**<CHN>CHAPTER THREE**

<CHT>BASIC CRYPTOGRAPHY

<COOT>Labs included in this chapter

<COOH1>

* Lab 3.1 Encrypting Files and Exploring Certificates
* Lab 3.2 Demonstrating Encryption Security
* Lab 3.3 Examining the Relationship Between EFS and NTFS Permissions
* Lab 3.4 Key-Certificate Management Policy
* Lab 3.5 Breaking the Code

<COOBT>CompTIA Security+ Exam Objectives

<COOBL>Domain Lab

<COOB>1.0 Threats, Attacks, and Vulnerabilities 3.4, 3.5

2.0 Technologies and Tools 3.1, 3.2, 3.4

4.0 Identity and Access Management 3.3, 3.4

<COOB\_LAST>6.0 Cryptography and PKI 3.1, 3.2, 3.3, 3.4, 3.5

# **<H1>Lab 3.1 Encrypting Files and Exploring Certificates**

**<H2>Objectives**

<TX1>The best way to ensure data privacy, in transit or in storage, is solid encryption built on top of a solid identification/authentication/authorization process.

<TX2>The development of Bring Your Own Device (BYOD) environments has brought a new set of vulnerabilities into the workplace. The widespread use of laptop computers has created serious data loss problems. Laptops are frequently lost or stolen, and once an attacker has physical possession of a computer, it’s a simple matter to bypass the authentication system by placing the laptop’s hard drive into a computer on which the attacker has full rights and permissions. Full disk encryption is becoming a popular method for securing data stored on laptops.

<TX2>Microsoft systems supports the Encrypting File System (EFS), which allows the encryption of folders and files and, with some editions of Windows 8 and later editions, full-drive encryption using BitLocker.

<TX2>After completing this lab, you will be able to:

* <BL>Explore the use of digital certificates in EFS
* Encrypt files from a command prompt
* Identify components of digital certificates

**<H2>Materials Required**

<TX1>This lab requires the following:

* <BL>Windows 10 with VirtualBox Software installed
* Windows 8.1 ISO

**<H2>Activity**

<FE1TX1>Estimated completion time: 1 hour

<TX1>In this activity, you will encrypt a file using the command line utility cipher.

1. <NL\_FIRST>Open your Windows 10 desktop.
2. <NL\_MID>Launch your browser and navigate to **https://technet.microsoft.com/en-us/windows/windows-8.aspx**.
3. Click the **Download the Windows 8.1 Enterprise Evaluation** under the heading “Try Windows 8.1.”
4. Click the **Evaluate Now** menu item.
5. Click **Windows 8.1 Enterprise.**
6. Click the **Sign In** button under “Windows 8.1 Enterprise.”
7. Enter your login credentials or create a new account.
8. Click the **Register to continue** button and then click **Continue.**
9. Select the **Bit version** (32 or 64) of your OS and select your language.
10. Click **Download.** Save the ISO to your hard drive.
11. Launch VirtualBox and create a new VM with the Windows 8.1 ISO. Select all the defaults except for the default hard drive space. Make the drive size at least 25GB.
12. In the virtual environment, open a Microsoft Management Console as follows: Click the **Start** button, then click the **search magnifying glass icon**. In the search window type mmc, and then click **mmc.exe**. Click **Yes** on the User Account Control Window.
13. In the Console1 window, click the **File** menu, click Add/Remove Snap-in. In the Add or Remove Snap-ins window, in the Available snap-ins box, select Certificates, click the Add button, select **Computer Account**, and then click OK. Your console should look Figure 3-1. Your list may differ depending on what’s installed on your computer.

<FGN>3-1</FGN>

<FGLB>**Figure 3-1 Certificates MMC**</FGLB>

1. In the Console1 window, expand the Certificates node in the left pane and select the Personal folder. The Object Type pane in the middle indicates that there are no items to show. Click the **File** menu, and then click Save As. In the File name box, type **<your name> Certs**, click the Desktop icon to direct the file to your desktop, and click Save. Close the **<your name> Certs** console.
2. Open a command prompt window. Navigate to the root of C: by typing cd \ and pressing Enter as many times as needed. Type cipher /? and then press Enter. Review the syntax and options used by the cipher command. Type cipher and press Enter. Your results should be similar to Figure 3-2. The “U” indicates that the items listed are unencrypted.

