EECE_5699 Computer Hardware and System Architecture

Homework 1

Sakthivel P. Sivakumar

NUID: 002312294

Lab Module 1: Performance of Ciphers (30 Pts)

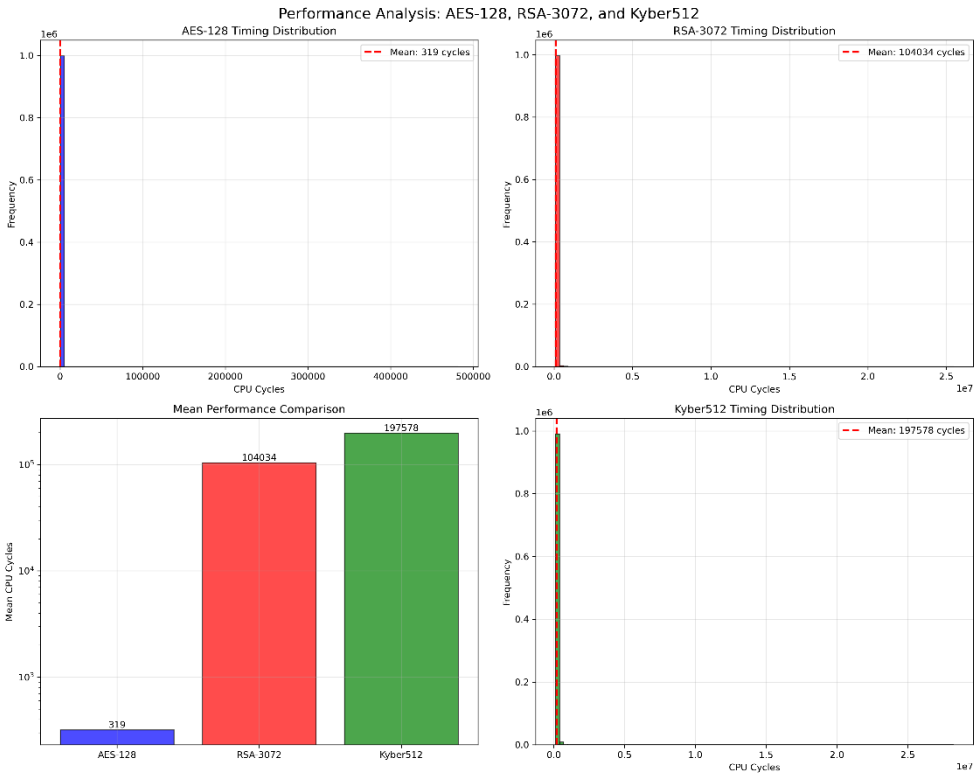
All the 3 algorithms (AES, RSA and Kyber) Collected exactly 1,000,000 timing measurements with 128-bit security level.

1. AES-128 (Symmetric encryption)
2. RSA-3072 (Asymmetric encryption)
3. Kyber512 (post-quantum key encapsulation mechanism)

Based on 1,000,000 timing samples for each algorithm:

A	B	C	D
Algorithm	**Mean (CPU Cycles)**	**Median (CPU Cycles)**	**Std Dev (CPU Cycles)**
AES-128	318.51	288	831.01
RSA-3072	104034.35	95082	75189.84
Kyber512	197577.95	186883	80188.11

Performance Comparison Chart for all the 3 Algorithms



Key Observations:

- From the above observation we can clearly see that **AES is the fastest among the 3**, which is expected for symmetric vs asymmetric cryptography.
- RSA shows **better performance** than the post-quantum alternative Kyber512

Question 1: First compare the performance of RSA, AES-128, and Kyber512 (on the same size of plaintext- 16 bytes). Choose the appropriate key size for RSA to achieve the same security level as AES 128. How much slower is RSA/Kyber than AES? With this implementation cost, discuss what scenarios RSA, AES, and Kyber are mainly used for, respectively. Why would someone want to use Kyber rather than RSA?

