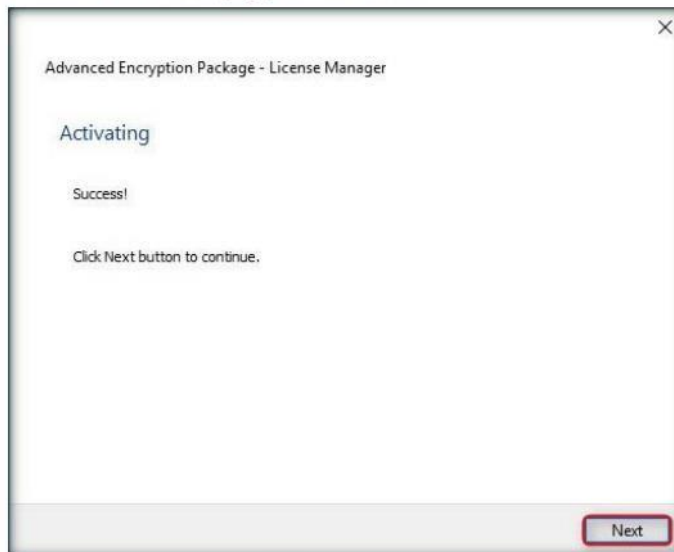
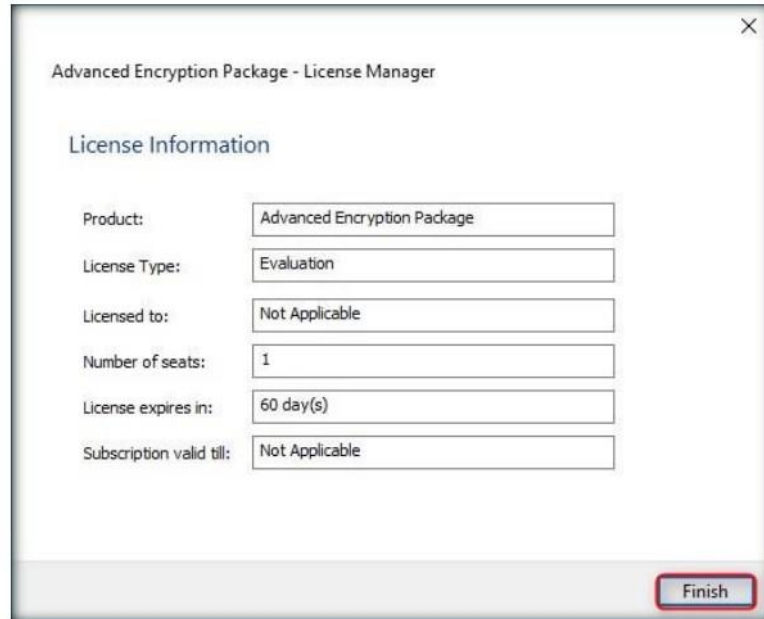


FIGURE 4.3: Activation Window

Module 20 - Cryptography

5. Leave all the options set to default in **License Information** step, and click **Finish**.



Advanced Encryption Package - License Manager

License Information


Product:	Advanced Encryption Package
License Type:	Evaluation
Licensed to:	Not Applicable
Number of seats:	1
License expires in:	60 day(s)
Subscription valid till:	Not Applicable


Finish

FIGURE 4.4: License Information section

Module 20 - Cryptography

- The main window of **Advanced Encryption Package** appears.
- A sample file named **Sample.docx** is provided at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\Advanced Encryption Package 2017**. Select the sample file, and click **Encrypt** in the toolbar.

 Advance Encryption Package is easy to use for novices.

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

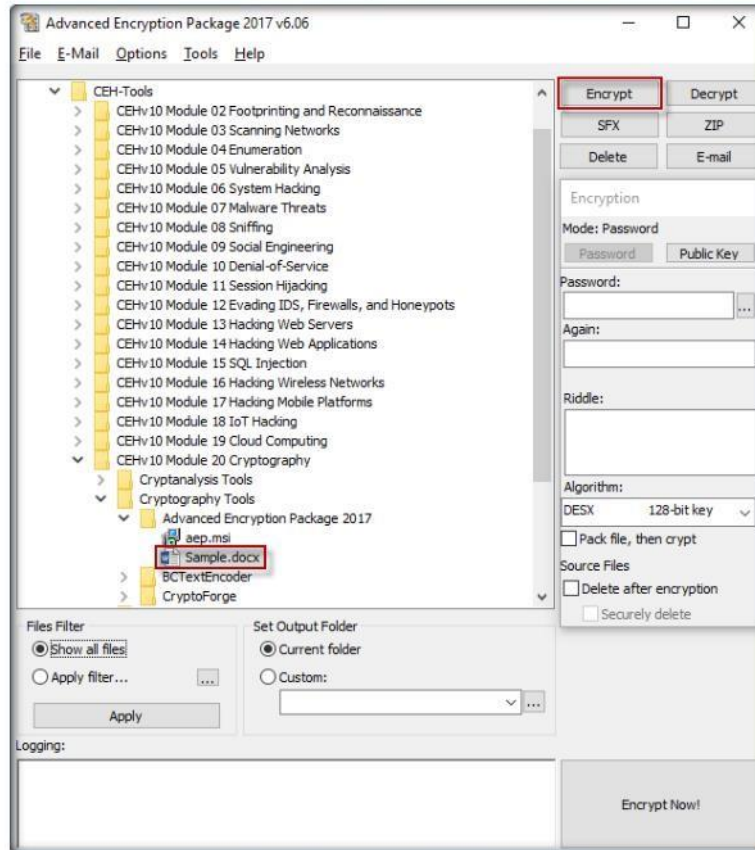



FIGURE 4.5: Main window of Advance Encryption Package

Module 20 - Cryptography

- You need to provide a password for encryption. Enter the password in **Pwd** field, retype it in the **Again** field, and click **Encrypt Now!**.
- In this lab, the password is **test@123**.

 It creates encrypted self-extracting files to send as email attachments.

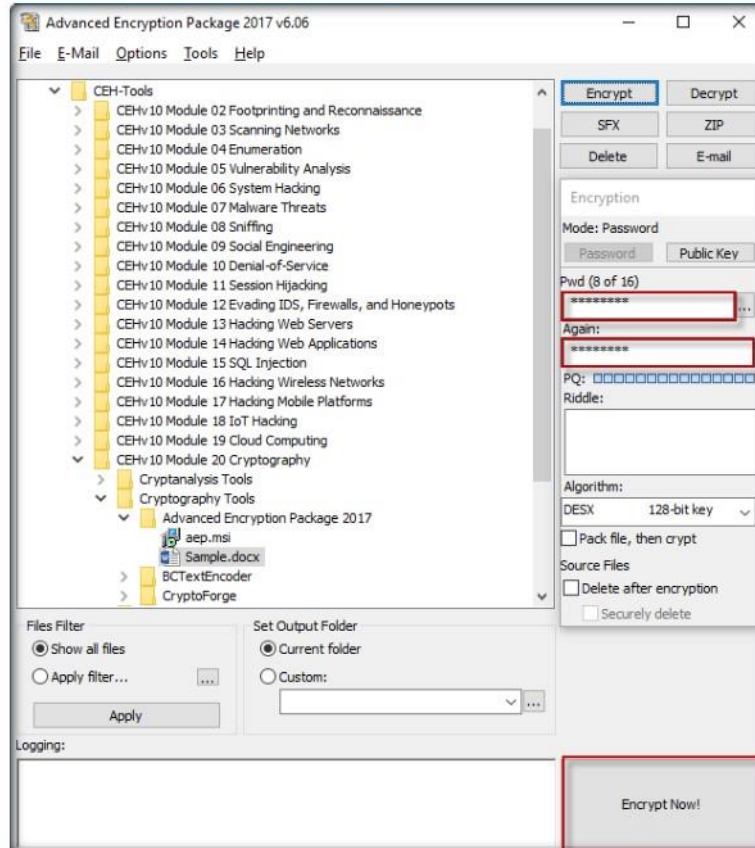


FIGURE 4.6: Encrypting the selected file

Module 20 - Cryptography

10. The encrypted Sample File appears in the same location as the original file (i.e., **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\Advanced Encryption Package 2017**).

11. To **decrypt** the file, first select the encrypted file, and click on **Decrypt**.

Note: Navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\Advanced Encryption Package 2017** and delete the unencrypted source file, as conflicts might occur while decrypting the encrypted file in the same location.

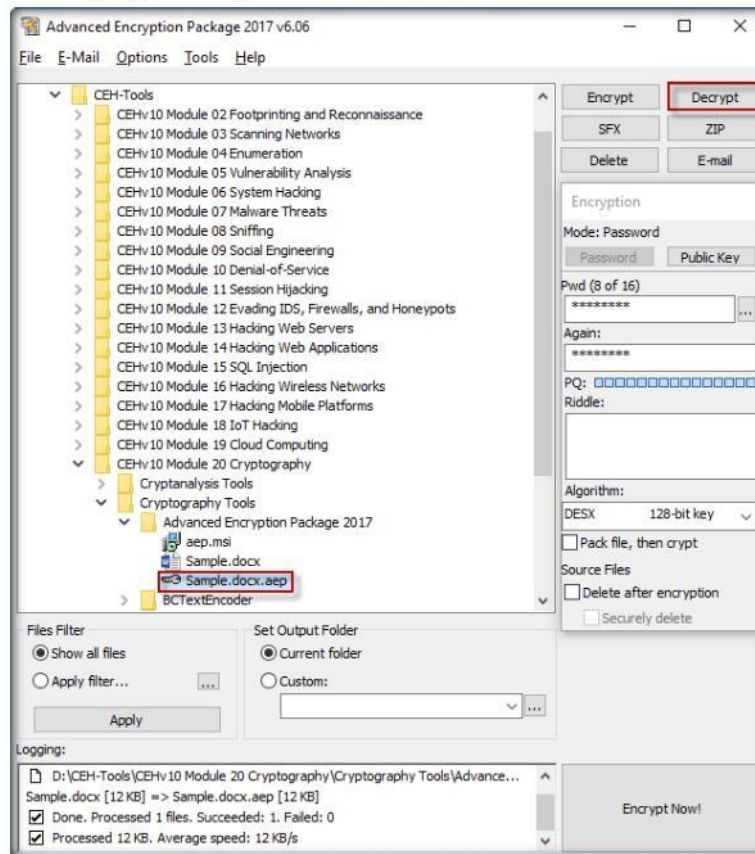


FIGURE 4.7: Decrypting the selected file

Module 20 - Cryptography

12. You will be prompted to enter the password.
13. Because the unencrypted source file is already present in the same location, click **Leave it alone**, under **Source file(s)**, and click **Decrypt Now!**

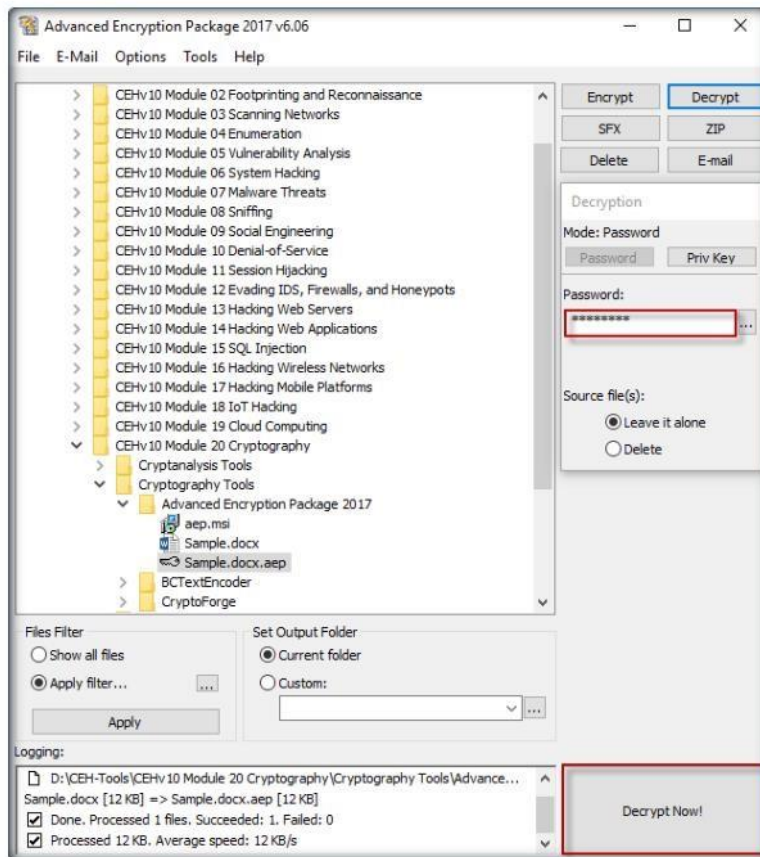


FIGURE 4.8: Decrypting the selected file

Module 20 - Cryptography

14. The decrypted file appears in the same location as shown in the screenshot:

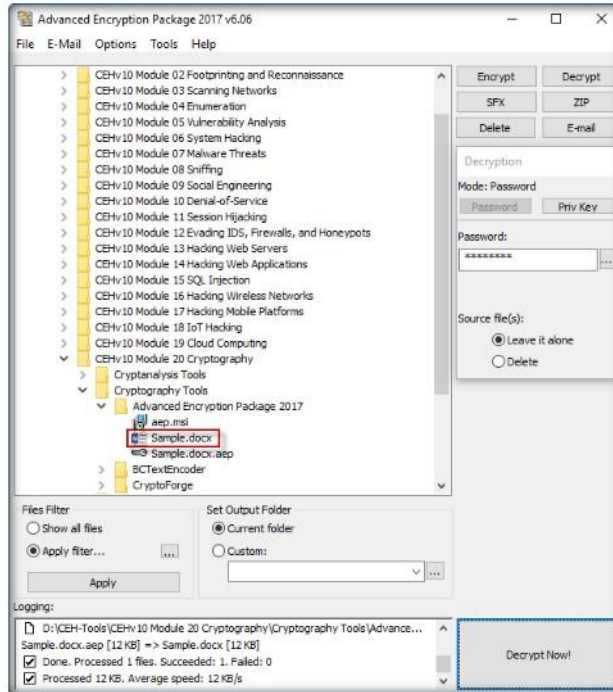


FIGURE 4.9: Decrypted file

15. In real time, network administrators or ethical hackers use this tool to encrypt files and send it to the intended persons to safeguard the integrity of the files.

Lab Analysis

Analyze and document the results related to this lab exercise.

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

Internet Connection Required	
<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input type="checkbox"/> iLabs



Encrypting and Decrypting the Data using BCTextEncoder

BCTextEncoder simplifies encoding and decoding text data. Plain text data are compressed, encrypted and converted to text format, which can then be easily copied to the clipboard or saved as a text file.