<FGN>3-2</FGN>

<FGLB>**Figure 3-2 Cipher command**</FGLB>

1. Type md Confidential and press Enter. Use the cipher command again to determine the encryption status of the Confidential directory. It should be unencrypted. Type copy con C:\Confidential\passwords.txt and press Enter. Type No attacker would ever guess that I use the password Pa$$word for every account. Press Enter, press Ctrl+z, and then press Enter again. Type type C:\Confidential\passwords.txt and press Enter. You should see the content of the passwords.txt file you just made.
2. Type cipher /e C:\Confidential\passwords.txt and press Enter. When the encryption process has completed, type cipher C:\Confidential and press Enter. The directory C:\Confidential is still unencrypted. Type cipher C:\Confidential\passwords.txt and press Enter. The “E” indicates that the passwords.txt file has been encrypted. Type C:\Confidential\passwords.txt and press Enter. Are you able to open and read the encrypted file. How can this be if the file is encrypted?
3. From the desktop, open <your name> Certs. If necessary, expand the Certificates node, and expand the Personal folder. It should look different from what you saw in Step 14. Click the Certificates folder inside the Personal folder. Double-click the <your name> digital certificate in the middle pane. In the General tab, determine the purpose of this certificate. Explore the details tab and examine the Serial number, the Signature Hash Algorithm, and the public key the information provided on the Details. How is this information useful? What does it tell about the certificate?
4. Close all windows (clicking Yes when asked to save console settings to <your name> Certs) and power off the virtual machine.

**<H2>Certification Objectives**

<TX1>Objectives for CompTIA Security+ Exam:

* <BL>2.3 Given a scenario, troubleshoot common security issues.
* 6.1 Compare and contrast basic concepts of cryptography.
* 6.4 Given a scenario, implement public key infrastructure.

**<H2>Review Questions**

1. <MULT>What is the purpose of a digital certificate?
2. **to store the public key of the issuer of the certificate**
3. to hold the true identity of the issuer
4. to encrypt a file
5. to encrypt the public key of the issuer of the certificate
6. <MULT>What is the importance of the Valid to option in the Details tab on the certificate?
   1. **It provides the time duration that the certificate will be acceptable.**
   2. It provides the names of the hashing algorithm.
   3. It provides the public key.
   4. It provides the name of the certificate.
7. <MULT>If the user’s file encryption key has been updated, you can use the cipher command with the \_\_\_\_\_\_\_\_\_\_\_\_\_\_ option to update files that have been encrypted with the previous key.
   1. /X
   2. /Y
   3. /R
   4. **/U**
8. <MULT>What is the cipher command used to encrypt?
   1. Files but not directories
   2. Directories but not files
   3. Directories only
   4. **Files and directories**
9. The MMC automatically gets updated with any new certificate that is created on the computer. **True** or False?

# **<H1>Lab 3.2 Demonstrating Encryption Security**

**<H2>Objectives**

<TX1>The Encrypting File System in Windows is not, strictly speaking, a file system, due to the fact that it does not track data location. (A file system is a scheme by which the operating system and the BIOS [Basic Input/Output System] track where data is located on storage media.) Instead, it uses asymmetric and symmetric encryption to increase data confidentiality. When a user encrypts a file, a File Encryption Key (FEK) is generated. This is a symmetric key; it both encrypts and decrypts the file. Once the file is encrypted, one copy of the FEK is encrypted using the user’s public key, and the encrypted FEK is attached to the file. Another copy of the FEK is encrypted using the recovery agent’s public key and is also attached to the file. Thus, only someone who has access to either the user’s private key or the recovery agent’s private key would be able to decrypt the file.

<TX2>After completing this lab, you will be able to:

* <BL>Demonstrate how the EFS protects data from unauthorized users
* Obtain information regarding the certificates that are associated with an encrypted file
* Explain how asymmetric and symmetric encryption is used by EFS
* Use the runas command to assume the credentials of different users in order to test configurations

<H2>Materials Required

<TX1>This lab requires the following:

* <BL>Windows 10 with VirtualBox installed
* Windows 8.1 ISO
* A second email account
* Successful completion of Lab 3.1

**<H2>Activity**

<FE1TX1>Estimated completion time: 10–15 minutes

<TX1>In this activity, you will test the security of the file you encrypted in Lab 3.1.