Ans:

Performance Comparison:

- RSA-3072: **326.6 times** slower than AES-128
- Kyber512: **620.3 times** slower than AES-128

Security Level Equivalence:

- **AES-128**: 128-bit security level
- **RSA-3072**: 128-bit equivalent security level
- **Kyber512**: 128-bit equivalent security level

Use Case Analysis:

1. AES-128 Use Cases:

- Bulk data encryption (files, network traffic, disk encryption)
- Real-time communications (VPN, messaging)
- High-throughput applications
- Battery-powered devices where efficiency matters

2. **RSA-3072 Use Cases:**

- Digital signatures and authentication
- Key exchange and agreement protocols
- PKI infrastructure (certificates, code signing)
- Secure email, TLS/SSL handshakes

3. Kyber512 Use Cases:

- Post-quantum secure key exchange
- Futureproofing against quantum computers
- Long-term data protection (25+ year lifespans)
- Government and military applications requiring quantum resistance

Why quantum over RSA:

RSA will become completely broken when large-scale quantum computers become available (estimated 10-30 years). Kyber512 remains secure against both classical and quantum attacks. NIST has standardized Kyber (now called ML-KEM) as the primary post-quantum key encapsulation mechanism, ensuring future interoperability.

While AES-128 dominates in performance, the choice between RSA and Kyber depends on future security requirements. Organizations planning for long-term data protection should begin transitioning to post-quantum algorithms like Kyber512 despite the current performance penalty

Lab Module 2: AES Encryption Mode (20 pt)

In this module we are going to compare AES-128 encryption in ECB (Electronic Codebook) and CBC (Cipher Block Chaining) modes by encrypting the same penguin image and analysing the security implications of each approach.

Image Processing Steps:

1. Take the Original Image 'penguin.PPM'
2. Extracting the **PPM header (15 bytes)** containing the format information
3. Separating the image data (586,080 bytes) for encryption
4. Applying AES in both ECB and CBC modes
5. Combining header with encrypted data to create viewable images

File Size Analysis:

	A	B	C
1	**File**	**Size (bytes)**	**Change**
2	**Original**	586095	-
3	**ECB Encrypted**	586111	+16 bytes
4	**CBC Encrypted**	586111	+16 bytes

Both modes added exactly 16 bytes due to PKCS#7 padding requirements.

Next, we will look into the characteristics of each mode

ECB Mode:

- Identical plaintext blocks produce identical ciphertext blocks, repetitive patterns are clearly visible (` 2d 73 1c 68...` appears multiple times),
- Original image patterns may remain partially visible and susceptible to pattern analysis and frequency attacks.

CBC Mode:

- Each ciphertext block depends on the previous block, No visible repetitive patterns in the encrypted data
- Complete randomization of image data, Resistant to pattern analysis attacks

Statistical Analysis:

	A	B	C	D
1	**Metric**	**Original**	**ECB**	**CBC**
2	Mean Pixel Value	62.24	120.92	127.32
3	Standard Deviation	101.39	71.16	73.98

These stats show both encrypted versions have more uniform distributions, but **CBC provides better randomization**.