ICON KEY

Valuable information

Test your knowledge

Web exercise

Workbook review

Lab Scenario

To be an expert ethical hacker and penetration tester, you must have knowledge of cryptography functions.

Lab Objectives

This lab will provide you experience on encrypting data and show you how to do it. It will teach you how to:

- Use Encode/decode text data encrypted with a password

Lab Environment

To complete this lab, you will need:

- BCTextEncoder located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\BCTextEncoder**
- If you decide to download the latest version, then screenshots shown in the lab might differ
- Run this tool on Windows Server 2016 machine
- Administrative Privileges to run the tool

Lab Duration

Time: 10 Minutes

Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography

Overview of BCTextEncoder

BCTextEncoder uses public key encryption methods, as well as password-based encryption. This utility software uses strong and approved symmetric and public key algorithms for data encryption.

Lab Tasks

TASK 1 Encrypt the Data

1. Navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptography Tools\BCTextEncoder** and double-click **BCTextEncoder.exe**.
2. The main window of BCTextEncoder appears as shown in the following screenshot:

BCTextEncoder utilizes the following encryption algorithms:

- ZLIB compression algorithm
- AES (Rijndael) encryption algorithm for password based encryption
- RSA asymmetric encryption algorithm for public key encryption

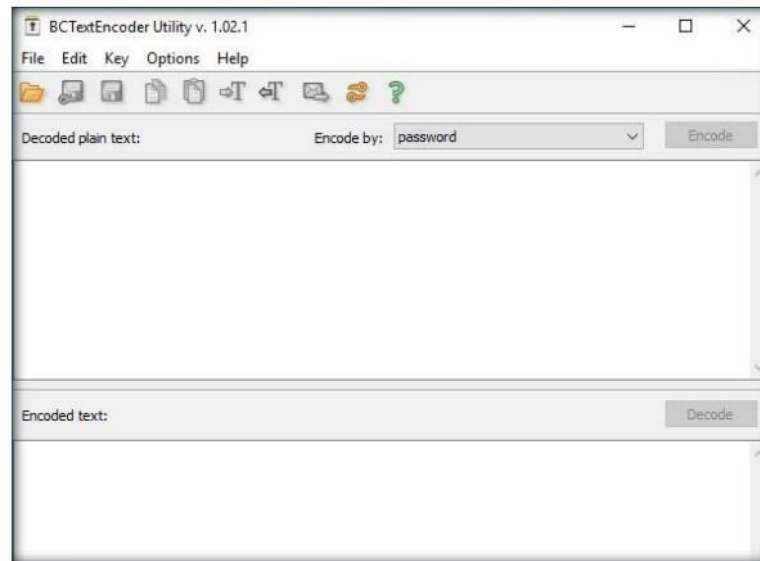


FIGURE 5.1: Main window of BCTextEncoder

Module 20 - Cryptography

3. To encrypt the text, type the text in the **clipboard**. Or, select the secret data, and paste it to the clipboard by pressing **Ctrl+V** and click **Encode**.

BCTextEncoder is intended for fast encoding and decoding text data

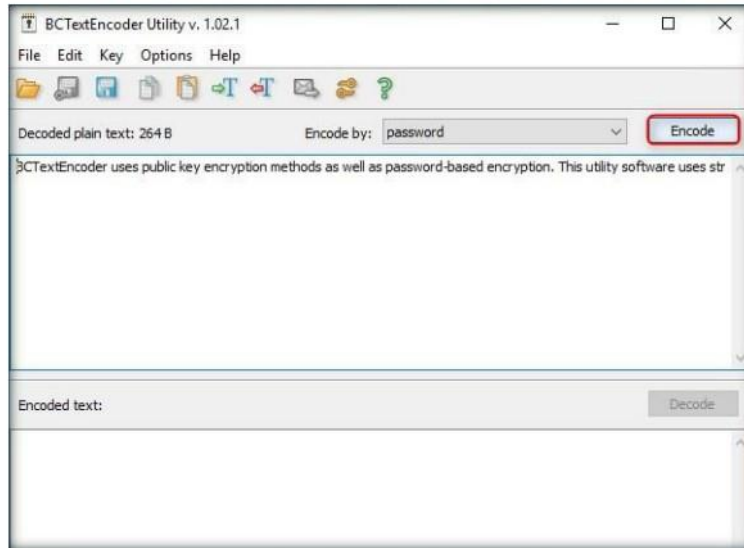



FIGURE 5.2: Secret information in clipboard

4. The **Enter password** dialog-box appears; set the **Password** (**qwerty@123**), and **Confirm** it in the respective field.
5. Click **OK**.

 The main advantage of BCTextEncoder is that it supports public-key encryption.

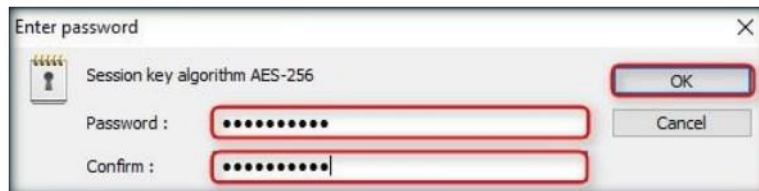


FIGURE 5.3: Set the password for encryption

Module 20 - Cryptography

6. BCTextEncoder encodes the text and displays it in the **Encoded text** section, as shown in the screenshot:

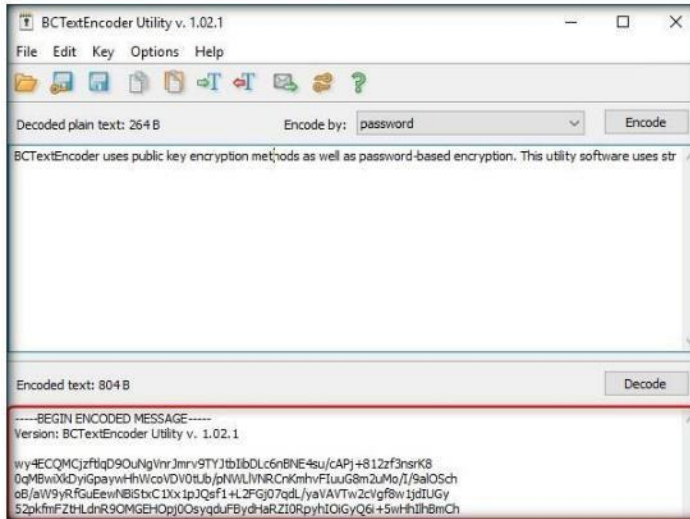


FIGURE 5.4: Encoded text

TASK 2

Decrypt the Data

7. To **decrypt** the data, first you need to clean the **Decoded plain text** in the clipboard.
8. Click **Decode**.

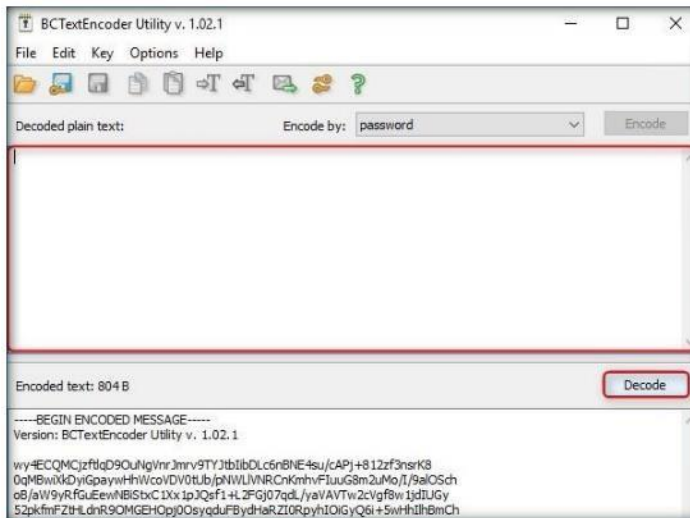


FIGURE 5.5: Decoding the data

Module 20 - Cryptography

BCArchive includes the BC Key Manager utility to manage your own public/secret key pair as well as public keys you have received from other people. BCTextEncoder not only encrypts, but also compresses the data.

9. Enter password for encoding text dialog-box appears; enter the Password (qwerty@123) in password field, and click **OK**.



FIGURE 5.6: Enter the password for decoding

10. Decoded plain text appears, as shown in the screenshot:

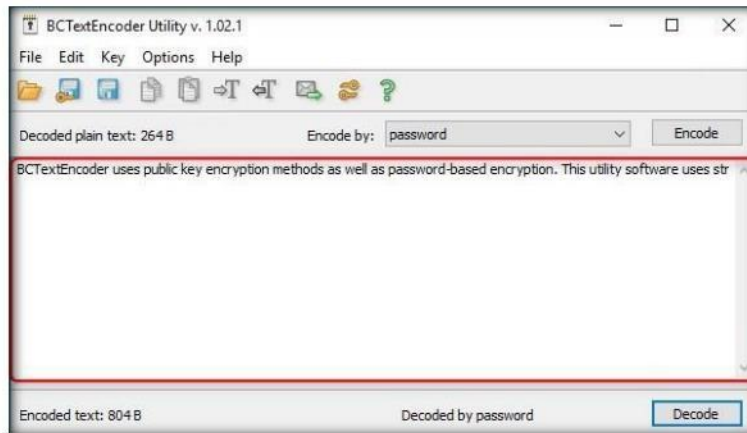


FIGURE 5.7: Output decoded text

11. This way, you need to encode the text while sending it to the intended user along with the password used for encryption. The user for whom the text is intended should have the BCTextEncoder application installed on his/her machine.
12. He/she will have to paste the encoded text in the Encoded text section and use the password you shared, to decode it to plain text.

Lab Analysis

Analyze and document the results related to this lab exercise.

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


Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs





Creating and using Self-Signed Certificates

SSL is an essential part of securing your IIS 7.0 site. Creating a self-signed certificate in IIS 7 is much easier to do than in the previous versions. SSL certificates enable the encryption of all traffic sent to and from your IIS website, preventing others from viewing sensitive information. It uses public-key cryptography to establish a secure connection. This means that anything encrypted with a public key (the SSL certificate) can only be decrypted with a private key and vice-versa.

ICON KEY

 Valuable information

 Test your knowledge

 Web exercise

 Workbook review

Lab Scenario

A self-signed certificate is an identity certificate signed by the same entity whose identity it certifies. In general, self-signed certificates are widely used for testing servers.

Lab Objectives

This lab will give you experience on how to create self-signed certificates.

Lab Environment

To complete this lab, you will need:


- Windows Server 2016
- Administrative privileges required to perform this lab

Lab Duration

Time: 10 Minutes

Overview of Lab

In cryptography and computer security, a self-signed certificate is an identity certificate signed by the same entity whose identity it certifies. However, the term has nothing to do with the identity of the person or organization that actually performs the signing procedure.

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

Lab Tasks

TASK 1

Verifying Self-Signed Certificate

1. Before you start the lab, you will need to check with your local sites whether they include a self-signed certificate.
2. Launch a web browser, type **https://www.goodshopping.com** in the address bar, and press **Enter**. In this lab, we are using Google Chrome.



FIGURE 6.1: www.goodshopping.com before adding Certificate

3. As you are using an https channel to browse, it displays a page stating that **This site can't be reached**.
4. As the site does not have a self-signed certificate, it displays a **Not Found** page, as shown in the screenshot. Close the web browser.

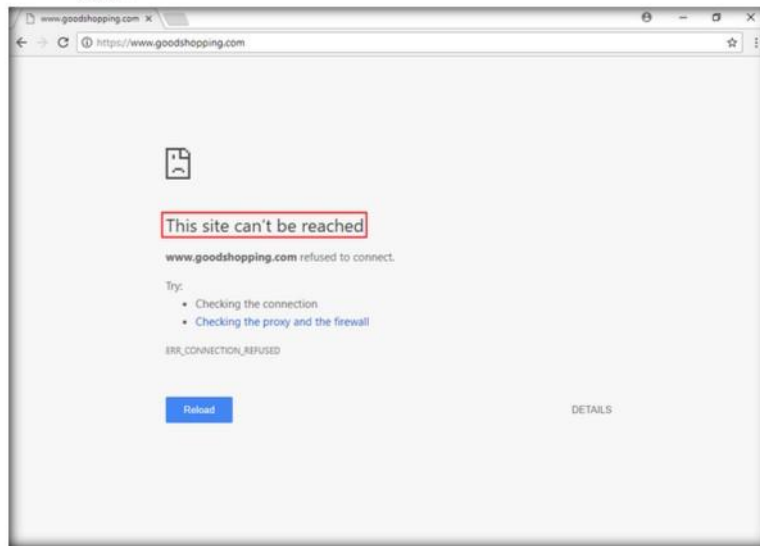


FIGURE 6.2: Connection is not Private

In technical terms a self-signed certificate is one signed with its own private key.

TASK 2

Launch IIS Manager

5. Click **Start** menu present on the lower-left corner of the **Desktop**.
6. Scroll down for **Windows Administrative Tools** folder.

Module 20 - Cryptography

- Click **Internet Information Services (IIS) Manager** application to launch IIS Manager, as shown in the screenshot:

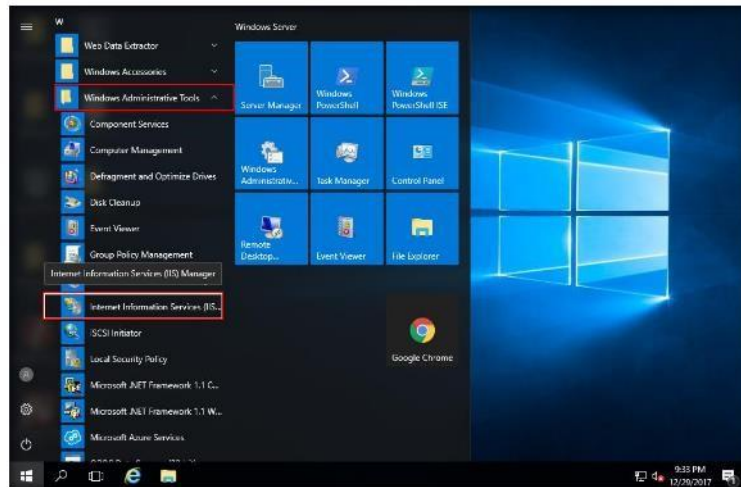


FIGURE 6.3: Windows Start menu Apps

- If the **Do you want to get started with Microsoft Web Platform ...** pop-up appears, click **Cancel**.
- The Internet Information Services (IIS) Manager window appears; click the **Machine** name in the Connections pane, and double-click **Server Certificates**.

