Name: _____ **Student No.:** _____ **Date:** __/__/____

Modern Cryptography

ADVANCED ENCRYPTION STANDARD (AES)

It is a **symmetric block cipher** that replaced the Data Encryption Standard (DES) as the encryption standard.

Originally known as the **Rijndael algorithm**, it was developed by Belgian cryptographers **Joan Daemen and Vincent Rijmen.**

It encrypts and decrypts data in fixed **128-bit blocks** using **key sizes of 128, 192, or 256 bits.**

Today, it is widely used to secure electronic data.

In this worksheet, you will:

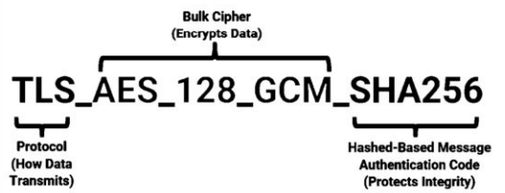
1. Explore how AES is **used in practice.**
2. Examine **secure key exchange methods.**
3. Experiment with different **modes of operation.**

How is AES used in practice to secure data?

It secures your Internet connection.

Here's how your data is encrypted as it travels between servers and your computer:

1. **Open Firefox:** If you don't have it installed, download and install it first.
2. **Visit a website:** You can check any website, but for consistency, use **mail.google.com** and **log in with your institutional email.**
3. **Check the connection security:** Click the **lock icon** 🔒 in the **Firefox address bar.**
4. **View security details:** When the **popover** appears, click **"Connection Secure"** and the **"More Information."**
5. **Open the security tab:** A **Page Info** window will appear, directing you to the **Security** tab.
6. **Find the encryption details:** Under **Technical Details**, next to **Connection Encrypted**, you will see a string like "TLS_AES_128_GCM_SHA256" —this is known as a **cipher suite**.



CIPHER SUITE

A set of algorithms, including **a cipher, a key-exchange algorithm and a hashing algorithm**, which are used together to establish a **secure TLS connection.**

7. **Learn about the cipher suite:** Visit **ciphersuite.info** and search for the specific cipher suite to view details about the **protocol, key exchange, authentication, encryption, and hash functions** used.
8. **Verify AES usage:** Confirm that **mail.google.com** is using the Advanced Encryption Standard.

✓ Questions for Analysis

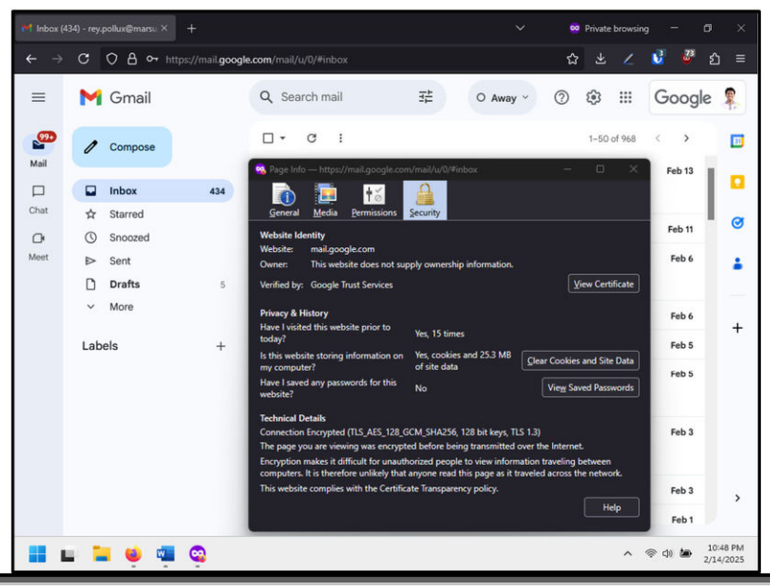
1. What **key length** is used?
   a. 128 bits
   b. 192 bits
   c. 256 bits
2. What **mode of operation** is used?
   a. Electronic Codebook (ECB)
   b. Cipher Block Chaining (CBC)
   c. Cipher Feedback (CFB)
   d. Output Feedback (OFB)
   e. Counter (CTR)

9. **Understanding AES Application:** According to the Technical Details in the Page Info window:
   *The page you are viewing was **encrypted before being transmitted** over the Internet. Encryption **makes it difficult for unauthorized people to view information traveling between computers.** It is therefore unlikely that anyone read this page as it traveled across the network.*

✓ Files to Upload

Take a **full-screen screenshot** of **Firefox** with **Google Mail open and Page Info on top**, showing the **date and time in the taskbar.**

Example:



Note: AES is used in various applications, such as **encrypting archived files**, securing **database information**, and much more. **Its applications are practically limitless!**

How can you exchange keys for AES?

AES is a **symmetric-key cipher,** which means both parties use the **same key** for encryption and decryption. They must share this key to **communicate securely.**

While **physically delivering the key** is one method, **it is impractical today.** Instead, we can use the **Diffie-Hellman Key Exchange** to **securely share the key over the Internet.**

DIFFIE-HELLMAN KEY EXCHANGE

It is the **first published public-key algorithm, defining public-key cryptography.**

Many commercial products use this key exchange technique.