* 1. <NL\_FIRST>Open your Windows 10 desktop and Start the Windows 8.1 VM from Lab 3.1.
  2. <NL\_MID>Click the **Start** button, start typing the word **account**,and click **Add a new user** in PC settings.
  3. Click **Manage another account**.
  4. Enter the password for the master account, if necessary. Then click **Add user account** and follow the steps to add a second account using the alternative email.
  5. When finished with the new account switch to the account.
  6. Open a File Explorer window. Navigate to C:\Confidential. Notice that the passwords file is green. Double-click the passwords.txt file. What result did you get? Why?
  7. Click the **Start** button and type **cmd**. Right click the command window and choose **Run as administrator**. Type the admin password and press **Enter**. Navigate to the C:\Confidential folder. Type **passwords.txt** and hit **Enter**. Why does the file open?
  8. In the new command prompt, navigate to C:\Confidential, type passwords.txt, and press Enter. The file now opens. Note that only the program launched using the *runas* command—the cmd program in this case—recognizes the default account as having been authenticated. Any other programs running in the default user’s desktop, including the first command prompt, are only aware of the default user as having been authenticated.
  9. In the original command prompt, try the **type passwords.txt** command again. What was the result? Why?
  10. Type cipher /c and press Enter. Why can more than one user account decrypt the passwords.txt file? In order to preserve the ability to decrypt company files if something happens to the account of the user who originally encrypted the file, a recovery agent is provided. The File Encryption Key is encrypted using the user’s public key, but a recovery agent also has a key pair that can be used to access a second copy of the File Encryption Key. In a stand-alone computer or a computer in a peer-to-peer network, the local administrator is the recovery agent. In a domain environment, the first administrator in the domain is the recovery agent. On the line below, make a note of the recovery agent’s certificate thumbprint here so that you will be able to identify it later. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
  11. Close all windows and log off.

**<H2>Certification Objectives**

<TX1>Objectives for CompTIA Security+ Exam:

* <BL>2.3 Given a scenario, troubleshoot common security issues.
* 6.1 Compare and contrast basic concepts of cryptography.
* 6.4 Given a scenario, implement public key infrastructure.

**<H2>Review Questions**

1. <MULT>Which of the following statements regarding the Encrypting File System is correct?
   1. **The file is encrypted with a symmetric key.**
   2. The file is encrypted with the user’s private key.
   3. The file is encrypted with the user’s public key.
   4. The file is encrypted with the recovery agent’s public key.
2. <MULT>Which of the following statements regarding the Encrypting File System is correct? (Choose all that apply.)
3. **An encrypted file can be configured so that multiple users can decrypt it.**
4. **The recovery agent can be determined by right-clicking an encrypted file, clicking Properties, clicking the Advanced button, and then clicking the Details button.**
5. **In a domain environment, by default, the recovery agent is determined by settings in Public Key Policies.**
6. **In a stand-alone Windows 8 system, by default, the recovery agent is determined by settings in Public Key Policies.**
7. <MULT>By default, in a Windows Server 2012 environment, the recovery agent is determined by settings in \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
8. a GPO set at the site level
9. **a GPO set at the domain level**
10. a GPO set at the OU level
11. none of the above
12. <MULT>Which of the following file systems supports EFS?
13. FAT-12
14. FAT-16
15. FAT-32
16. **NTFS**
17. <TF>Both the user and the recovery agent use the same key to decrypt a file. <TFA>True</TFA> or False?

# <H1>Lab 3.3 Examining the Relationship Between EFS and NTFS Permissions

**<H2>Objectives**

<TX1>In Labs 3.1 and 3.2 you did the following:

* <BL>As a regular user, you created a folder called Confidential and a file within the folder called passwords.txt, in the root of C:, where all users have access.
* Using the Admin user account, you encrypted the file passwords.txt, so that only the user account that encrypted the file could read the file. Neither another regular user nor the first administrator in the domain, the recovery agent, was able to open the encrypted file.
* The default permissions on the user account’s folder and encrypted file were assigned to the local computer as seen in the MMC.

<TX2>Although the EFS appears to be working as intended for the default user account, it is not clear how a recovery agent would be able to recover the encrypted file should the account or encryption keys become corrupted. Access control, in the form of NTFS permissions, can be used to maintain data confidentiality, as can encryption. How are they related, and what role do NTFS permissions play in enabling the recovery agent to decrypt a file? You will find the answer to these and other questions as you work through this lab.

<TX2>After completing this lab, you will be able to:

* <BL>Take ownership of files and folders
* Modify NTFS file and folder permissions
* Explain the relationship between EFS settings and NTFS permissions

**<H2>Materials Required**

<TX1>This lab requires the following:

* <BL>Windows 10 with VirtualBox installed
* Windows 8.1 ISO
* Successful completion of Lab 3.2

**<H2>Activity**

<FE1TX1>Estimated completion time: 15–20 minutes

<TX1>In this lab, you modify file and folder ownership and NTFS permissions in order to determine the relationship between EFS settings and NTFS permissions.