TASK 3 Configure Server Certificates

In typical Public Key Infrastructure (PKI) arrangements, a digital signature from a Certificate Authority (CA) attests that a particular public key certificate is valid (i.e., contains correct information). Users or their software on their behalf, check that the private key used to sign some certificate matches the public key in the CA's certificate.

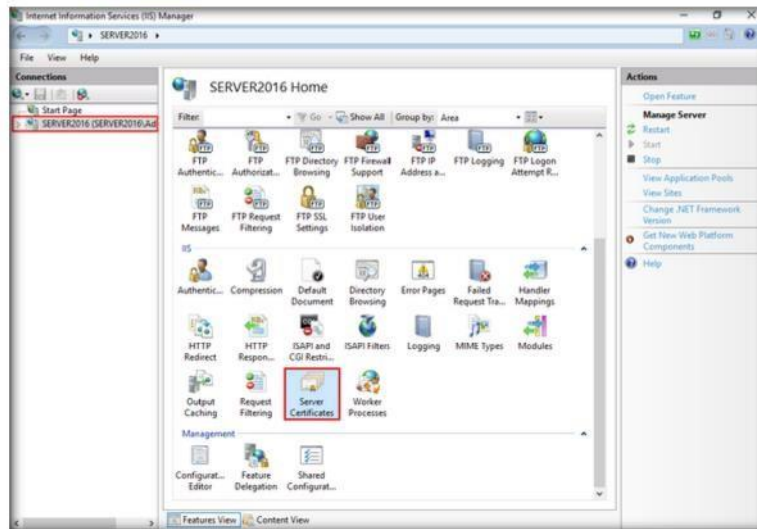


FIGURE 6.4: IIS Manager Server Certificates

TASK 4

Create Self-Signed Certificate

- In the **Server Certificates** wizard, click **Create Self-Signed Certificate** in the **Actions** pane.

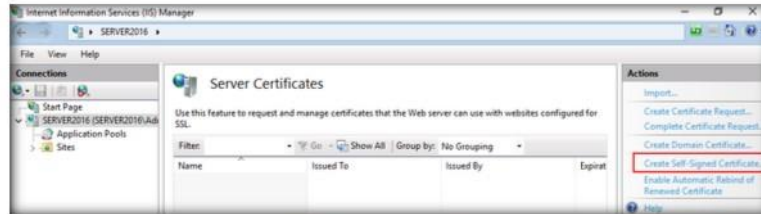


FIGURE 6.5: Server Certificates

- The **Create Self-Signed Certificate** wizard appears; type a name in the **Specify a friendly name for the certificate** field.
- Choose **Personal** in the **Select a certificate store for the new certificate** field drop-down list, and click **OK**.

Since CA certificates are often signed by other, "higher-ranking," CAs, there must necessarily be a highest CA, which provides the ultimate in attestation authority in that particular PKI scheme.

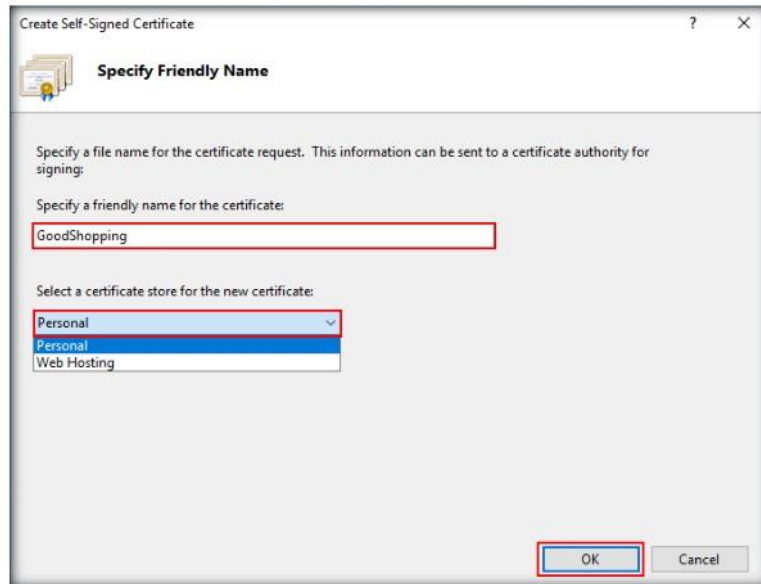


FIGURE 6.6: Specify Friendly Name

Obviously, the highest-ranking CA's certificate can't be attested by some other higher CA (there being none), and so that certificate can only be "self-signed." Such certificates are also termed root certificates.

Module 20 - Cryptography

13. The New Self-Signed Certificate will be displayed in the Server Certificates pane, as shown in screenshot:

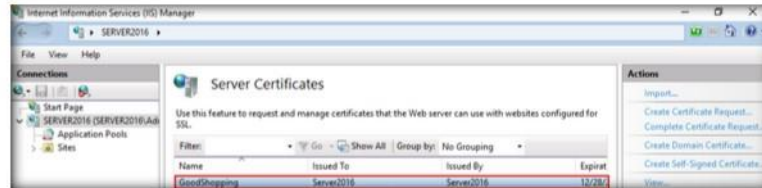


FIGURE 6.7: Server Certificates

TASK 5 Edit Bindings

14. Expand the **Sites** node, and select **GoodShopping** in the **Connections** pane, and click **Bindings** in the **Actions** pane.



FIGURE 6.8: Editing Site Bindings

15. The **Site Bindings** wizard appears; click **Add**.

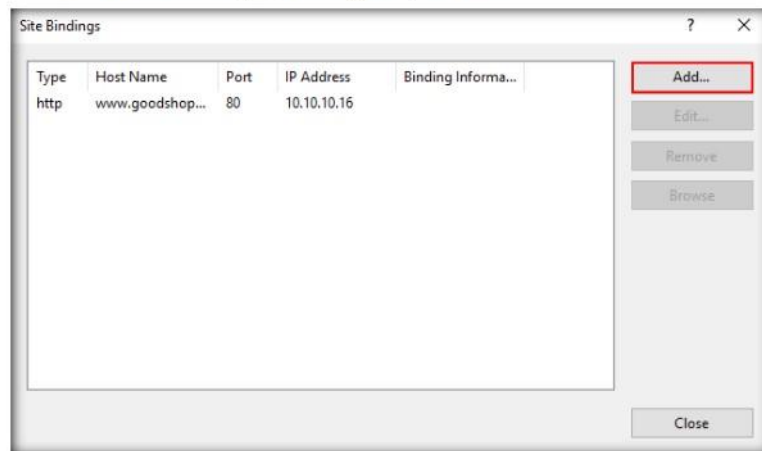



FIGURE 6.9: Site Bindings Wizard

In a web of trust certificate scheme there is no central CA, and so identity certificates for each user can be self-signed. In this case, however, it has additional signatures from other users which are evaluated to determine whether a certificate should be accepted as correct or not.

- The **Add Site Binding** window appears; choose **https** from the **Type:** field drop-down list and click **OK**.

 A certificate serves two essential purposes: distributing the public key and verifying the identity of the server so that visitors know they aren't sending their information to the wrong person.

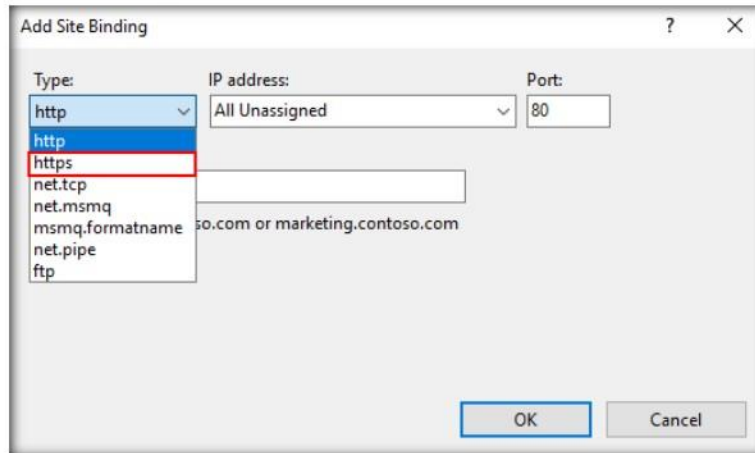



FIGURE 6.10: Adding Site Bindings

- Once you choose the **https** channel in the **Port** field, it will automatically change to **443** (the channel on which HTTPS runs).
- Choose the **IP address** in which the site is hosted, or leave the default setting.
- Specify the **Host name** **www.example.com**. In this lab, you will be applying certificate for the **Goodshopping** site.

 It can properly verify the identity of the server only when it is signed by a trusted third party because any attacker can create a self-signed certificate and launch a man-in-the-middle attack.

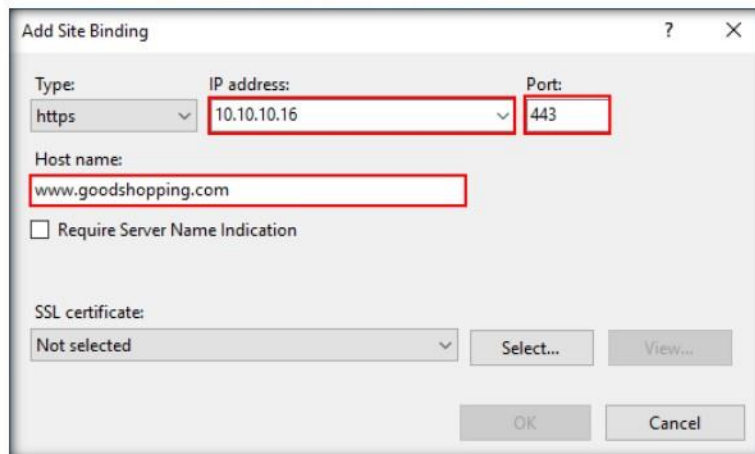



FIGURE 6.11: Adding Site Bindings-Host Name

Module 20 - Cryptography

20. In the **SSL certificate** field, choose Goodshopping from the drop-down list, and click **OK**.

 If a user just accepts a self-signed certificate, an attacker could eavesdrop on all the traffic or try to set up an imitation server to phish additional information out of the user.

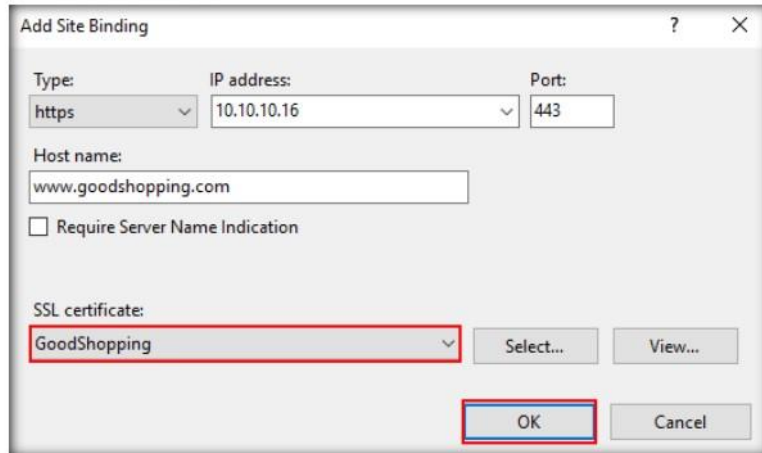



FIGURE 6.12: Adding Site Bindings-SSL Certificate

 An Intranet. When clients only have to go through a local Intranet to get to the server, there is virtually no chance of a man-in-the-middle attack.

21. In the **Site Bindings** wizard, the newly created SSL certificate is added, as shown in the screenshot. Click **Close**.

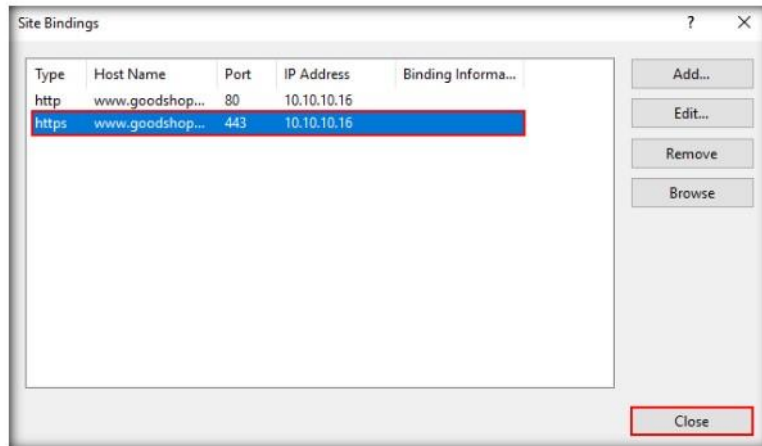



FIGURE 6.13: Added HTTPS Channel

Module 20 - Cryptography

22. Now, right-click the name of the site for which you have created the self-signed certificate, and click **Refresh** from the context menu. Minimize the IIS Manager window.

 A development server. There is no need to spend extra cash buying a trusted certificate when you are just developing or testing an application.

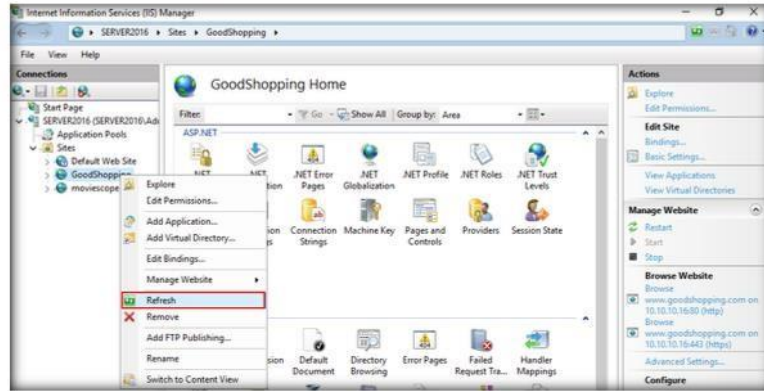


FIGURE 6.14: Added HTTPS Channel

23. Open a browser, type **https://www.goodshopping.com** in the address bar, and press **Enter**.

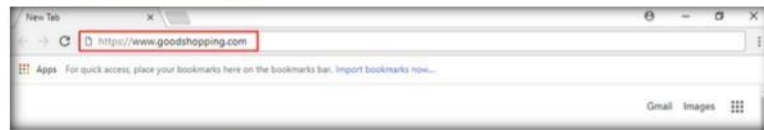



FIGURE 6.15: www.goodshopping.com before adding Certificate

 Personal sites with few visitors. If you have a small personal site that transfers non-critical information, there is very little incentive for someone to attack the connections.

24. As you are using an https channel to browse, it displays a page stating that the connection is not private; click **ADVANCED** to proceed.