The algorithm itself is **limited to the exchange of secret values.**

Let's demonstrate secure **text chat** using **AES 256-bit encryption,** with the **session key derived from the Diffie-Hellman Key Exchange.**

1. **Pair up: Find a lab partner** and decide who will act as the **server** and who will be the **client.**
2. **Connect to the same network:** Ensure **both computers** are on the **same network** to enable communication through the text chat.
3. **Download the programs: Scan the QR code** to **download** the **server and client programs** for the text chat.
4. **Run the programs:** Open `server.exe` on the **server's computer** and `client.exe` on the **client's computer.**
5. **Enter the IP address:** The **server program** will display its **IP address. Enter this address** into the **client program** to begin the **key exchange.**
6. **Initiate key exchange:** Once the connection is established, the **Diffie-Hellman Key Exchange process** will begin.
7. **Learn how the Diffie-Hellman Key Exchange works:** The programs will display the **process** of how the **shared key** is **generated** using the **private keys** of both computers.
8. **Generate the session key:** Once the **shared key** is generated, it will be used to derive a **256-bit session key** for **encrypting data** with **AES-256 encryption.**

9. **Give it a try:** Type a message in the chat and see how it's encrypted with the session key before being sent to the other computer.

Note: For this demonstration, the **prime p = 353 and the generator g = 3 are fixed**. In practice, a **2048-bit prime** (about 617 decimal digits) **or larger** is recommended for security.

✓ Questions for Analysis

1. What are the **private and public keys** generated for your **client?**

   Public key: _____ Private key: _____

2. What are the **private and public keys** generated for your **server?**

   Public key: _____ Private key: _____

3. What is the **shared secret key?**

   _____

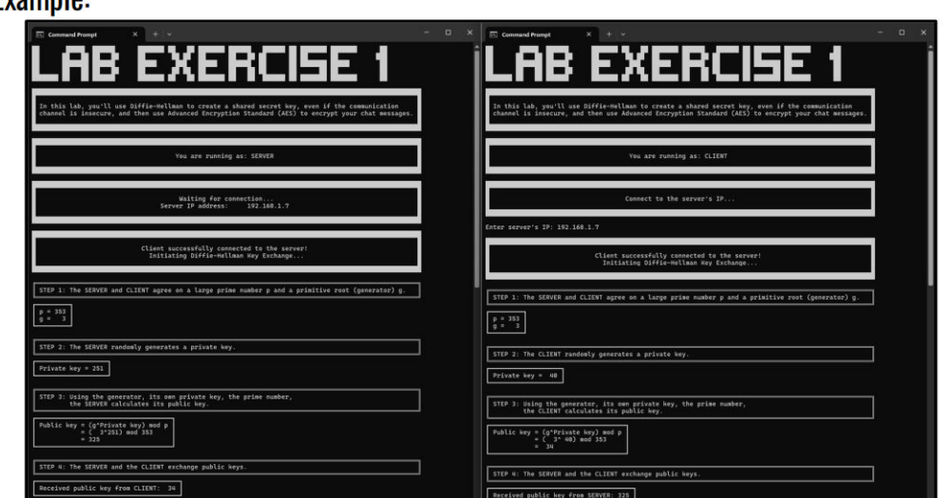
4. What is the **session key** used for encryption?

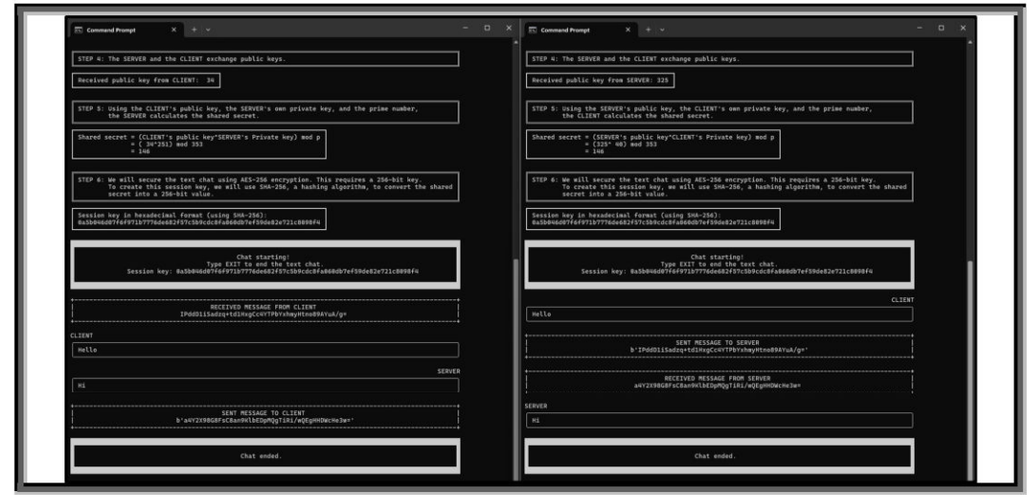
   _____

✓ Files to Upload

Take a **full-screen screenshot** of both the **server and client terminals,** showing the **complete** Diffie-Hellman Key Exchange process.