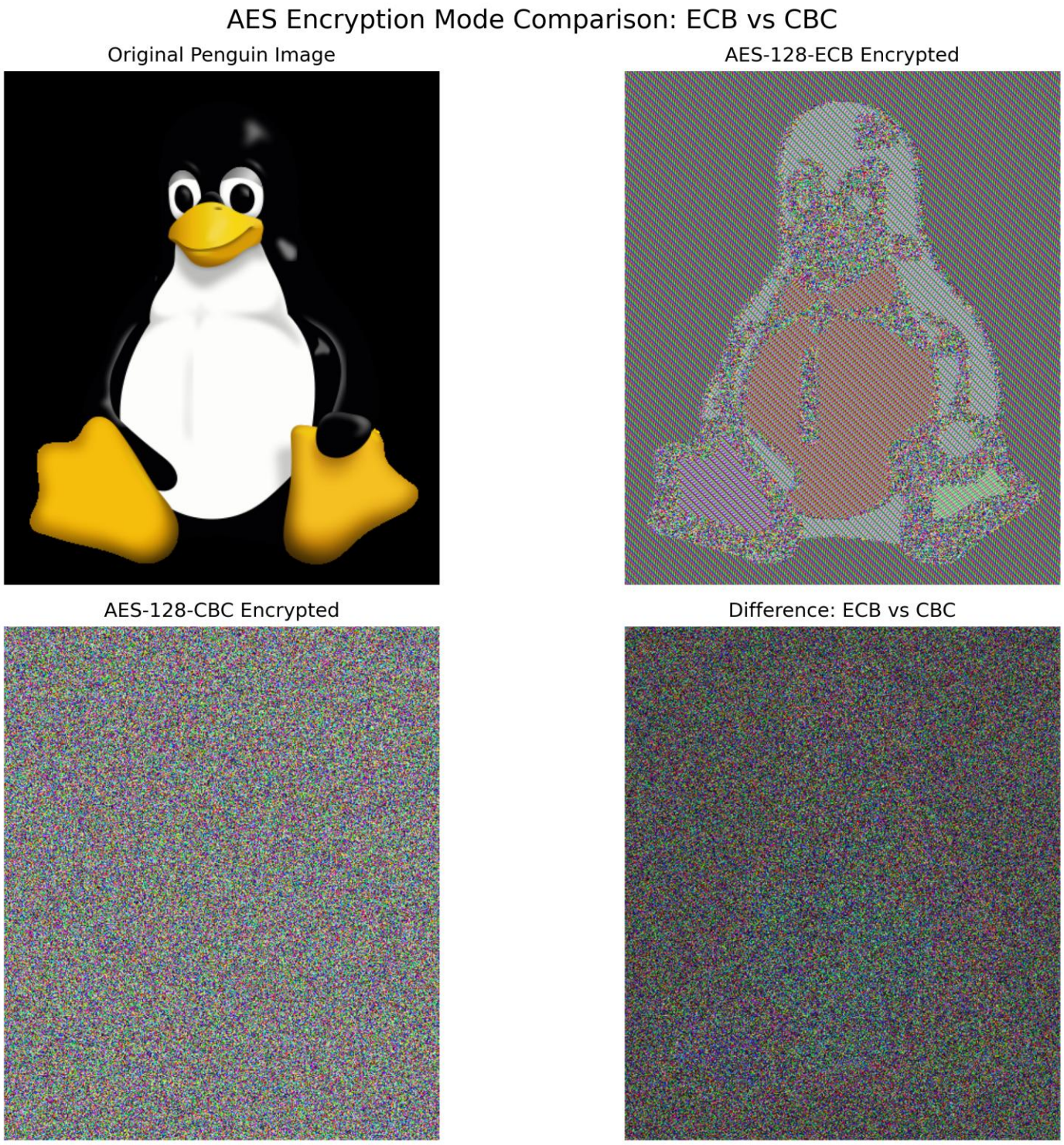
Question 2: Compare these two encrypted images and comment on the security. What is the downside of CBC mode in terms of performance? Suggest one operation mode you think is the best and give your reason.

By comparing these two encrypted images, we can see that ECB mode creates Identical ciphertext block, potentially revealing image structures. Same plaintext always produces same ciphertext (no randomness)

Whereas in CBC, Chaining effect ensures **identical plaintext blocks produce different ciphertext blocks**, Same plaintext produces different ciphertext with different IVs, thus creating strong resistance to frequency attacks and pattern analysis.

From the above comparison, the recommended operation mode is “**CBC Mode**”

When Choosing encryption modes, **security should be the primary consideration**, **making CBC mode the clear winner over ECB** for general-purpose encryption needs.



Lab Module 3: Secure Communication (50 pt)

This module contains client implementations for both RSA and Kyber (post-quantum) cryptographic protocols

Results obtained:

1. RSA Communication (Port 12000), the secret obtained is: "**Welcome to 5699**!" and the algorithm used are: RSA-4096 + AES-128
2. Kyber Communication (Port 13000), the secret obtained is: "**KyberPostQuantum** and the algorithm used are Kyber512 KEM + AES-128

Both clients successfully demonstrate secure communication using classical and post-quantum cryptographic protocols.