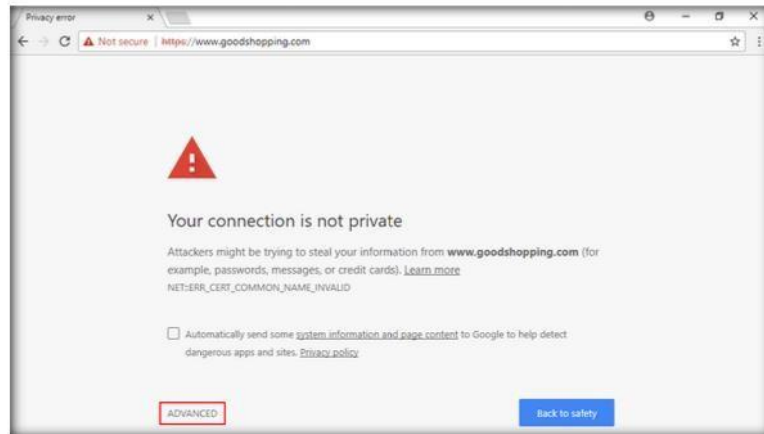


FIGURE 6.16: Connection is not Private

Module 20 - Cryptography

25. Click **Proceed to www.goodshopping.com (unsafe)**.

📖 Creating a self-signed certificate in IIS 7 is much easier to do than in previous versions of IIS. IIS now provides a simple interface for generating a self-signed certificate. One drawback is that the common name of the certificate is always the server name instead of the site name.

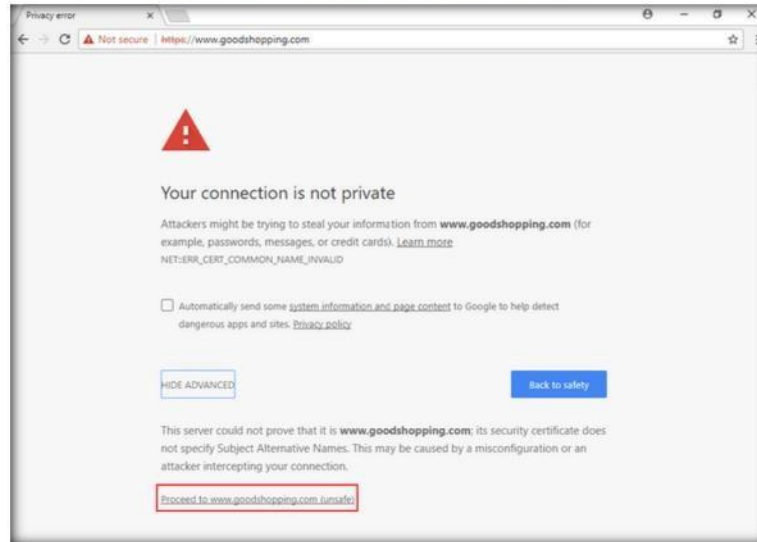


FIGURE 6.17: Proceed to Unsafe Page

26. Now you can see the **Goodshopping webpage** with **ssl certificate** assigned to it, as shown in the screenshot:

📖 If SSL utilizes public key cryptography to encrypt the data stream traveling over the Internet, why is a certificate necessary? The technical answer to that question is that a certificate is not really necessary - the data is secure and cannot easily be decrypted by a third party.

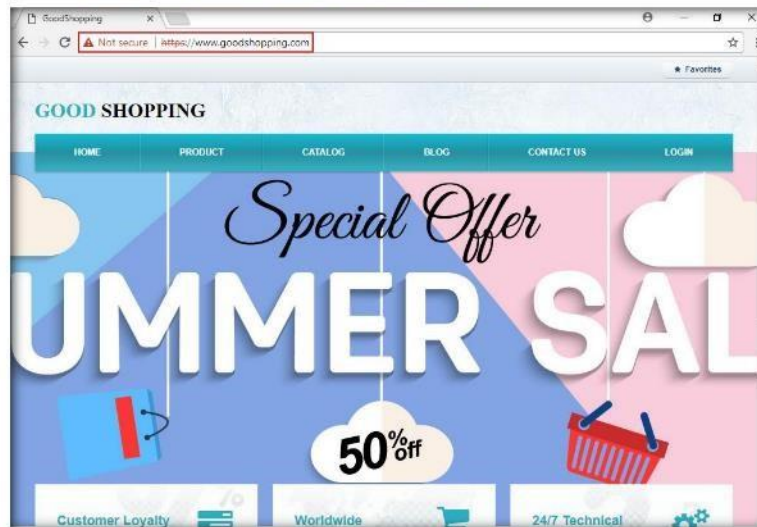


FIGURE 6.18: Self-Signed Certificate Page

Lab Analysis

Analyze and document the results related to this lab exercise.

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





Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs



Basic Disk Encryption 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.

ICON KEY

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


Lab Scenario

Disk encryption encrypts all data on a system, including the files, folders, and the operating system. This is most appropriate when the physical security of the system is not assured. Examples include traveling laptops or desktops that are not in a physically secured area. When properly implemented, Disk Encryption provides an enhanced level of assurance that the data, while encrypted, cannot be viewed or otherwise discovered by unauthorized parties in the event of theft, loss, or interception.

Lab Objectives

This lab will give you experience in encrypting data and show you how to do so. It will teach you how to:

- Create a virtual encrypted disk with a file

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**

Lab Environment

To complete this lab, you will need:

- VeraCrypt located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Disk 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 might differ
- Run this tool in Windows Server 2016
- Follow the wizard driven installation instructions

Module 20 - Cryptography

- Administrative privileges to run tools

Lab Duration

Time: 10 Minutes

Overview of VeraCrypt

VeraCrypt is a software application used for On-The-Fly Encryption (OTFE). It can create a virtual encrypted disk within a file, or encrypt a partition or entire storage device. It is distributed free of cost, and the source code is available.

TASK 1

Create a Volume

Lab Tasks

1. Click **Start** menu present on the lower-left corner of the **Desktop**.



FIGURE 7.1: Windows Server 2016 – Desktop view

2. The **Start menu** appears, scroll down to view installed apps.

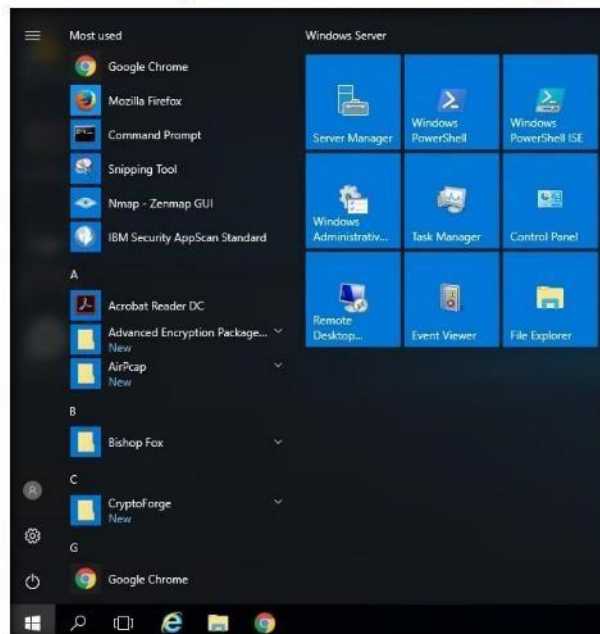



FIGURE 7.2: Windows Server 2016 – Apps

Module 20 - Cryptography

3. Click **VeraCrypt** to launch the application.

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

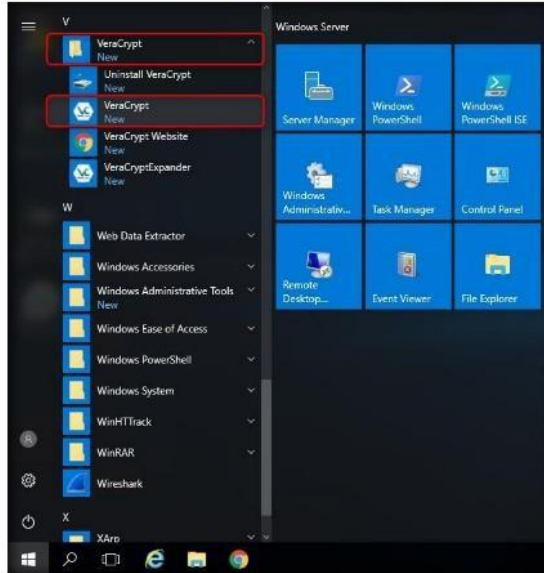


FIGURE 7.3: Windows Server 2016 – Apps

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

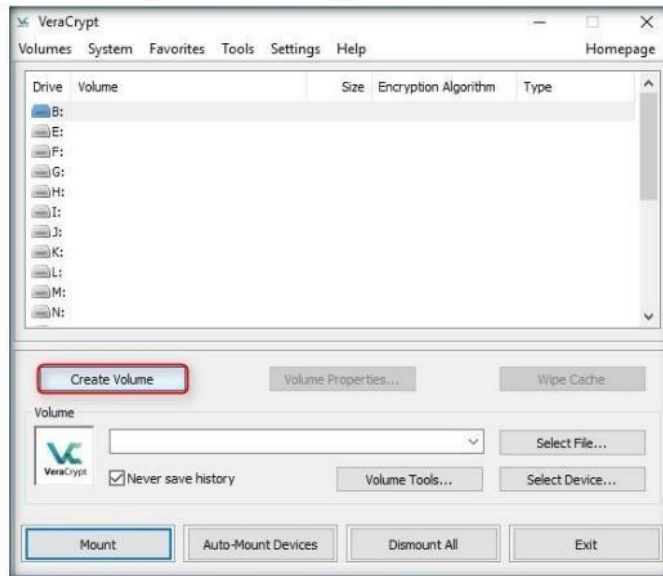


FIGURE 7.4: VeraCrypt Main window

Module 20 - Cryptography

5. The **VeraCrypt Volume Creation Wizard** window appears.
6. Select **Create an encrypted file container** to create a file containing a virtual, encrypted disk and click **Next** to proceed.


 **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 you are creating now.



FIGURE 7.5: VeraCrypt Volume Creation Wizard

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


 **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).



FIGURE 7.6: VeraCrypt Volume Creation Wizard-Volume Type

Module 20 - Cryptography

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

VeraCrypt supports a concept called plausible deniability.

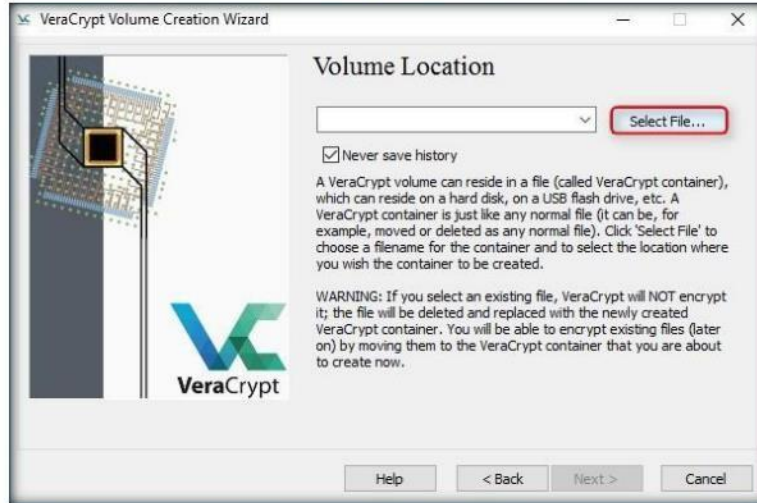


FIGURE 7.7: VeraCrypt Volume Creation Wizard-Volume Location

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

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

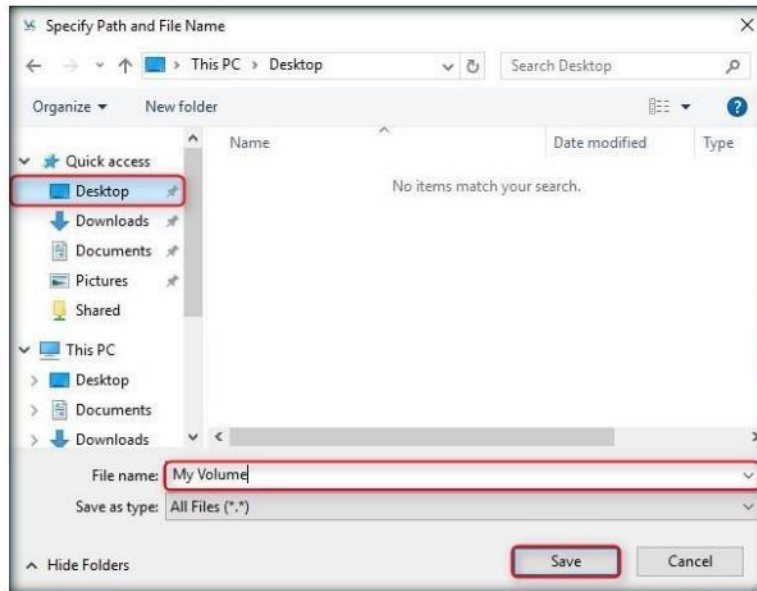


FIGURE 7.8: Windows Standard-Specify Path and File Name Window

Module 20 - Cryptography

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

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

VeraCrypt currently supports the following hash algorithms:

- RIPEMD-160
- SHA-512
- Whirlpool

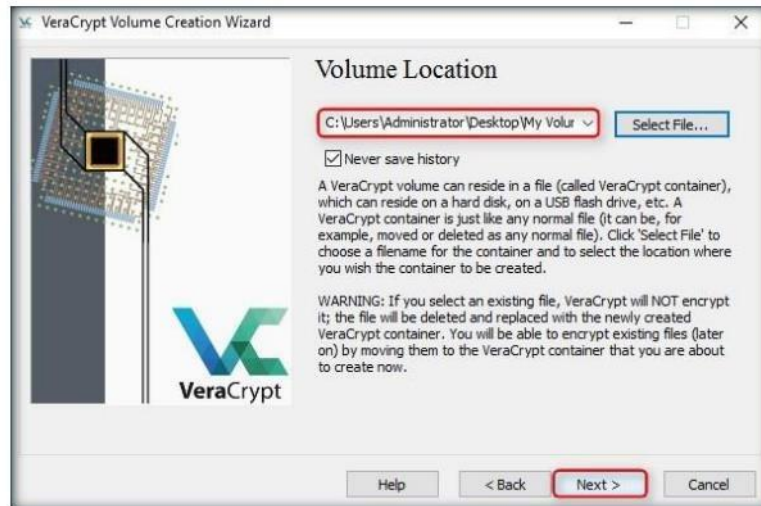


FIGURE 7.9: VeraCrypt Volume Creation Wizard-Volume Location

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

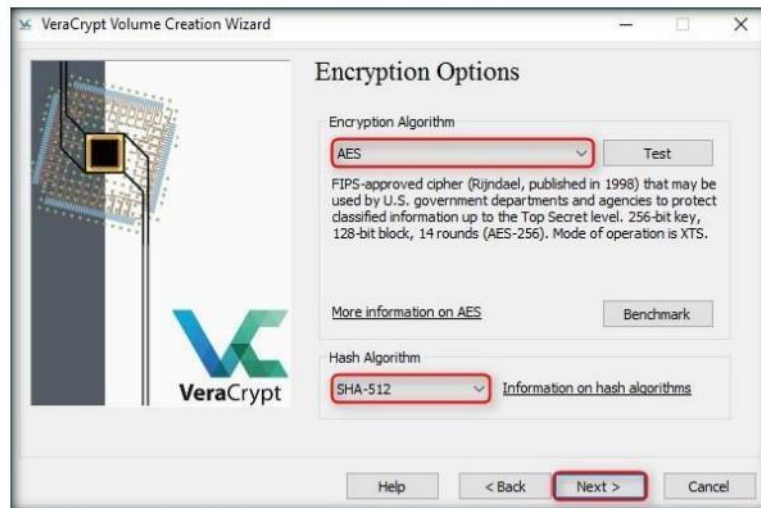


FIGURE 7.10: VeraCrypt Volume Creation Wizard-Encryption Options

Module 20 - Cryptography

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

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

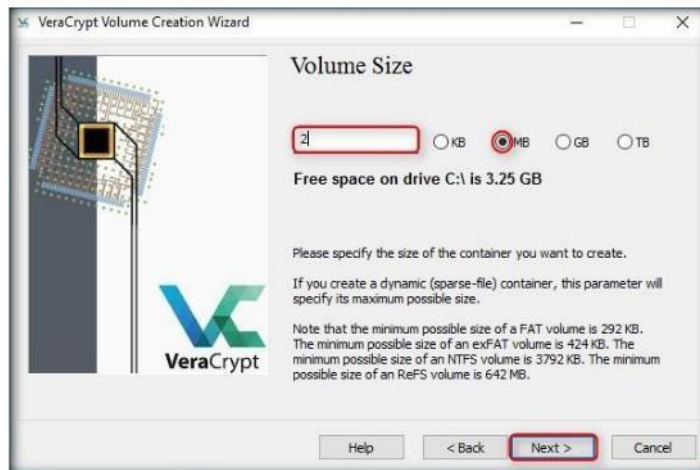


FIGURE 7.11: VeraCrypt Volume Creation Wizard-Volume Size

14. The **Volume Password** wizard appears; provide a **good password** in the **Password** field, retype it in the **Confirm** field, and click **Next**.
15. In this lab, the password used is **qwerty@123**.

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

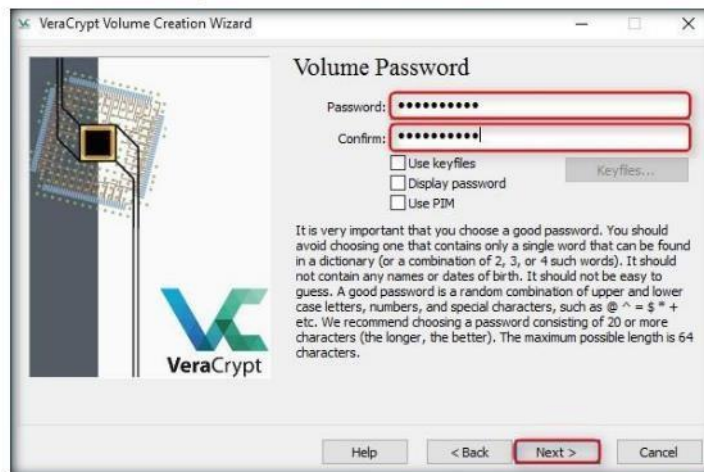


FIGURE 7.12: VeraCrypt Volume Creation Wizard-Volume Password

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

16. The Volume Format option appears. Select **FAT Filesystem**, and set the cluster to **Default**.

Module 20 - Cryptography

17. Move your mouse as **randomly** as possible within the Volume Creation Wizard window for at least **30 seconds**.
18. Click **Format**.

VeraCrypt volumes have no "signature" or ID strings. Until decrypted, they appear to consist solely of random data.

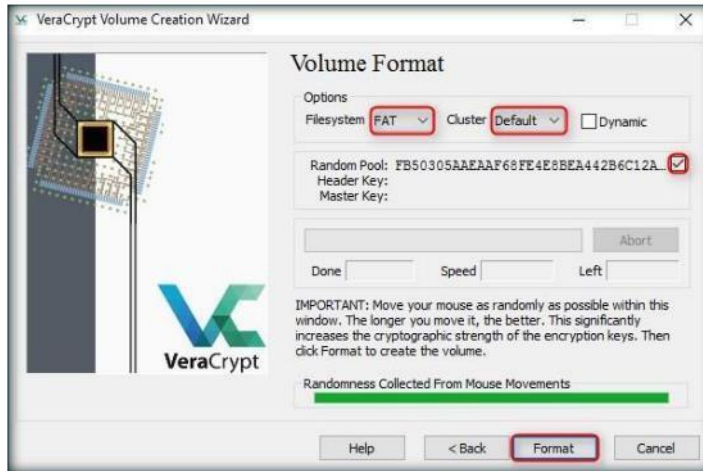


FIGURE 7.13: VeraCrypt Volume Creation Wizard-Volume Format

19. After clicking **Format**, VeraCrypt will create a file called **My Volume** in the provided folder. This file depends on the VeraCrypt container (it will contain the encrypted VeraCrypt volume).
20. Depending on the **size of the volume**, it may take some time for volume creation.
21. 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.

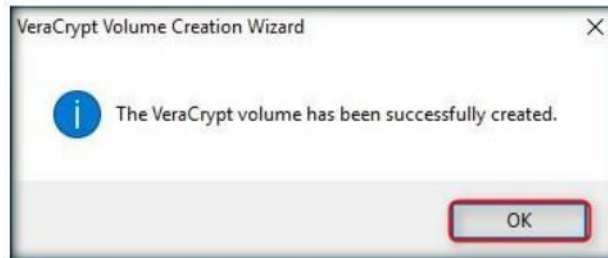


FIGURE 7.14: VeraCrypt Volume Creation Wizard Dialog Box

22. Click **OK** to close the dialog box.
23. You have **successfully** created a **VeraCrypt volume** (file container).

Module 20 - Cryptography

24. In the VeraCrypt Volume Creation wizard window, click **Exit**.

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

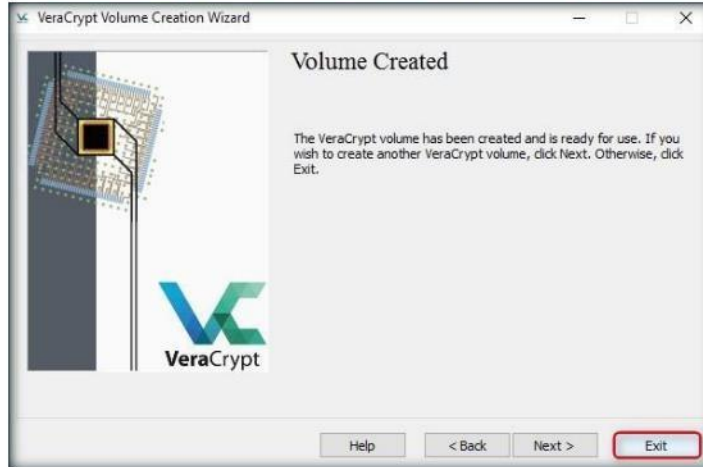


FIGURE 7.15: VeraCrypt Volume Creation Wizard-Volume Created

TASK 2

Mount a Volume

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

Mount Options affect the parameters of the volume being mounted. The Mount Options dialog can be opened by clicking on the Mount Options button in the password entry dialog box.

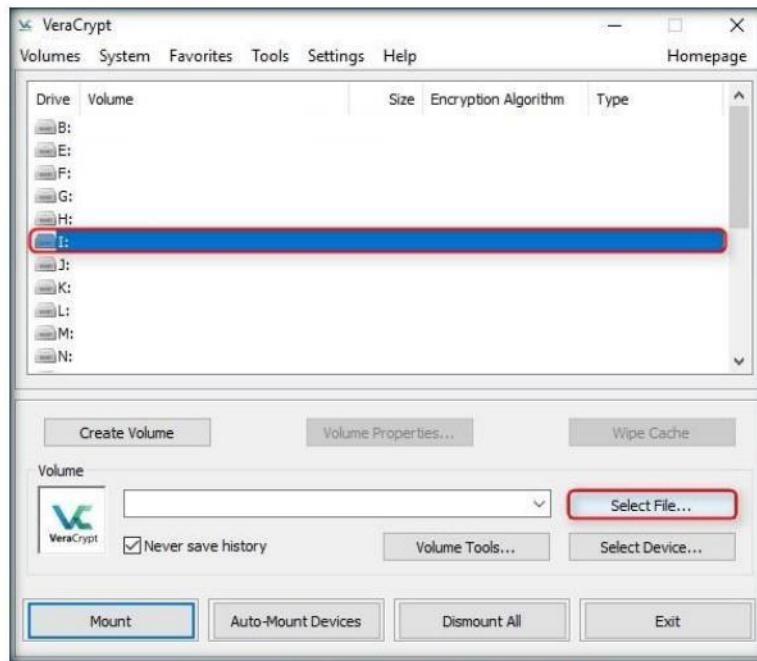


FIGURE 7.16: VeraCrypt Main Window with Select File Button

Module 20 - Cryptography

26. The **Select a VeraCrypt Volume** window appears; navigate to **C:\Users\Administrator\Desktop**, click **My Volume**, and click **Open**.

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

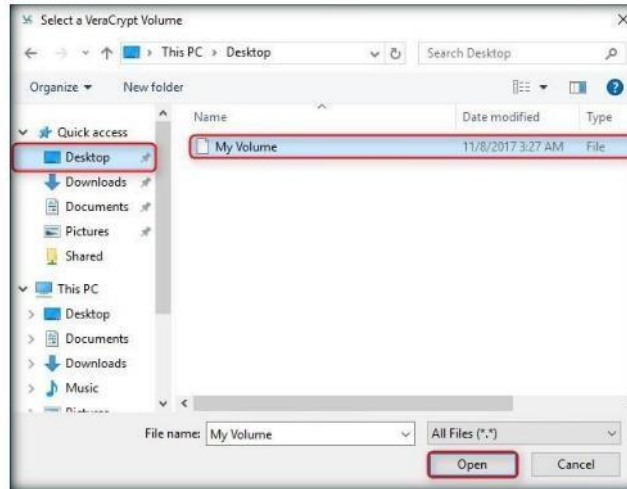


FIGURE 7.17: Windows Standard File Selector Window

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

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.

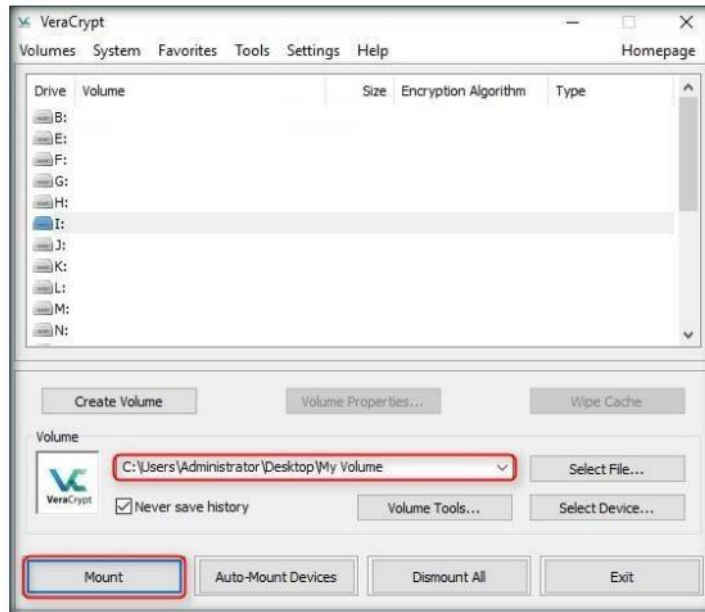


FIGURE 7.18: VeraCrypt Main Window with Mount Button

Module 20 - Cryptography

28. 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**.

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.



FIGURE 7.19: VeraCrypt Password Window

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

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

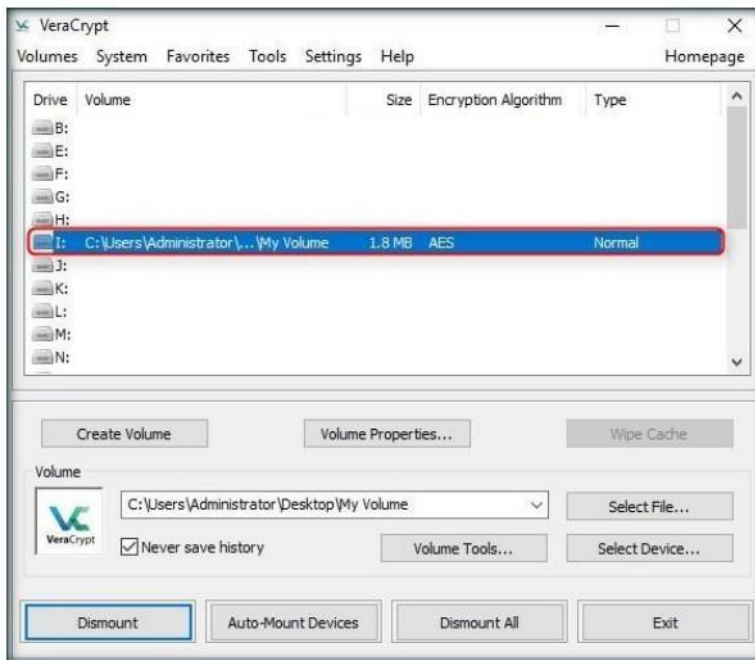



FIGURE 7.20: VeraCrypt Main Window

30. **My Volume** has **successfully** mounted the container as a virtual disk (**I:**).
31. The virtual disk is entirely **encrypted** (including file names, allocation tables, free space, etc.) and behaves like a **real disk**.

Module 20 - Cryptography

32. You can copy or move files to this virtual disk to encrypt them.
33. Create a text document on the **Desktop** and name it **Test**.
34. Open the **text document**, and enter some text in it.
35. Click **File** in the menu bar, and click **Save**.