* 1. <NL\_FIRST>Open your Windows 10 desktop and Start the Windows 8.1 VM from Lab 3.2
  2. <NL\_MID>Use Windows Explorer to navigate to C:\. The directory C:\Confidential was created by the default user account. Right-click C:\Confidential, select Properties, and click the Security tab to examine the permissions. Select the Authenticated Users group if necessary and note that they have every permission except Full control and special permissions. Select the Users group and note that they have only Read permissions (Read & execute, List folder contents, and Read). The System account has Full control. The default user account has no explicit permissions to the folder it created. The default permissions are inherited, as indicated by the grayed check marks.
  3. Click the Advanced button. Who is the owner of **C:\Confidential**? Note that all the accounts that have default permissions are local accounts. Click Cancel on the Advanced Security Settings for Confidential window, and click Cancel on the Confidential Properties window. Access the permissions of passwords.txt. Because the permissions have been inherited, they are no different than on the parent folder.
  4. Notice that all access is inherited from **C:\**.
  5. Select the **Users** Domain. Then Click the **View** button.
  6. Click the **Show advanced permissions**. Noticed the breakdown of the read write permissions that the file has. Click **Close**.
  7. How can this information be used to design and implement certificates on a user’s machine?
  8. What is the importance of creating the certificates at the root level?
  9. Close all windows and log off.

<H2>Certification Objectives

<TX1>Objectives for CompTIA Security+ Exam:

* <BL>4.4 Given a scenario, differentiate common account management practices.
* 6.1 Compare and contrast basic concepts of cryptography.
* 6.4 Given a scenario, implement public key infrastructure.

<H2>Review Questions

* 1. <MULT>Why is it best practice to create certificates at the root level of the user account?

1. The root level has the fewest number of privileges.
2. **The root level will allow the certificates to be inherited by the most users**.
3. The permissions on the root level are the easiest to control.
4. There is no reason to create the certificates at the root level.
   1. <MULT>Which of the following are not principal accounts for certificates in the advanced settings?
5. System
6. Administrators
7. **Domain Controllers**
8. Users
   1. <MULT>As a result of your work in this lab, it is reasonable to conclude that \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
9. **EFS security is not dependent on NTFS security**
10. EFS security is dependent on NTFS security only when the owner of an encrypted file is also listed on the file’s access control list
11. the owner of a folder always has full control of the folder
12. EFS security is effective only when implemented by domain accounts
    1. <TF>Certificates on work on NTFS file systems True or **False**?
    2. <MULT>Which of the following is not an advanced permission for a principle?
13. List folder/ read data
14. Delete
15. **Copy**
16. Take ownership

# <H1>Lab 3.4 Key-Certificate Management Policy

**<H2>Objectives**

<TX1>After completing Labs 3.1, 3.2, and 3.3, you have explored key management and encryption practices. The keys and the certificates used in the previous labs need to be governed. The lab will examine a common industry standard template for a key-certificate management policy. The research the student will perform will help identify the components and the information within the policy.

<TX2>After completing this lab, you will be able to:

* <BL>Define what key maintenance is
* Define what certificate maintenance is
* Create a key-certificate policy
* Identify needs of key and certificate maintenance

<H2>Materials Required

<TX1>This lab requires the following:

* <BL>Windows 10

**<H2>Activity**

<FE1TX1>Estimated completion time: 35–40 minutes

<TX1>In this lab, you will create a key-certificate maintenance policy.

* 1. <NL\_FIRST>Open your Windows 10 desktop.
  2. <NL\_MID>Launch your web browser and navigate to **https://www.sans.org/reading-room/whitepapers/vpns/key-certificate-management-public-key-infrastructure-technology-735**. Save the PDF to your local machine.
  3. This policy deals with Key and Certificate Management. Read through the template.
  4. Use your web browser to do some internet searches and identify best practices for key and certificate management.
  5. Using this information and the template from SANS, create a key-certificate management policy that would work for a medium size company of 250-300 employees.
  6. Assume the company is financially stable and has a modest IT department of 8 people. You should modify the content of the template to represent the company you are making the policy for.
  7. Take your time and make sure you address each section of the template.