Example:





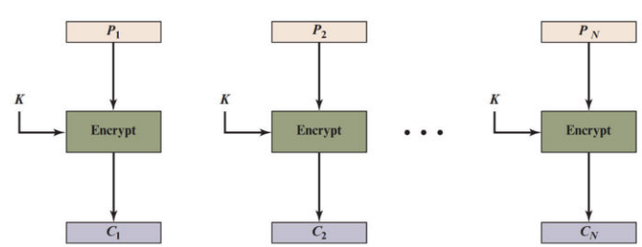
How can the security of AES be improved?

AES is a **block cipher,** which means it breaks the plaintext into **equal-sized blocks** for encryption using a **common key.**

However, **when multiple blocks are encrypted using the same key, a number of security issues arise.** To address these issues, NIST defines **five modes of operation** to enhance the **effectiveness** of a **symmetric block cipher.**

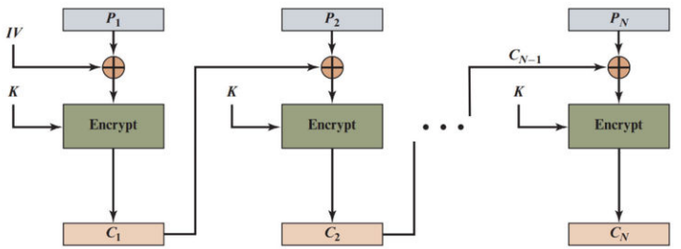
The table summarizes these modes.

Mode	Description	Typical Application
Electronic Codebook (ECB)	Each block of plaintext bits is encoded independently using the same key.	• Secure transmission of single values (e.g., an encryption key)
Cipher Block Chaining (CBC)	The input to the encryption algorithm is the XOR of the next block of plaintext and the preceding block of ciphertext.	• General-purpose block-oriented transmission • Authentication
Cipher Feedback (CFB)	Input is processed s bits at a time. Preceding ciphertext is used as input to the encryption algorithm to produce pseudorandom output, which is XORed with plaintext to produce next unit of ciphertext.	• General-purpose stream-oriented transmission • Authentication
Output Feedback (OFB)	Similar to CFB, except that the input to the encryption algorithm is the preceding encryption output, and full blocks are used.	• Stream-oriented transmission over noisy channel (e.g., satellite communication)
Counter (CTR)	Each block of plaintext is XORed with an encrypted counter. The counter is incremented for each subsequent block.	• General-purpose block-oriented transmission • Useful for high-speed requirements



The **simplest** mode is the **electronic codebook (ECB).** However, if a **plaintext block repeats,** ECB **produces the same ciphertext** each time.

To fix this, we need a technique that **generates different ciphertext** for **repeated plaintext blocks. Cipher block chaining (CBC)** mode does just that by inputting the **XOR of the current plaintext block and the previous ciphertext block** into the encryption algorithm, all while using the **same key** for each block.



Let's demonstrate the **differences** between the two AES encryption modes and **compare their security using an image.**

1. **Download the programs: Scan the QR code** to **download** the **programs** encrypt images using **AES** with **ECB** and **CBC** modes.
2. **Download the image:** Download the image **assigned** to you. To understand the process, we will focus on **encrypting images with objects on a white background.**
   Note: Ensure that both the **programs** and the **image** are in the **same directory.**
3. **Open a terminal:** Use the **command prompt** and **navigate** to the **directory** containing the **programs** and the **image** using the `cd` command.
4. **Encrypt using AES-ECB:** To encrypt the image in ECB mode, run:
   `aes-ecb.exe name_of_image.jpg`
5. **Encrypt using AES-CBC:** To encrypt the image in CBC mode, run:
   `aes-cbc.exe name_of_image.jpg`
6. **Open the encrypted images:** After running the programs, open the **resulting encrypted bitmap images** (e.g., `image-aes-ecb.bmp` and `image-aes-cbc.bmp`).
7. **Compare the results:** Observe how the encryption **differs** between the **two modes.**

✓ Questions for Analysis

1. How do the two images encrypted with **AES** in **ECB** and **CBC** modes **differ**?
   _____
   _____
   _____
2. Which **mode of operation** is **more secure? ECB or CBC?** _____

✓ Files to Upload

Upload the **encrypted images** with the **mode of operation in the file name.**

Resources:

Cryptography and Network Security, Stallings
https://www.ssl.com/guide/tls-standards-compliance/
https://www.britannica.com/topic/cryptology/Secret-sharing
https://sectigostore.com/blog/what-is-an-ssl-tls-cipher-suite/
https://www.practicalnetworking.net/series/cryptography/diffie-hellman/