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

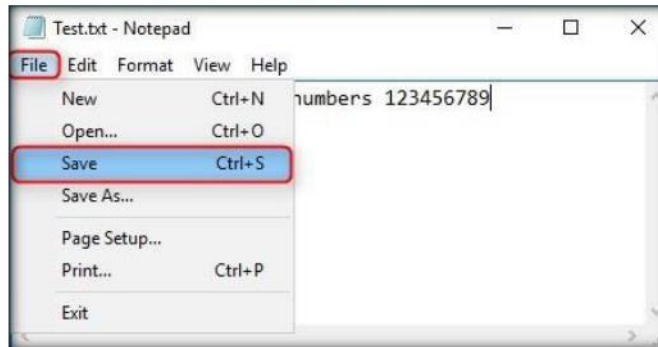


FIGURE 7.21: VeraCrypt Main Window with Dismount Button

36. Copy the file from the **Desktop**, and paste it in **I**. **Close** the window.

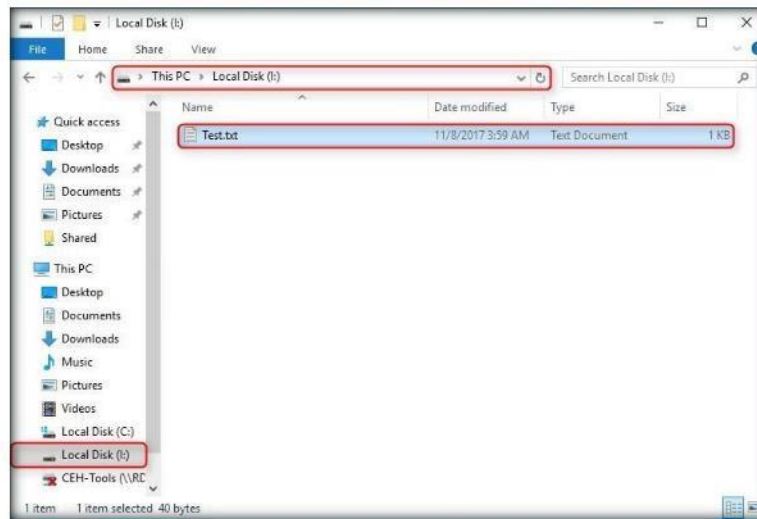


FIGURE 7.22: Test.txt file in Encrypted Container

Module 20 - Cryptography

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

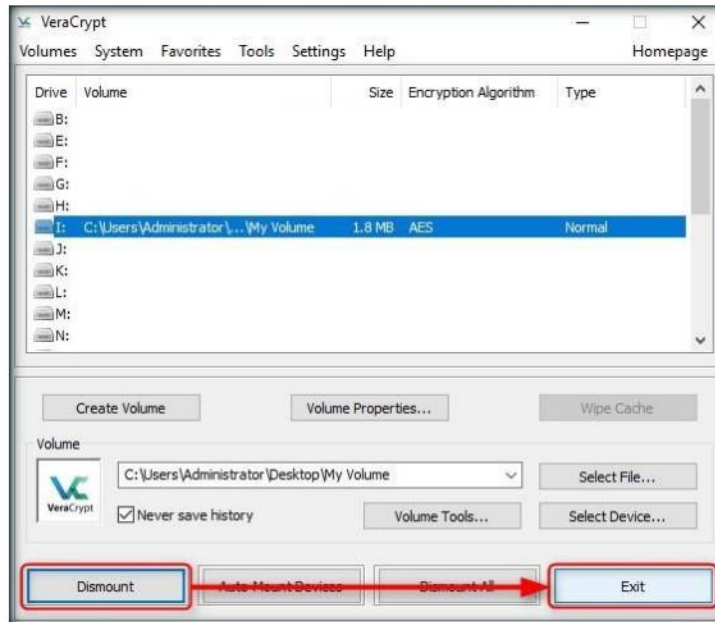


FIGURE 7.23: VeraCrypt Main Window with Dismount Button

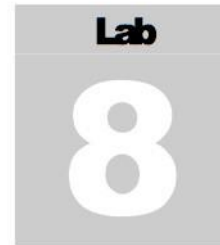
38. The **I:** located in **This PC** disappears. This lab is used to demonstrate that, in cases of system hacks, if an attacker manages to gain remote access or complete access to the machine, he/she cannot find the encrypted volume—including its files—unless he/she is able to obtain the password. Thus, all sensitive information located on the encrypted volume is safeguarded.

Lab Analysis

Analyze and document the results related to this lab exercise.

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

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs



Basic Data Encrypting using Rohos Disk Encryption

Rohos Disk is a program used to create hidden and protected partitions on a computer or USB flash drive, that password-protects/locks access to your Internet applications.

Lab Scenario

Disk encryption works in a manner similar to text-message encryption. By using an encryption program for the user's disk, the user can safeguard all information burned onto the disk and save it from falling into the wrong hands. Disk-encryption software scrambles the information on the disk into an illegible code. The information must be decrypted to be read and used. To be an expert ethical hacker and penetration tester, you must have knowledge of these cryptography functions.

Lab Objectives





This lab will give you experience on encrypting data and show you how to do it. It will teach you how to:

- Create an encrypted drive for Windows
- Create a virtual encrypted drive for an external USB

Lab Environment

To complete this lab, you will need:

- Rohos Disk Encryption located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Disk Encryption Tools\Rohos Disk Encryption**
- You can also download the latest version of Rohos Disk Encryption from the link **<http://www.rohos.com/products/rohos-disk-encryption/>**
- If you decide to download the latest version, then screenshots shown in the lab might differ
- Follow the wizard driven installation instructions

ICON KEY	
	Valuable information
	Test your knowledge
	Web exercise
	Workbook review

Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography

Module 20 - Cryptography

- Windows Server 2016 running in host machine
- Administrative Privileges to run the tool

Lab Duration

Time: 15 Minutes

Overview of Rohos Disk Encryption

Rhos Disk Encryption creates hidden and password-protected partitions on a computer or USB flash drive. It uses an NIST-approved AES encryption algorithm with 256-bit encryption key length. Encryption is automatic and on-the-fly.

Lab Tasks

Note: Plug in a USB device to your machine before performing this lab.

1. To install Rohos Disk Encryption, navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Disk Encryption Tools\Rhos Disk Encryption**.
2. Double-click **rohos.exe**; the **Select Setup Language** dialog box appears.
3. Select the language as **English**, and click **OK**.

 **TASK 1**

Install Rohos Disk Encryption


 You can also download Rohos from <http://www.rohos.com>.



FIGURE 8.1: Select the Language

Module 20 - Cryptography

4. The **Setup** window appears; read the instruction, and click **Next**.

Portable Rohos Disk Browser allows to use encrypted partition on any PC without Admin rights, without installation.



FIGURE 8.2: Rohos setup wizard

5. The **License Agreement** window appears; read the agreement carefully, select **I accept the agreement**, and click **Next**.

Encryption is automatic and on-the-fly. AES 256 bit key length. Using NIST compliant encryption standards.



FIGURE 8.3: License agreement window

Module 20 - Cryptography


6. Select the location in which you want the program to place the shortcut.

File Virtualization: prevents secret data leak outside encrypted disk on TEMP folders, Registry, Recent documents list, etc.



FIGURE 8.4: Select the destination folder

7. Check **Create a desktop**, and click **Next**.

 Any file or folder can be easily moved into Encrypted Rohos Disk with shredding afterwards.

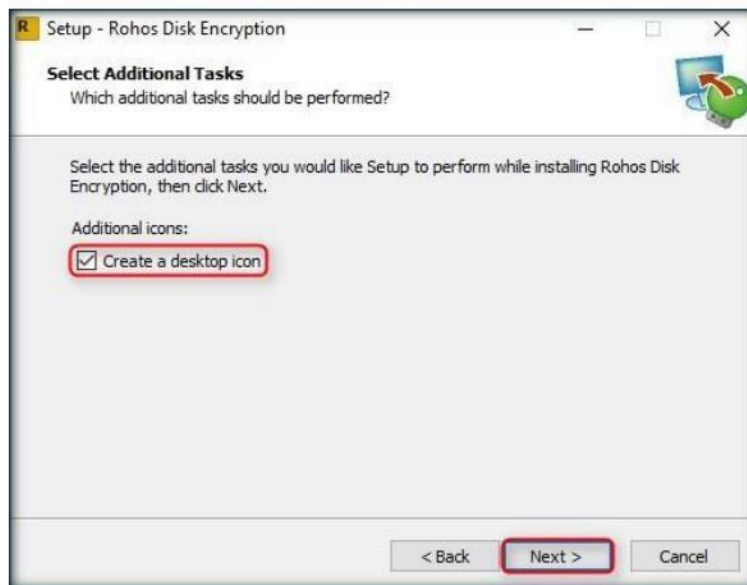


FIGURE 8.5: creating Rohos desktop icon

Module 20 - Cryptography

8. Click **Install** to begin installation.

Secure virtual keyboard - protect encrypted disk password from a keylogger.

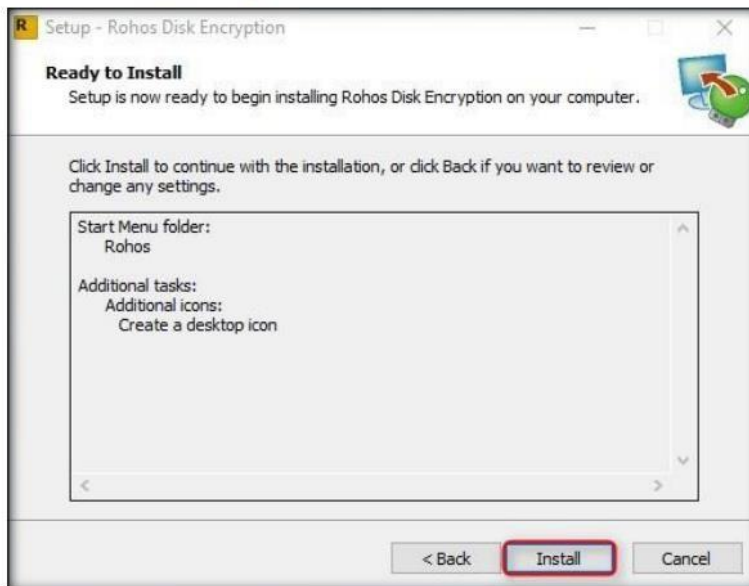


FIGURE 8.6: Rohos disk encryption installation

9. On completion of installation, click **Finish**.

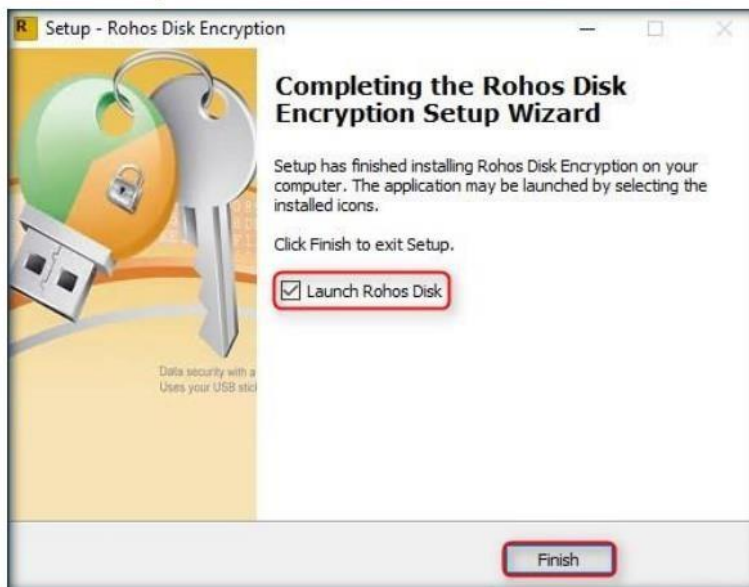


FIGURE 8.7: installation of Rohos disk encryption completed

TASK 2
Create an encrypted disk for Local Machine

10. The **Rohos Disk Encryption** window appears, click **Create new disk...**



FIGURE 8.8: Create new disk



FIGURE 8.9: Select password for accessing the disk

11. Wait until the encrypted volume is created.



FIGURE 8.10: Disk creation in progress

Rohos disk uses NIST approved AES encryption algorithm, 256-bit encryption key length.

This option brings affordable and an AES 256 strength encryption solution to improve security issues by preventing unauthorized access to your Internet apps, such as Google Chrome, Firefox.

Rohos cares about usability: Your first Encrypted Drive can be turned on with a single click or automatically on system startup.

Module 20 - Cryptography

- On creating the encrypted volume, a new **3000 MB (2.92 GB)** drive (**R:**) appears in **This PC**, as shown in the screenshot:

Partition password reset option allows creating a backup file to access your secured disk if you forgot your password or lost USB key.

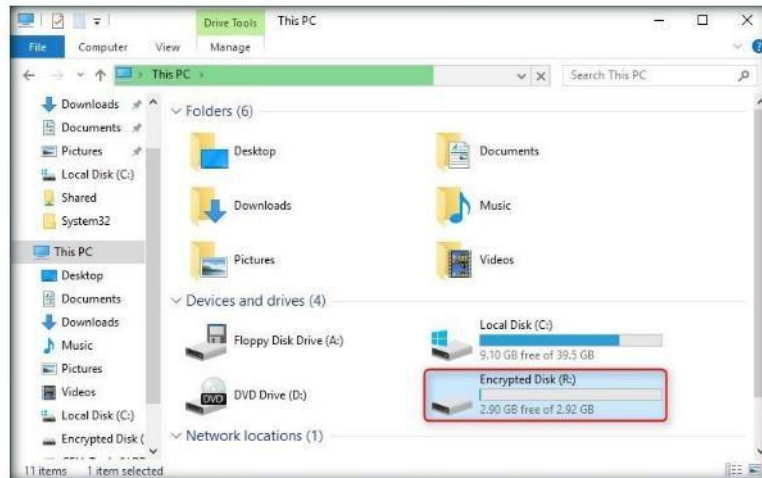


FIGURE 8.11: Encrypted disk successfully created

- This drive appears only when you are connected to **Rohos Disk Encryption**, and disappears when you exit it.
- If you want to hide any important files/directories from anyone accessing your system, you can place them in this drive and access them whenever required (by launching Rohos and entering the password).
- To create an encrypted USB drive, click **Encrypt USB drive** in the Rohos Disk Encryption GUI.

TASK 3

Create an Encrypted Disk USB Disk



FIGURE 8.12: Encrypting a USB device

Module 20 - Cryptography

16. The **Encrypt USB drive** dialog box appears; click **Change...** in the Encrypted partition properties section.



FIGURE 8.13: Encrypt USB drive dialog-box

17. The **Disk details** window appears; choose the Disk letter **M:**, set the disk size to **60**, and click **OK**.



FIGURE 8.14: Disk details window

Module 20 - Cryptography

18. This creates an Encrypted USB drive (**M:**) of 60 MB.
19. You need to apply a password for the disk, so that whenever someone wants to access the drive, they need to specify the password.
20. Specify the password (here, **test@123**) in both fields, and click **Create disk**.