<H2>Certification Objectives

<TX1>Objectives for CompTIA Security+ Exam:

* <BL>1.6 Explain the impact associated with types of vulnerabilities
* 2.3 Given a scenario, troubleshoot common security issues.
* 4.4 Given a scenario, differentiate common account management practices.
* 6.1 Compare and contrast basic concepts of cryptography.
* 6.4 Given a scenario, implement public key infrastructure.

**<H2>Review Questions**

* 1. <TF>Key and certificate management policies should always be grouped together True or <TFA>False</TFA>?
  2. <MULT>What is the purpose of dual or multiple key pairs?
     1. to add a level of unneeded complexity to the management process
     2. **to support distinct services for different roles**
     3. to streamline the certificate process
     4. to connect a server to a workstation
  3. <MULT>A certificate binds a \_\_\_\_\_\_\_\_\_\_ to the entity’s unique distinguished name.
     1. Private key
     2. **Public Key**
     3. Hash algorithm
     4. Public and Private keys
  4. <MULT>\_\_\_\_\_\_\_\_\_\_\_\_\_ compliments the Key backup process?
     1. Key update
     2. Key Validation
     3. **Key Recovery**
     4. Key Encryption
  5. <MULT>Which term refers to the cancellation of a certificate prior to its natural expiration?
     1. **Certificate Revocation**
     2. Certificate Authentication
     3. Certificate Expiration
     4. Certificate Archive

# **<H1>Lab 3.5 Breaking the Code**

<H2>Objectives

<TX1>All encryption algorithms can be broken. Even the algorithm considered the strongest by the U.S. government, AES (Advanced Encryption Standard), can be broken, although that might take a while. According to NIST, if you build a machine that can break 255 DES (Data Encryption Standard) keys per second, it will take that machine an estimated 172 trillion years to crack a 128-bit AES key.

<TX2>Early cryptographic algorithms were simple. The simplest schemes are stream ciphers in which one symbol of plaintext is converted to one symbol of ciphertext during encryption. The easiest type of stream ciphers to crack are monoalphabetic substitution ciphers, where only one symbol stands for only one letter. An example of a monoalphabetic substitution cipher would be a scheme in which 1 = A, 2 = B, 3 = C, and so on. The Caesar cipher took this idea a little further: A = D, B = E, C = F, and so on. Lv wkdw vr kdug?

<TX2>One way to approach substitution ciphers is to bear in mind the frequency with which letters are used in the English language. “E” is by far the most commonly used letter in common words. “T” is second, and tied for third are “A,” “O,” “I,” “S,” and “N.” Of course, if the ciphertext were made up of words, there would need to be some symbol representing a space and, particularly in longer ciphertexts, the space would be the most common symbol.

<TX2>Also useful to know when cracking a monoalphabetic substitution cipher is the frequency of two-letter combinations (digraphs) and three-letter combinations (trigraphs). “th,” “he,” “an,” “in,” “er,” “on,” “re,” and “ed” are some of the most common digraphs. “the,” “and,” “tha,” “ent,” “ion,” and “tio” are some of the most common trigraphs.

<TX2>After completing this lab, you will be able to:

* BL>Explain monoalphabetic substitution ciphers
* Decrypt a simple stream cipher

**<H2>Materials Required**

<TX1>This lab requires the following:

* <BL>Windows 10

**<H2>Activity**

<FE1TX1>Estimated completion time: 30 minutes

<TX1>In this lab, you crack a stream cipher.

* 1. <NL\_FIRST>The following is a sentence encrypted with a monoalphabetic substitution cipher. Your task is to decrypt it. Take some time to examine the cipher text. Make notes of your findings. If, after trying to crack the encryption code, you need a hint, go on to Step 2.

54:68:69:73:20:69:73:20:6e:6f:74:20:61:20:73:65:63:75:72:65:20:6d:65:73:73: 61:67:65:20:62:65:63:61:75:73:65:20:62:6f:74:68:20:68:65:78:61:64:65:63:69: 6d:61:6c:20:61:6e:64:20:41:53:43:49:49:20:61:72:65:20:77:65:6c:6c:20:6b:6e:6f:

77:6e:20:63:68:61:72:61:63:74:65:72:20:73:65:74:73:20:61:6e:64:20:74:68:65:72:

65:20:61:72:65:20:61:75:74:6f:6d:61:74:69:63:20:63:6f:6e:76:65:72:74:65:72:73:

20:6f:6e:6c:69:6e:65:2e

* 1. <NL-MID>Do not read the rest of this step until you have tried to decipher the code as instructed to do in Step 1. One thing to consider is whether the colons are delimiters—that is, do they separate symbols? It would be a reasonable assumption that they are delimiters. Take another look at the code, bearing in mind that each two-character symbol is probably a letter or a space or a punctuation mark. If you still need a hint after trying to crack the code, go on to Step 3.
  2. Do not read the rest of this step until you have worked with the hints provided in Step 2. Again, assuming that each two-character combination is a symbol, which symbol recurs most frequently? If you are ambitious, you can type the code in a Word document and use the Find function to determine how many of each two-character pairs there are. To save you the trouble, here are some results: “20” occurs 15 times, “65” occurs 7 times, “61” occurs 4 times, “6e” occurs 7 times, “63” occurs 5 times, and “6f” occurs 2 times. See if this information helps you decrypt the message. Remember that you don’t have to decipher all the symbols to deduce the pattern. If you still need help, go on to Step 4.
  3. Do not read the rest of this step until you have worked with the hints provided in Step 3. There is definitely a pattern in terms of the numbers used, particularly the first number in each pair. Most are 6s or 7s, and there are a large number of “20” pairs. It is reasonable to assume that “20” indicates a space between words. There is also a pattern in the letters used: they seem not to represent the entire alphabet. Also, once you mark the “20” pairs as being spaces, see if you can guess the small, two- and three-letter words. What are the most common two- and three-letter words? Use this information to help you solve the puzzle, but if it is still a mystery after considering these ideas, go on to Step 5.
  4. Do not read the rest of this step until you have worked with the hints provided in Step 4. The fact that most of the pairs start with 6 or 7 and that the letters only range from A to F should be a strong indication that (a) a progressive number/letter system is being used and (b) the system is likely to be a hexadecimal-ASCII conversion. Try once more to solve the problem, but go on to Step 6 if you are still not sure.
  5. Do not read the rest of this step until you have worked with the hints provided in Step 5. At this point, it is a good idea to save yourself some time. Go to http://www.dolcevie.com/js/converter.html and enter the ciphertext in the Hex: box. Then click the Hex To ASCII button. All is revealed.
  6. Repeat the process with the following hash algorithm:

54:68:69:73:20:69:73:20:6e:6f:74:20:61:20:73:65:63:75:72:65:20:6d:65:73:73: 61:67:65:20:62:65:63:61:75:73:65:20:62:6f:74:68:20:68:65:78:61:64:65:63:69: 6d:61:6c:20:61:6e:64:20:41:53:43:49:49:20:61:72:65:20:77:65:6c:6c:20:6b:6e:

6f:77:6e:20:63:68:61:72:61:63:74:65:72:20:73:65:74:73:20:61:6e:64:20:74:68:

65:72:65:20:61:72:65:20:61:75:74:6f:6d:61:74:69:63:20:63:6f:6e:76:65:72:74:

65: 72:73:20:6f:6e:6c:69:6e:65:2e

* 1. Close all windows and log off.

**<H2>Certification Objectives**

<TX1>Objectives for CompTIA Security+ Exam:

* <BL>1.6 Explain the impact associated with types of vulnerabilities.
* 6.2 Explain cryptography algorithms and their basic characteristics.

**<H2>Review Questions**

* 1. <MULT>Which of the following descriptors applies to the Caesar cipher? (Choose all that apply.)

1. Steganography
2. **Symmetric encryption**
3. Asymmetric encryption
4. **Stream cipher**
   1. <MULT>The Caesar cipher was sometimes used in an odd way. A messenger would have his head shaved and the ciphertext would be written on his head using a permanent marking method. Before the messenger was sent to deliver the message, his hair was allowed to grow until it covered up the ciphertext. This way, if captured by the enemy, the ciphertext would not be apparent. When the messenger got to his destination, his head would be shaved to reveal the coded message. Which of the following descriptors applies to this implementation of the Caesar cipher? (Choose all that apply.)
5. **Steganography**
6. **Symmetric encryption**
7. Asymmetric encryption
8. Block cipher
   1. <MULT>Which of the following is a symmetric encryption algorithm? (Choose all that apply.)
9. **AES**
10. **3DES**
11. RSA
12. SHA1
    1. <MULT>Which of the following is an asymmetric encryption algorithm? (Choose all that apply.)
13. AES
14. **Diffie-Hellman**
15. **RSA**
16. MD5
    1. <MULT>Which of the following security standards is used by the U.S. federal government to ensure the security of its information systems?
17. **FIPS**
18. SANS
19. CERT-ACID
20. ISO 17799