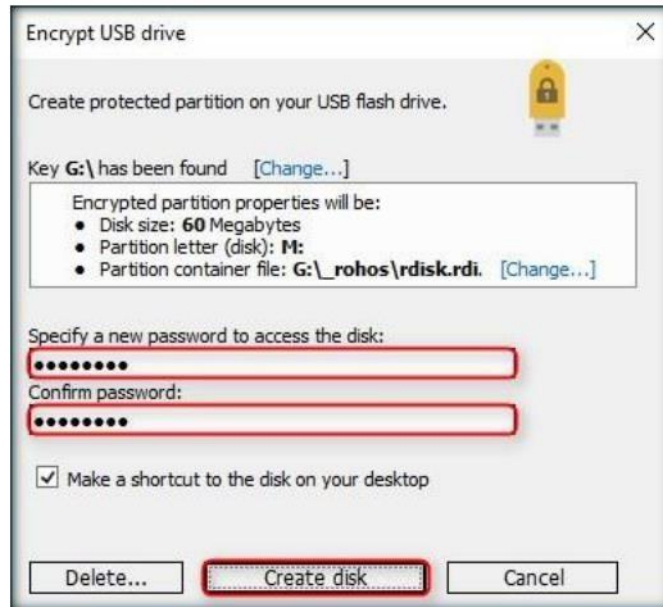


FIGURE 8.15: Encrypt USB drive window

21. Wait until the disk is created.



FIGURE 8.16: Disk creation in progress

Module 20 - Cryptography

22. On successful creation of the disk, a **Rohos Disk Encryption** dialog box appears; click **OK**.



FIGURE 8.17: Rohos Disk Encryption dialog-box

23. The **Encrypted disk** (here, **M:**) of **60 MB** is created successfully, as shown in the screenshot:

You can open or Save your protected documents right from MS Word (Excel) by clicking on the personal disk icon.

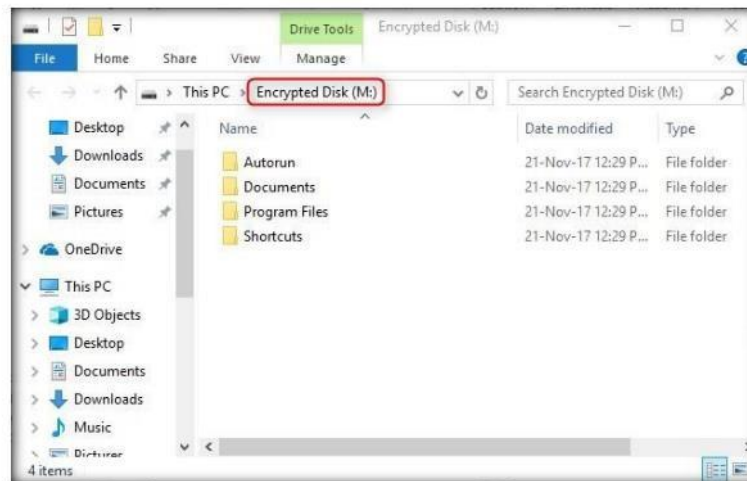


FIGURE 8.18: Newly created Encrypted disk window

24. The files you place in this drive will automatically be placed in the external USB.

Module 20 - Cryptography

25. In this lab, the folder **Rohos Disk Encryption** is being copied from **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Disk Encryption Tools** to **M:**.

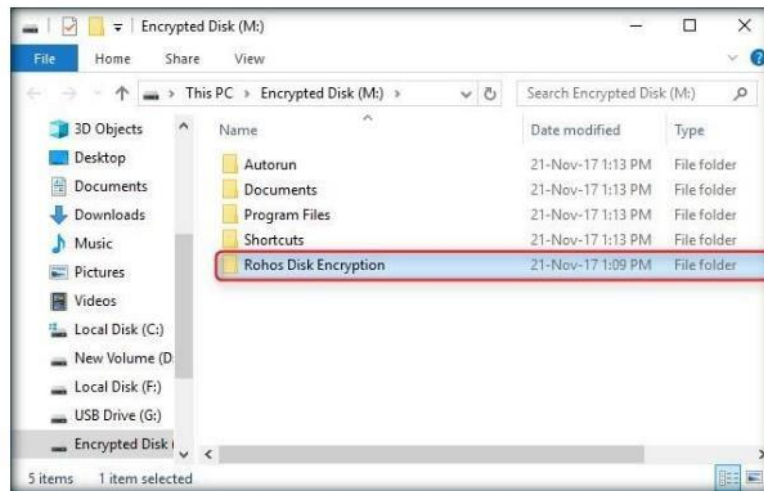


FIGURE 8.19: Copying a folder to the encrypted disk

TASK 4

Access Files in the Encrypted Disk

26. Now, if you want to access this file, open the external USB drive which has been connected to your computer, and double-click **Rohos Mini Drive (Portable).exe**.

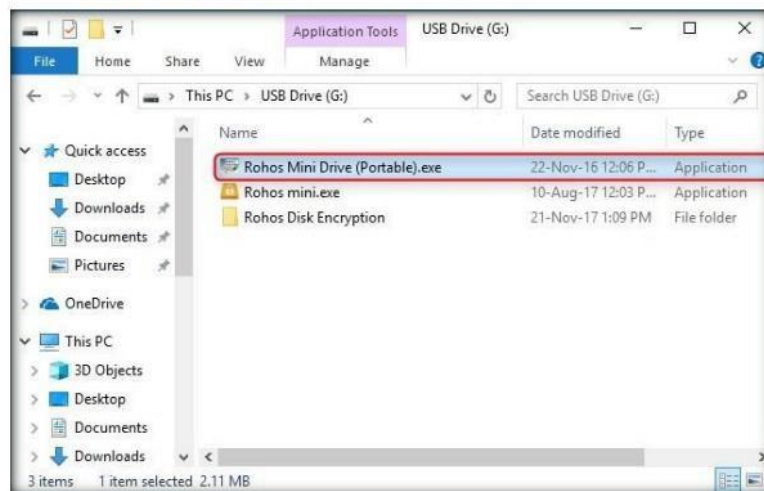


FIGURE 8.20: Launching Rohos Mini Drive

Module 20 - Cryptography

27. A **Rohos** dialog box appears asking you to enter the password. You need to enter the password which you had specified at the time of creating the encrypted USB disk (**M:**).



FIGURE 8.21: Rohos dialog-box

28. A **Rohos Disk Browser** window appears, displaying the folder that was placed in **M:**, as shown in the screenshot:

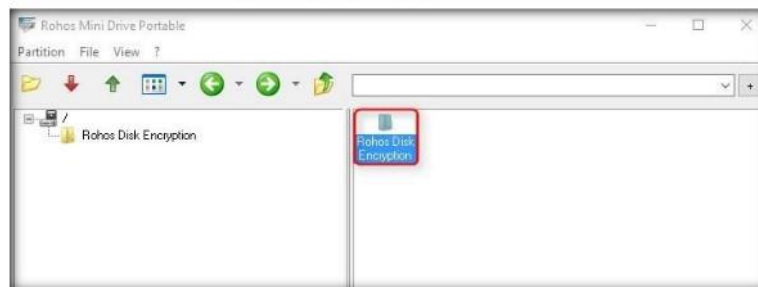


FIGURE 8.22: Rohos Disk Browser window containing the file placed in M:\

29. When you want to share sensible information with someone via USB, you can use this application to store the files in an encrypted disk, and share the password with that person.
30. The person with whom you want to share the files can access them only after entering the correct password.
31. This way, you can protect the files from being viewed by a third person and thereby safeguard them.

Lab Analysis

Analyze and document the results related to this lab exercise.

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





Internet Connection Required	
<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input type="checkbox"/> iLabs



Basic Data Encryption using CrypTool

CrypTool is a freeware program that enables you to apply and analyze cryptographic mechanisms. It has the typical look and feel of a modern Windows application. CrypTool includes every state-of-the-art cryptographic function and allows you to learn and use cryptography within the same environment.

ICON KEY

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


Lab Scenario

Most security initiatives are defensive strategies aimed at protecting the perimeter of the network. But these efforts may ignore a crucial vulnerability sensitive data stored on networked servers and are at risk from attackers who only need to find one way inside the network to access this confidential information. Additionally, perimeter defenses like firewalls cannot protect stored sensitive data from internal threats such as employees who have the means to access and exploit this data. Encryption can provide strong security for sensitive data stored on local or network servers. To be an expert ethical hacker and penetration tester, you must have knowledge of cryptography functions.

Lab Objectives

This lab will give you experience on encrypting data and show you how to do so. It will teach you how to:

- Use encrypting/decrypting command
- Visualize several algorithms
- Calculate hash values and analysis

 **Tools demonstrated in this lab are available in Z:\CEH-Tools\CEHv10 Module 20 Cryptography**


Lab Environment

To complete this lab, you will need:

- CrypTool located at **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptanalysis Tools\CrypTool**

Module 20 - Cryptography

- You can also download the latest version of CrypTool from the link <http://www.cryptool.org/en>
- If you decide to download the latest version, then screenshots shown in the lab might differ
- Windows Server 2016 running on host machine
- Windows 10 running on virtual machine
- Administrative Privileges to run the tool

 CrypTool is a free e-learning application for Windows.

Lab Duration

Time: 10 Minutes


Overview of CrypTool

CrypTool is a free, open-source e-learning application used in the implementation and analysis of cryptographic algorithms. It was originally designed for internal business application for information security training.

Lab Tasks

TASK 1 Encrypting the Data

1. Navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptanalysis Tools\CrypTool**, double-click **SetupCrypTool_1_4_40_en.exe**, and follow the wizard driven installation steps to install the application.
2. On completing the installation, launch **CrypTool** application from the **Apps** list.

 You can also download CrypTool from <http://www.cryptool.org>.

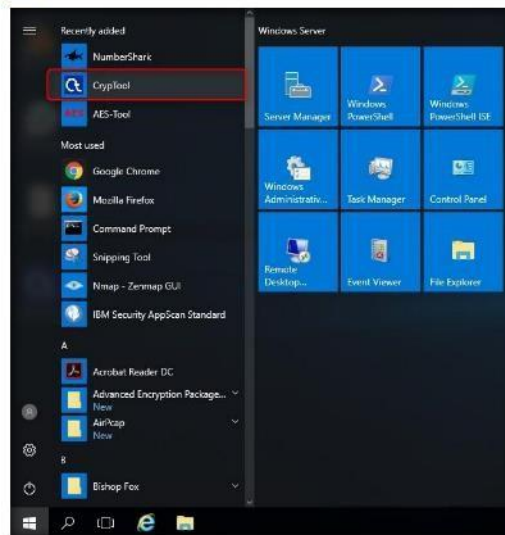


FIGURE 9.1: Launching CrypTool from Apps list

- The **How to Start** dialog box appears; check **Don't show this dialog again**, and click **Close**.


 CrypTool Online provides an exciting insight into the world of cryptology with a variety of ciphers and encryption methods.



FIGURE 9.2: How to Start Dialog box

- The main window of **CrypTool** appears; close the **startingexample-en.txt** window.

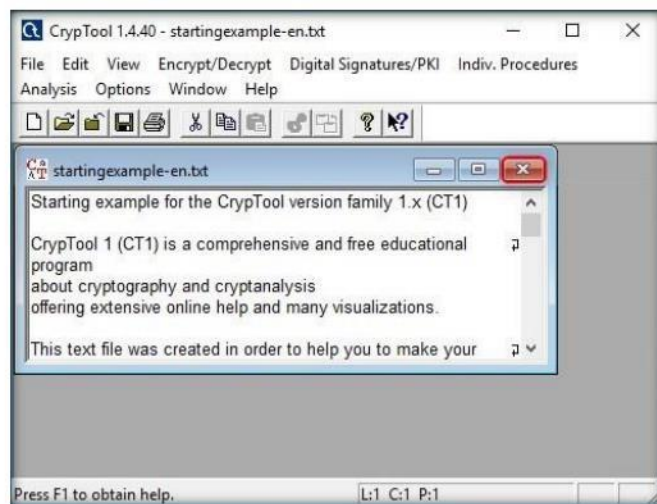


FIGURE 9.3: startingexample-en.txt window in CrypTool

Module 20 - Cryptography

- To **encrypt** data, click the **File** option from the menu bar, and select **New**.

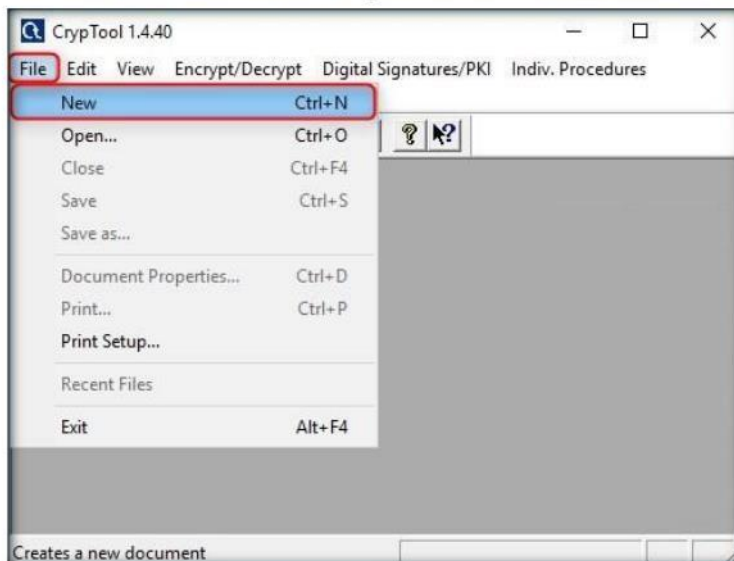



FIGURE 9.4: Choosing a new file to crypt

 CrypTool was originally designed for internal business application for information security.

- Type some content in the opened **Unnamed1** Notepad of CrypTool. You will be encrypting this content.
- Select **Encrypt/Decrypt** → **Symmetric (modern)** → **RC2...** in the Menu bar.

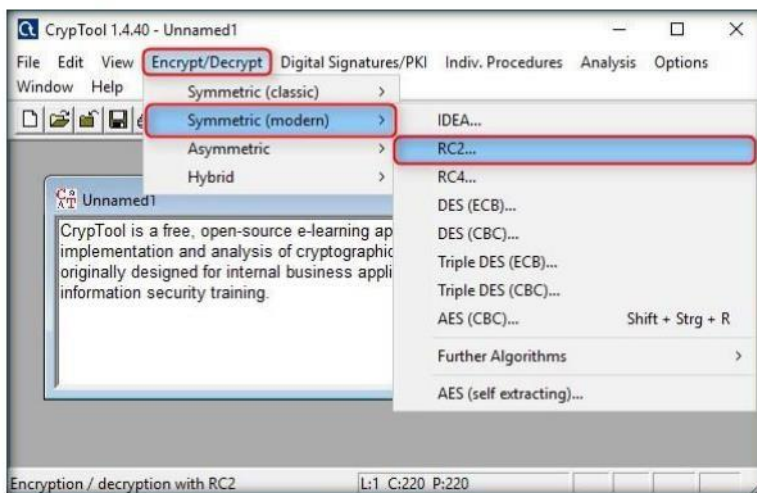


FIGURE 9.5: Encrypting the file

Module 20 - Cryptography

- The **Key Entry: RC2** dialog box appears; select **Key length** (here, **8 bits**) from the drop-down list.
- Enter the key using **hexadecimal characters (05)**, and click **Encrypt**.

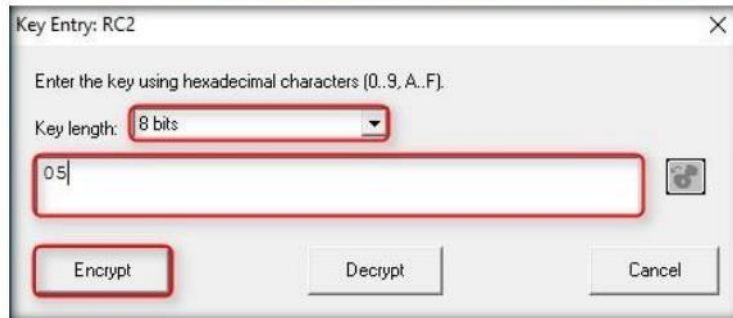


FIGURE 9.6: Encrypting the file

- The RC2 encryption of **Unnamed1** notepad displays, as shown in the screenshot:

GrypTool includes every state-of-the-art cryptographic function and allows you to learn and use cryptography within the same environment.

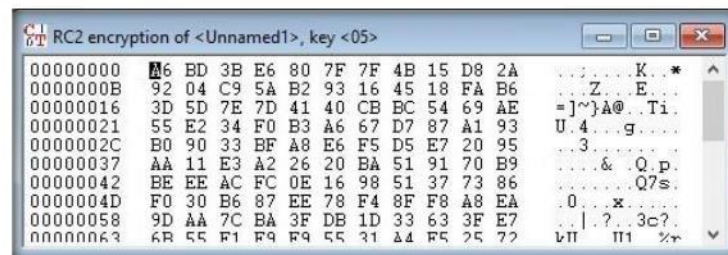


FIGURE 9.7: Output of RC2 - encrypted data

- To save the file, click **File** in the menu bar, and select **Save**.

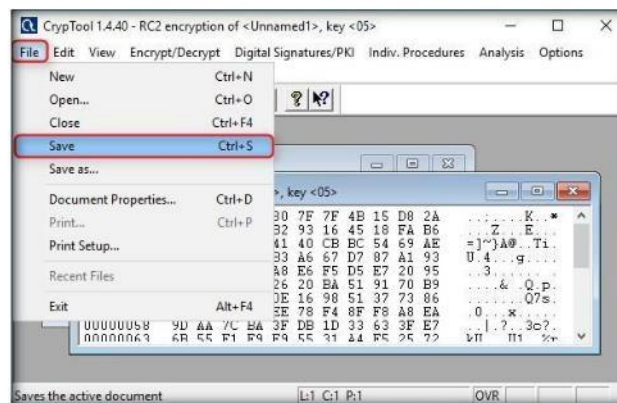



FIGURE 9.8: Saving the encrypted file

Module 20 - Cryptography

12. The **Save As** dialog-box appears; choose a location where you want to save the file (**Desktop**), specify a file name (**Cry-RC2-Unnamed1.hex**), and click **Save**.

Note: The file name may differ in your lab environment.

 CrypTool Online provides an exciting insight into the world of cryptology with a variety of ciphers and encryption methods.

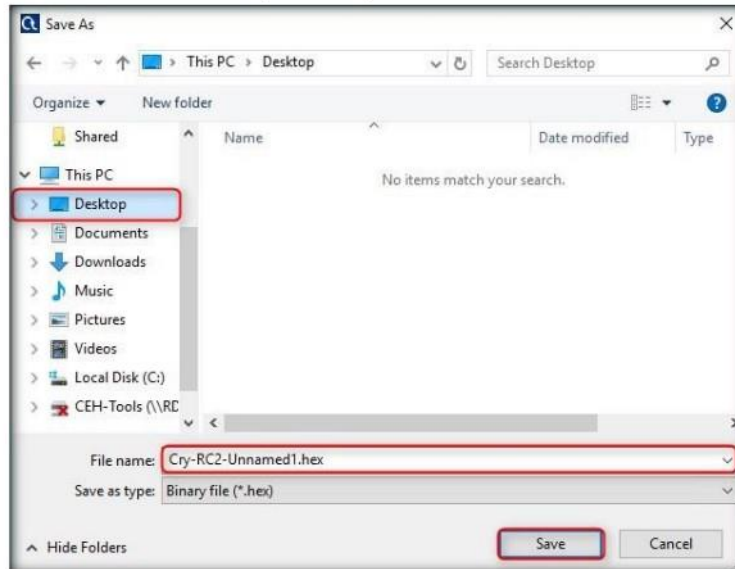


FIGURE 9.9: Saving the encrypted file

13. Now, you can send this file to the intended person by email or any other means and provide him/her with the hex value, which will be used to decrypt the file.
14. To share the file, you may copy the encrypted file from the Desktop to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptanalysis Tools\CrypTool**.
15. Assume that you are the intended recipient (working on Windows 10) of the Crypted file, through the shared network drive.
16. Log into **Windows 10** virtual machine, navigate to **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptanalysis Tools\CrypTool**, double-click **SetupCrypTool_1_4_40_en.exe**, and follow the steps to install the application.
17. In the meanwhile, copy the Crypted hex file (**Cry-RC2-Unnamed4.hex**) from **Z:\CEH-Tools\CEHv10 Module 20 Cryptography\Cryptanalysis Tools\CrypTool**, and save it to the **Desktop**.
18. Launch the **CrypTool** application.
19. The **How to Start** dialog box appears; check **Don't show this dialog again**, and click **Close**.

TASK 2

Decrypting the Data

Module 20 - Cryptography

20. The main window of **CrypTool** appears; close the **startingexample-en.txt** window.

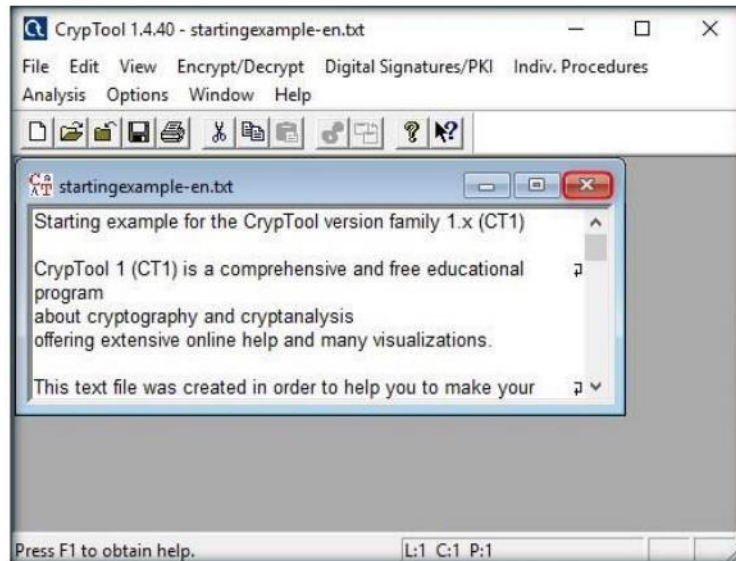


FIGURE 9.10: startingexample-en.txt window in CrypTool

21. To **decrypt** data, click **File** in the menu bar, and select **Open....**

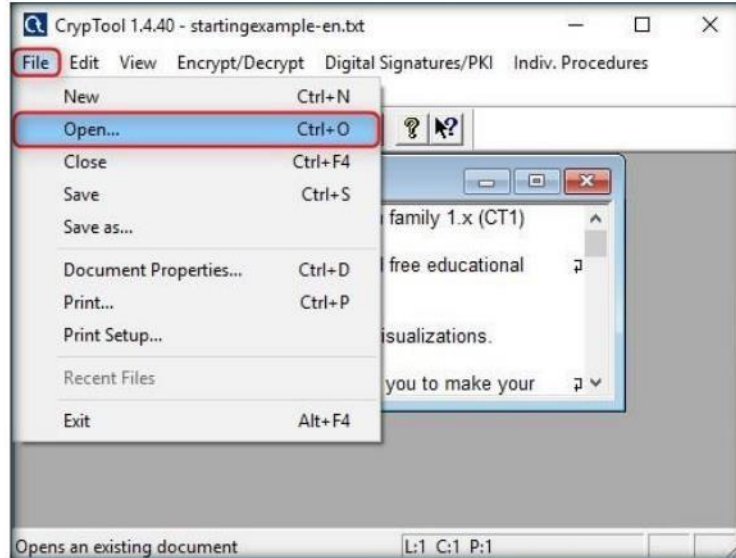


FIGURE 9.11: Opening a Crypted file

Module 20 - Cryptography

22. The **Open** dialog-box appears; select **All files** from the drop-down list, navigate to the location of the file (**Desktop**), select it, and click **Open**.

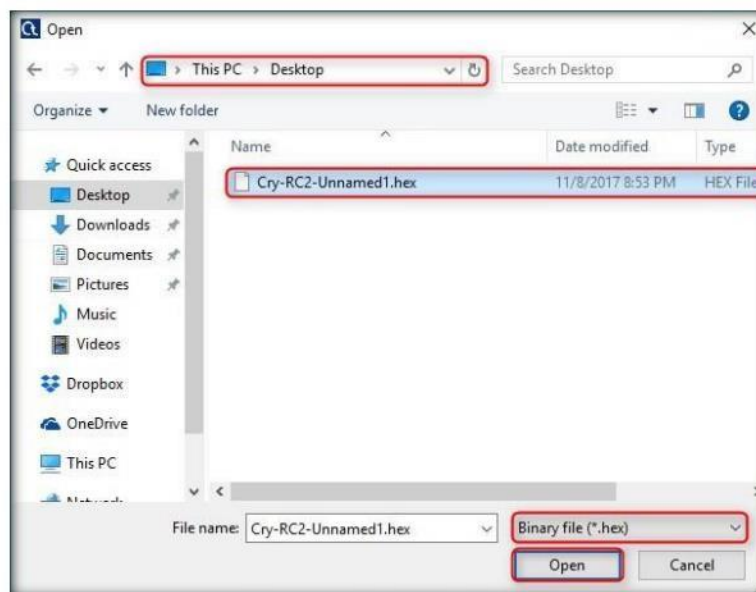


FIGURE 9.12: Opening a Crypted file

23. Select **Encrypt/Decrypt** → **Symmetric (modern)** → **RC2...** from the menu bar.

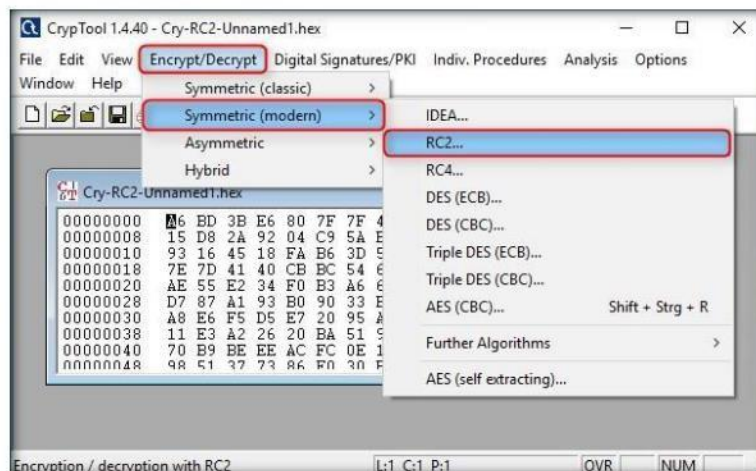


FIGURE 9.13: Select the RC2 Encryption algorithm

Module 20 - Cryptography

- The **Key Entry: RC2** dialog-box appears; select **Key length** (here, **8 bits**) from the drop-down list.
- Enter the **hexadecimal key (05)** that was used to encrypt the file, and click **Decrypt**.

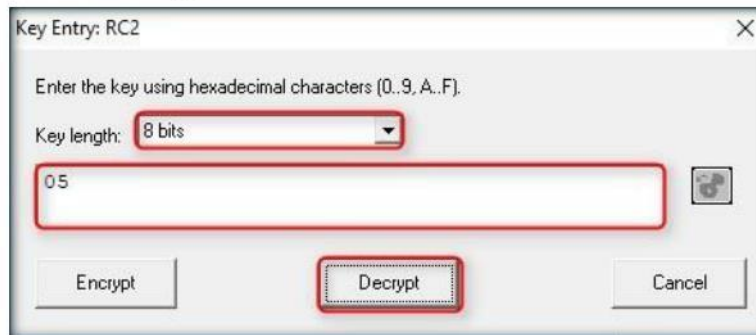


FIGURE 9.14: Decrypting the file

- The decrypted text appears, as shown in the screenshot:

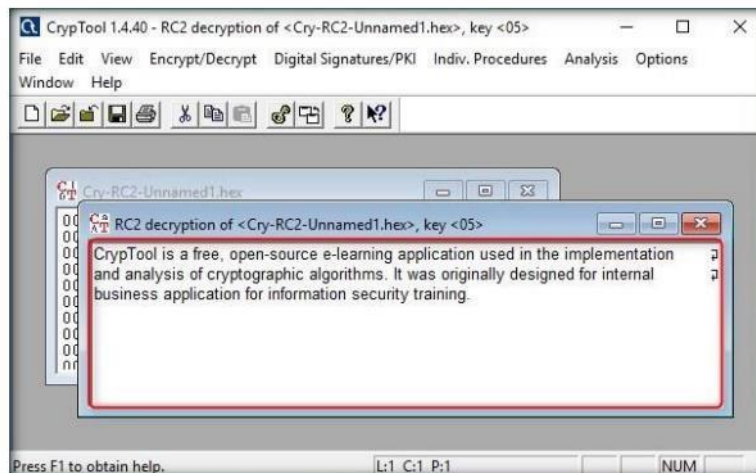


FIGURE 9.15: Decrypted the file successfully

- This way, files can be encrypted using CrypTool and shared with an individual in a secure manner, so that no one can intercept its data.

Lab Analysis

Analyze and document the results related to this lab exercise.

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

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs