CSS454: Computer and Communication Security

Midterm Mock Exam

curated by The Peanuts

Name . . Google . . . . . ID . . 6622772422 . . . Section . . . Only 1 section . . Seat No . . . 1 . . . . . .

Conditions: Closed Book

Directions:

1. This exam contains 24 pages (including this one). If your exam has fewer, your paper has been subject to a *denial-of-service attack*.
2. Write your name and student ID clearly at the top.
3. This is a closed-book exam. No calculators, no dictionaries, no cheat sheets, and absolutely no neighbor's memory. Attempting to read someone else's answer is a *man-in-the-middle attack* and will be treated as such.
4. Answers must be written in English. Responses in Caesar-shifted Thai, Base64, or hexadecimal will not be graded.
5. Do not share answers with other students. Key reuse is a known vulnerability. Your answer is your private key, keep it that way.

For solution, ***click here****.*

Part I: True / False (10 points, 1 point each)

For each statement below, write **TRUE** or **FALSE** in the space provided.

1.1 True The CIA triad in information security stands for Confidentiality, Integrity, and Availability.

1.2 False Caesar Cipher is ~~resistant~~ to frequency analysis attacks because it uses ~~multiple~~ substitution alphabets.

1.3 True In Cipher Block Chaining (CBC) mode, each plaintext block is XOR-ed with the previous ciphertext block before being encrypted.

1.4 False Electronic Codebook (ECB) mode requires an Initialization Vector (IV) to function correctly.

1.5 False In RSA, a message is encrypted using the recipient's private key and decrypted using their public key.

1.6 True Elliptic Curve Cryptography (ECC) can achieve equivalent security to RSA with significantly smaller key sizes.

1.7 False In CP-ABE, the access policy is embedded inside the user's secret key SK_U rather than inside the ciphertext.

1.8 False OAuth 2.0 is primarily an ~~authentication~~ ~~protocol~~ that verifies the identity of a user.

1.9 True Among common biometric methods, iris scanning generally provides higher accuracy (lower false match rate) than fingerprint scanning.

1.10 True A digital certificate binds a public key to an identity and is digitally signed by a trusted Certificate Authority (CA).

Part II: Fill in the Blank (16 points, 2 points each)

Fill in each blank with the most appropriate term, value, or expression. Partial credit may be given for partially correct answers.

1. The general Caesar Cipher encryption formula is $C_i = (P_i + K) \bmod$ __26__, where all values represent character positions in the English alphabet.

2. In RSA, the public key is represented as (e, n), while the private key is (__d__, n). The security of RSA relies on the hardness of __integer factorization__.

3. For n users communicating using symmetric encryption, the total number of keys required is __$\frac{n(n-1)}{2}$__ (Formula). For $n = 10$ users, this equals __45__ keys.

4. In AES-CBC mode, the very first plaintext block is XOR-ed with __initialization vector (IV)__ before encryption. This value must be __unpredictable / public__ (secret / unpredictable / public) but should never be reused with the same key.

5. In CP-ABE, a user can successfully decrypt a ciphertext if and only if their __attribute__ set satisfies the __access policy (policy tree)__ embedded in the ciphertext.

6. OpenID Connect (OIDC) is an __authentication__ protocol built on top of __OAuth 2.0__. The user identity information is returned in a __JWT__.

7. In biometrics, the __False Match__ Rate (also known as False Accept Rate) occurs when two biometric samples from different individuals are incorrectly judged to belong to the same person.

8. In hybrid encryption, __AES__ is used to encrypt the actual data (fast), while __RSA__ is used to encrypt the session key (secure key exchange). This combination is often called __hybrid encryption__.

What?
555

- **FMR (False Match Rate):** System accepts an impostor (false positive)
- **FNMR (False Non-Match Rate):** System rejects a legitimate user (false negative)

Part III: Multiple Choice (30 points, 2 points each)

*Choose the **best** answer for each question. Circle or write the letter of your choice clearly.*

1. Which of the following is an example of a passive attack?

a) Modifying a transmitted message in transit

b) Replaying a captured authentication packet

c) Eavesdropping on network traffic without altering it

d) Launching a Denial-of-Service attack on a server

active

2. An organization disconnects its servers from the Internet to improve confidentiality. Which CIA property does this decision most directly sacrifice?

a) Confidentiality ↑ improve ความลับ (no attack)

b) Integrity ความถูกต้อง (no updates)

c) Availability ↓ legimate user เข้าถึงไม่ได้

d) Authentication

3. The Vigenère cipher was designed to overcome a known weakness of the Caesar cipher. Which weakness does it specifically address?

a) The use of a symmetric key

b) Vulnerability to frequency analysis on single-character substitution

c) Inability to encrypt numbers and symbols

d) The key must be as long as the message

4. Which block cipher mode of operation produces the same ciphertext every time the same plaintext block is encrypted with the same key?

a) Cipher Block Chaining (CBC)

b) Electronic Codebook (ECB)

c) Counter (CTR)

d) Output Feedback (OFB)

identical plaintext blocks produce different ciphertexts.

5. AES-GCM is classified as an Authenticated Encryption with Associated Data (AEAD) scheme. What happens if the authentication tag verification fails during decryption?

a) The plaintext is released with a warning flag

b) Only the corrupted blocks are discarded; the rest of the plaintext is released

c) Decryption is rejected and no plaintext is released

d) The ciphertext is automatically re-encrypted with a new key

Decryption process:

1. Decrypt ciphertext to produce candidate plaintext
2. Verify authentication tag
3. **If tag verification fails:** Reject entire message, output nothing
4. **If tag verification succeeds:** Release plaintext

6. Which of the following cipher modes converts a block cipher into a stream cipher by encrypting successive counter values?

a) Electronic Codebook (ECB)

b) Cipher Block Chaining (CBC)

c) Counter (CTR)

d) Cipher Feedback (CFB)

Explanation: CTR mode encrypts a counter value and XORs it with plaintext:

$$C_i = P_i \oplus E_K(\text{nonce}\|\text{counter}_i)$$

This effectively turns a block cipher into a stream cipher:

- No chaining (blocks are independent)
- Parallelizable encryption/decryption
- Random access (can decrypt any block without processing previous blocks)

CFB and OFB also create stream ciphers, but use feedback rather than counters.

7. **Given RSA parameters $p = 3$, $q = 11$, $e = 3$, compute Euler's totient $\phi(n)$.**

$\phi(n) = (p-1)(q-1) = (2)(10) = 20$

a) 30

b) 33

c) 24

d) 20

8. **In hybrid encryption, why is RSA typically used to encrypt the session key rather than the entire message?**

a) RSA produces smaller ciphertexts than AES

b) RSA is faster than AES for large data

c) RSA is computationally expensive; encrypting a small key is more practical

d) The session key must be stored in the RSA modulus n

9. **The key advantage of ECC over RSA at equivalent security levels is:**

a) ECC is based on integer factorization, which is harder than discrete logarithm

b) ECC requires significantly smaller key sizes, reducing computational overhead

c) ECC generates key pairs faster but verifies signatures more slowly

d) ECC does not use public-private key pairs

10. What security services does a digital signature primarily provide?

a) Confidentiality and availability

b) Confidentiality and integrity

c) Authentication and non-repudiation

d) Integrity and availability

11. A digital signature is created using:

a) The sender's public key

b) The recipient's public key

c) The sender's private key

d) The recipient's private key

12. In CP-ABE, which algorithm is called by the data owner to encrypt data and embed the access policy?

a) `Setup`(1^λ) AA (Attribute Authority)

b) `KeyGen`(MSK, S) AA issue User

c) `Encrypt`(PK, M, T) Data owner

d) `Decrypt`(CT, SK_U) User

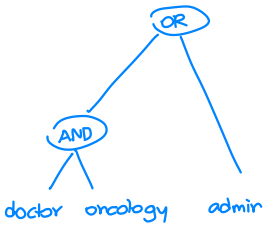
13. Consider the CP-ABE access policy `(doctor AND oncology) OR admin`. A user with attributes {`doctor, cardiology, manager`} attempts to decrypt. What is the result?

a) Decryption succeeds because the user is a doctor

b) Decryption succeeds because the user has the manager attribute

c) Decryption fails because the user does not satisfy the policy

d) Decryption succeeds using the collusion of manager and doctor attributes



14. What is the primary difference between OAuth 2.0 and OpenID Connect (OIDC)?

a) OAuth 2.0 is for authentication; OIDC is for authorization

b) OAuth 2.0 provides an ID token; OIDC provides an access token

c) OAuth 2.0 is for authorization (resource access); OIDC adds authentication (identity verification)

d) OAuth 2.0 uses JWT; OIDC uses XML-based SAML assertions

15. In OpenID Connect, what is the purpose of the `nonce` parameter included in the authorization request?

a) To identify the client application uniquely across requests

b) To bind the ID token to the client session and prevent replay attacks

c) To indicate the desired scopes for the access token

d) To encode the user's preferred language in the response

1. Client generates random nonce + include in auth request
2. Auth server includes the same nonce in ID token
3. Client verifies that returned nonce matches the one it sent.

Part IV: Short Answer (40 points)

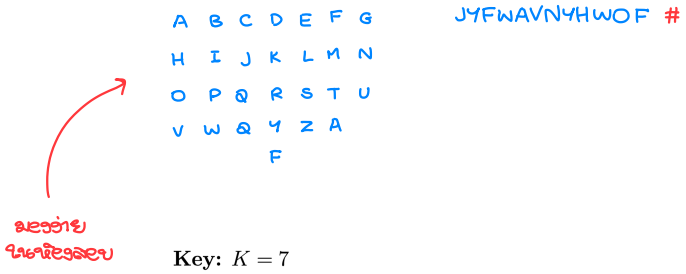
Answer each question clearly. Show all work for computation questions. Partial credit is awarded for correct reasoning even with a wrong final answer.

Question 1 (4 points)

Encrypt the plaintext **"CRYPTOGRAPHY"** using Caesar Cipher with key $K = 7$. Use the standard mapping $A = 0, B = 1, \ldots, Z = 25$, and the encryption formula:

$$C_i = (P_i + K) \bmod 26.$$

Show your computation for each character step by step and write the final ciphertext.



Key: $K = 7$

Formula: $C_i = (P_i + 7) \bmod 26$

Mapping: A=0, B=1, C=2, ..., Z=25

Step-by-step encryption:

Letter	Position (P_i)	$P_i + 7$	$(P_i + 7) \bmod 26$	Ciphertext
C	2	9	9	J
R	17	24	24	Y
Y	24	31	5	F
P	15	22	22	W
T	19	26	0	A
O	14	21	21	V
G	6	13	13	N
R	17	24	24	Y
A	0	7	7	H
P	15	22	22	W
H	7	14	14	O
Y	24	31	5	F

Question 2

(6 points)

Perform RSA key generation given the following parameters:

$$p = 3, \quad q = 11, \quad e = 3.$$

1. Compute n and $\phi(n)$.

$n = pq$
$= 3(11)$
$n = 33$

$\phi(n) = (p-1)(q-1)$
$= (2)(10)$
$= 20$

2. Verify that $\gcd(e, \phi(n)) = 1$ and find the private key exponent d such that $de \equiv 1 \pmod{\phi(n)}$.

อาจจะยากไปนะ หา d เนี่ย สุ่มเลขเอาก็ได้

ถ้าจำได้ gcd(a, b) = gcd(b, a % b)
if b == 0, return a

$\gcd(3, 20) = \gcd(3, 20)$
$= \gcd(20, 3 \% 20)$
$= \gcd(20, 3)$
$= \gcd(3, 20 \% 3)$
$= \gcd(3, 2)$
$= \gcd(2, 3 \% 2)$
$= \gcd(2, 1)$
$= \gcd(1, 2 \% 1)$
$= \gcd(1, 1) = \gcd(1, 1 \% 1) = \gcd(1, 0) = 1$ #

$d = 7$

$de \bmod \phi(n) = 1$
$(7)(3) \bmod 20 = 1$
$21 \bmod 20 = 1$
$1 = 1$ #

$\therefore$ Keys:
Public Key: $(e, n) = (3, 33)$
Private Key: $(d, n) = (7, 33)$

3. Using your public key (e, n), encrypt the message $M = 2$. Then verify by decrypting the resulting ciphertext back to M.

Encrypt:

$C = M^e \bmod n$
$= 2^3 \bmod 33$
$= 8 \bmod 33$
$\therefore C = 8$

Decrypt:

$M = C^d \bmod n$
$= 8^7 \bmod 33$
$= 2097152 \bmod 33$
$\therefore M = 2$ #

Question 3 (4 points)

Compute the following modular arithmetic expressions. Show your work; a bare answer receives no credit.

1. $(-17) \bmod 5$

$-17 = (-4)(5) + 3$

$\therefore 3$ #

2. $(-7) \bmod 3$

$-7 = (-3)(3) + 2$

$\therefore 2$ #

3. Find x such that $7x \equiv 1 \pmod{10}$. (Hint: try small values or use the extended Euclidean concept.)

Try small values:

$7 \times 1 = 7 \not\equiv 1 \pmod{10}$

$7 \times 2 = 14 \not\equiv 1 \pmod{10}$

$7 \times 3 = 21 \equiv 1 \pmod{10}$

$\therefore x = 3$ #

4. $17^2 \bmod 11$

$289 \bmod 11 = 3$ #

Question 4 (5 points)

Do you agree that "AES symmetric encryption alone is sufficient to secure data transmission between Alice and Bob over an untrusted network?" Support your answer by discussing:

- What AES provides (and does not provide),
- The key distribution problem,
- How hybrid encryption addresses this limitation.

I disagree.

What AES provides: Confidentiality (prevents eavesdroppers from read plaintext),
Strong encryption (128, 256: infeasible to break),
Fast!?

What AES does NOT provide: Key distribution (how do Alice and Bob securely share the symmetric key?)
Authentication (how does Bob know that message came from Alice?)
Integrity ??

The key distribution problem:

In symmetric encryption, both parties must possess the same secret key.

- **Problem:** How do Alice and Bob establish a shared key over an untrusted network?
- **Chicken-and-egg:** They need encryption to protect the key, but they need the key to enable encryption!
- **Insecure solutions:**
  - Sending key in plaintext → Eavesdropper intercepts it
  - Pre-sharing keys offline → Not scalable, keys can be compromised

How hybrid encryption solves this:

Hybrid encryption combines AES (symmetric) with RSA (asymmetric):

1. Bob generates RSA key pair: (PK_{Bob}, SK_{Bob})
2. Bob publishes PK_{Bob} (can be sent in the clear)
3. Alice generates random AES session key: k_{AES}
4. Alice encrypts data: $CT_M = \text{Enc}_{AES}(k_{AES}, M)$
5. Alice encrypts key: $CT_K = \text{Enc}_{RSA}(PK_{Bob}, k_{AES})$
6. Alice sends (CT_K, CT_M) over untrusted network
7. Bob decrypts key: $k_{AES} = \text{Dec}_{RSA}(SK_{Bob}, CT_K)$
8. Bob decrypts data: $M = \text{Dec}_{AES}(k_{AES}, CT_M)$

Benefits:

- **Solves key distribution:** RSA securely exchanges the AES key
- **Performance:** RSA only encrypts small key; AES handles bulk data
- **Scalability:** No pre-shared secrets needed

Still needed: Authentication (digital signatures) and integrity (HMAC or AEAD modes like AES-GCM).

Conclusion:

AES alone cannot secure data transmission because:

1. The symmetric key cannot be securely distributed over an untrusted network
2. Even with a shared key, AES provides no authentication or integrity guarantees
3. A complete solution requires hybrid encryption + digital signatures + AEAD

Question 5 (6 points)

Explain the role of each of the following CP-ABE algorithms. For each algorithm, state: (a) who calls it, (b) its inputs, and (c) its output.

1. `Setup`(1^λ) Run once

   a) AA (Attribute Authority) / Trusted Third Party

   b) 1^λ: security param (e.g. $\lambda = 128$ bit)

   c) MSK, PK
   - MSK → kept secret
   - PK → shared to all users

2. `KeyGen`$(MSK,\ S)$

   a) AA (Attribute Authority)

   b) MSK, S (Set of attribute assigned to the user)

   c) SK_U

3. `Decrypt`$(CT,\ SK_U)$

   a) User

   b) CT, SK_U

   c) M or $\perp$

Question 6 (5 points)

A company's authentication system uses biometric recognition at its data centre entrance. The system is currently configured with a threshold that results in:

$$\text{FMR} = 0.1\% \quad \text{and} \quad \text{FNMR} = 8\%.$$

1. Define False Match Rate (FMR) and False Non-Match Rate (FNMR) in plain terms.

False Match Rate: The probability that the biometric system incorrectly accepts an imposter
↳ False Positive

False Non-match Rate: The probability that the biometric system incorrectly rejects a legitimate user
↳ False Negative

2. The security officer wants to reduce FMR to near zero. What trade-off will this cause? Explain the relationship between FMR and FNMR.

Reducing FMR (making matching stricter) will increase FNMR (more false rejections of legitimate users

3. Suggest one biometric modality that provides inherently higher accuracy than fingerprint, and briefly justify your choice.

Answer: Iris scanning (or Retina scanning)

Justification:

Iris scanning provides significantly higher accuracy than fingerprints:

Biometric	Typical FMR	Unique Features
Iris	10^{-6} to 10^{-8}	200+ degrees of freedom
Fingerprint	10^{-3} to 10^{-4}	30–40 minutiae points

Why iris is more accurate:

- **More distinctive features:** Patterns, rifts, colors, rings, coronas, furrows
- **Stability:** Iris pattern remains unchanged from childhood through adulthood
- **Protected location:** Less susceptible to damage than fingerprints
- **Difficult to forge:** Requires live eye detection

Question 7 (5 points)

Consider the CBC and CTR modes of AES operation.

1. A 128-bit block of plaintext $P_1 =$ `0xABCD...` is to be encrypted. In CBC mode, what additional value must be XOR-ed with P_1 before the block cipher is applied? What is this value for the first block?

In CBC mode, the first plaintext block P_1 is XOR-ed with the Initialization Vector (IV) before encryption:

$C_1 = E_k(P_1 \oplus IV)$

- Must be unpredictable (random)
- Can be sent in the clear (public)
- Must never be reused with the same key

2. In CBC, if ciphertext block C_2 is corrupted during transmission, which plaintext blocks will be affected upon decryption? Explain why.

C_2'

Block 2 (P_2):

$P_2 = D_k(C_2) \oplus C_1$

$= D_k(C_2') \oplus C_1$

$\Rightarrow$ Random plaintext (unrecoverable)

Block 3 (P_3):

$P_3 = D_k(C_3) \oplus C_2$

$= D_k(C_3) \oplus C_2'$

$\Rightarrow$ Error in the same bit positions as corruption in C_2

Block 4 and beyond: Unaffected

3. Why is CTR mode often preferred over CBC for random access decryption (e.g., seeking to a specific position in an encrypted file)?

1. Block independence:

In CTR mode, each ciphertext block is computed independently

$C_i = P_i \oplus E_k(\text{nonce} \| \text{counter}_i)$

To decrypt block i (e.g., block 99), we need:

- Ciphertext block C_i
- Encryption key K
- Counter value i

2. CBC requires sequential processing:

In CBC, each plaintext block depends on the previous ciphertext block

$P_i = D_k(C_i) \oplus C_{i-1}$

To decrypt block P_{100}, we must:

1. Decrypt C_{100} to get intermediate value
2. XOR with C_{99} to get P_{100}
3. But to get C_{99}, we might need to process from the beginning

3. Parallelization

CTR allows parallel decryption of multiple blocks simultaneously, while CBC must process blocks sequentially

Question 8 (5 points)

Compare OAuth 2.0 and OpenID Connect (OIDC) by answering the following:

1. A mobile fitness application wants to read a user's step count from Google Fit and display the user's name and profile picture after login. Which protocol(s) should the developer use for each requirement? Justify your answer.

Protocol: OAuth 2.0 (Authorization)

we need access to Google Fit API (protected resource)

OAuth 2.0 provides access token for API calls

User grants permission

Token grants limited scope (e.g. scope = fitness.activity.read)

Protocol: OpenID Connect (OIDC) (authentication)

Need to verify user identity and retrieve profile information

OIDC provides ID token (JWT) contain user info.

2. In the OIDC Authorization Code Flow, what token is issued specifically to confirm the user's identity, and in what format is it typically encoded?

Token: ID token

Format: JWT (JSON Web Token)

Structure: The ID Token is a signed JWT with three parts:

- **Header:** Algorithm and token type ({"alg":"RS256", "typ":"JWT"})
- **Payload:** User identity claims (sub, iss, aud, exp, iat, nonce, name, email, etc.)
- **Signature:** Cryptographic signature from the authorization server

Encoding: Base64URL-encoded, separated by dots:

```
header.payload.signature
```

Example claims in payload:

```
{
  "iss": "https://accounts.google.com",
  "sub": "108123456789012345678",
  "aud": "812741506391-abc123.apps.googleusercontent.com",
  "exp": 1738123456,
  "iat": 1738119856,
  "nonce": "n-086_WzA2Mj",
  "name": "John Doe",
  "email": "johndoe@example.com",
  "picture": "https://lh3.googleusercontent.com/..."
}
```

3. **Do you agree** that "OAuth 2.0 provides true user authentication"? Explain in 2–3 sentences why or why not.

I disagree, OAuth 2.0 does not provide true authentication

OAuth 2.0 is an **authorization** framework, not an authentication protocol.

Why OAuth 2.0 is not authentication:

1. **Purpose:** OAuth 2.0 answers "What can this app access?" not "Who is this user?"
2. **Access token ambiguity:**
   - The access token grants permissions to resources
   - It does *not* contain standardized user identity information
   - Token format is **opaque** (could be a random string)
3. **Pseudo-authentication problem:**
   - Apps sometimes use OAuth 2.0 for authentication by calling a UserInfo endpoint with the access token
   - Logic: "If I can fetch the user's profile, they must be authenticated"
   - This is **pseudo-authentication** — using authorization as proof of authentication
   - **Vulnerability:** An attacker could obtain an access token through other means (e.g., stolen token) and impersonate the user
4. **No standardized identity claims:**
   - OAuth 2.0 does not define how user identity should be represented
   - Different providers return different formats

Part V: Case Study (24 points)

Background Scenario: MedChain Hospital Cloud System

MedChain Hospital is building a **cloud-based patient record management system**. Patient records are stored on a semi-trusted cloud server operated by a third party. Multiple classes of users — cardiologists, oncologists, administrators, and insurance auditors — need to access records, but with **strictly different access rights**:

- **Cardiologists** may read records tagged as `cardiology`.
- **Oncologists** may read records tagged as `oncology`.
- **Administrators** may read all records.
- **Insurance auditors** may read only billing summaries tagged `billing`.

MedChain's security requirements are:

I. Patient data must remain **encrypted at rest** on the cloud server.

II. The cloud server must **not** be trusted with plaintext data.

III. Access control must be enforced **cryptographically** (not just by the server's access control list).

IV. Uploaded records must be **verifiably authentic** (i.e., the receiver can confirm who uploaded the record and that it has not been tampered with).

V. The system should support **physician login via a web portal** using Single Sign-On (SSO) without requiring separate credentials for each hospital sub-system.

Case Study — Question A (10 points)

The hospital's security architect proposes the following encryption scheme for uploading a cardiology record M:

(1) Generate a random AES session key k_{AES}.

(2) Encrypt the record: $CT_M = \texttt{Enc}_{AES}(k_{AES},\ M)$.

(3) Define an access policy:
$T =$ `(role = cardiologist AND dept = cardiology) OR (role = admin)`.

(4) Encrypt the session key: $CT_K = \texttt{Enc}_{CP\text{-}ABE}(PK,\ k_{AES},\ T)$.

(5) Upload $(CT_M,\ CT_K)$ to the cloud.

Answer the following sub-questions. Show full reasoning.

1. **(2 pts)** Write the full decryption procedure a physician with attributes $S = \{$`role=cardiologist, dept=cardiology`$\}$ must perform to recover M. Use proper notation, e.g. $\texttt{Dec}_{CP\text{-}ABE}(\cdots)$.

1) $k_{AES} = \texttt{Dec}_{CP\text{-}ABE}(CT_K, SK_U)$
   → physician's secret key bound to attribute S

2) $M = \texttt{Dec}_{AES}(CT_M, k_{AES})$ #

2. **(2 pts)** An oncologist with attributes $S' = \{$`role=oncologist, dept=oncology`$\}$ attempts to decrypt CT_K. Will this succeed? Justify using the access policy T.

$\texttt{Dec}_{CP\text{-}ABE}(CT_K, SK_U) = \perp$ (Decryption will FAIL)

Because the attribute set S' does not satisfy policy T

This demonstrates CP-ABE's fine-grained access control, only users with attributes that satisfy the policy can decrypt.

3. **(3 pts)** Two users — Dr. Alpha (attributes: `role=cardiologist`) and Dr. Beta (attributes: `dept=cardiology`) — attempt to combine their secret keys to satisfy the policy T.

(a) What is this type of attack called?

(b) Does CP-ABE allow this? Explain briefly why or why not.

(c) What is the technical mechanism CP-ABE uses to prevent it?

a) Collusion attack

b) No. CP-ABE cryptographically prevents collusion attacks.

c) CP-ABE prevents collusion using secret randomization:

↳ each user's secret key SK_U contain a unique random value r_U chosen during key generation

this random value is embedded into every attribute component of the key

different users get different random values: $r_{Alpha} \neq r_{Beta}$

- To decrypt, the user must combine attribute components from their key
- All components must be **bound to the same random value** r_U
- The ciphertext structure enforces that only matching r values combine correctly

4. **(3 pts)** The hospital decides to switch from `AES-CBC to AES-GCM` for encrypting M.

(a) What additional security property does AES-GCM provide that AES-CBC alone cannot?

(b) In AES-GCM, what component is responsible for producing the authentication tag?

(c) If an attacker flips a bit in CT_M during transit, what will happen when the recipient attempts to decrypt it with AES-GCM?

a) AES-GCM provides authenticated encryption ๆ ใช่

- Integrity
- Authenticity

AES-CBC alone ให้ได้แค่ confidentiality

but it lacks:

- No integrity check
- No authentication
- Padding oracle attack

AES-GCM adds:

- Authentication tag
- Tag verification before releasing plaintext

AES-GCM is an AEAD scheme.

b) The authentication tag is produced by GMAC which is a part of GHASH

c) สมมติว่า recipient receives corrupted CT_M' (bit flip)

recipient attempts decryption:

1. Decrypt ciphertext: Produce corrupted plaintext M'
2. Recompute authentication tag: $T' = GMAC(CT_M')$
3. Compare with received tag: $T' \neq T$ (stored in ciphertext)

⇓

Tag verification FAILS

⇓

Decryption is REJECTED!

ซึ่งมันเป็นสิ่งที่ดีมากเลยนะ คุณอ่าน

ข้อมูลโรงพยาบาลจะให้ corrupted ได้ไง

10 mg → 90 mg (ตายได้เลย 555)

A+ → B-

AES-CBC

จะให้คุณอ่ะ ไม่ได้นะ!

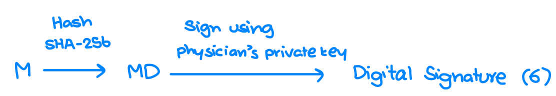
Case Study — Question B (8 points)

Requirement IV states that uploaded records must be verifiably authentic. The hospital architect proposes using **digital signatures**.

1. **(3 pts)** Describe step-by-step how a physician (sender) would digitally sign a patient record M before uploading it, and how the cloud audit system (verifier) would verify the signature. Your answer must specify:

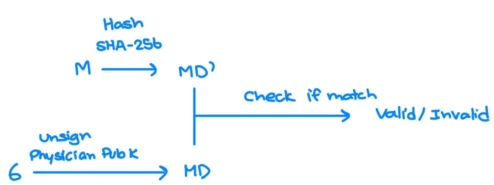
   - Which key is used at each step,
   - The role of a hash function,
   - What a successful verification proves (and what it does not prove).

Signing:



Then upload (M, 6) to the cloud

Verifying:



What successful verification PROVES:

1. **Authentication:** The record was signed by the physician who owns the private key corresponding to $PK_{physician}$
2. **Integrity:** The record has not been modified since signing (any change to M would change h' and cause verification to fail)
3. **Non-repudiation:** The physician cannot deny having signed the record (only they possess the private key)

What it does NOT prove:

1. **Confidentiality:** The record M is *not* encrypted by the signature (it's typically in plaintext or encrypted separately)
2. **Freshness/Timeliness:** The signature doesn't prove *when* it was created (could be an old signature)
   - Solution: Include timestamp in M and use trusted timestamping services
3. **Content validity:** The signature proves the physician signed it, but doesn't prove the medical information is *correct* (physician could sign incorrect data)
4. **Authorization:** The signature doesn't prove the physician was *authorized* to create this specific type of record (requires additional access control checks)

2. **(2 pts)** The cloud audit system must trust the physician's public key. Explain the role of a **Certificate Authority (CA)** and a **digital certificate** (X.509) in establishing this trust. What problem arises if a physician's private key is compromised?

The CA is a trusted third party that:

1. Verifies identity:
   - Confirms the physician's identity through official documents
   - Checks credentials (medical license ...)
2. Issue digital certificate (x.509)
   - Binds physician's public key to their verified identity
   - ของ cer มีข้อมูลอะไรบ้าง ก็ลองดู
3. Signs the certificate

Audit System —trusts→ CA —certifies→ Physician

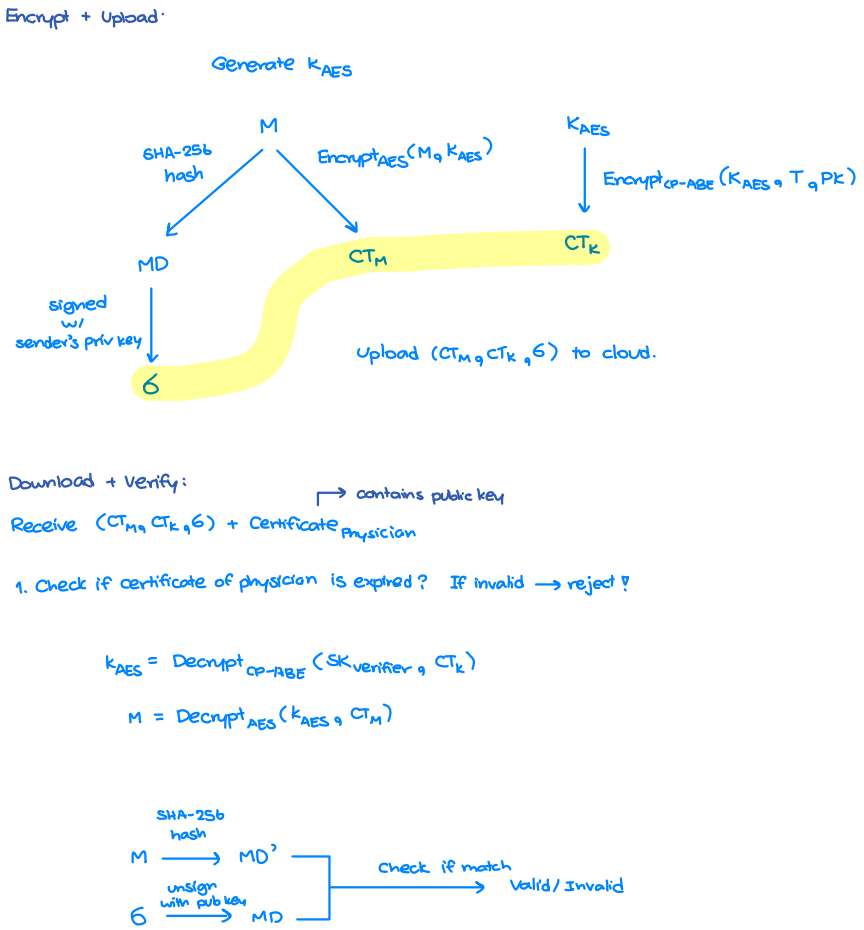
If private key $SK_{physician}$ exposed.

1. Impersonation
2. Tampering
3. Liability (ต้องรับผิดชอบ all signature made with their key)
4. Loss of non-repudiation

จะหลุดไปแล้ว ทำไง???

1. Report to CA
2. CA revokes the certificate by adding it to:
   - CRL
   - OCSP
3. Generate new key pair (user should do this?)
4. Obtain new certificate from CA

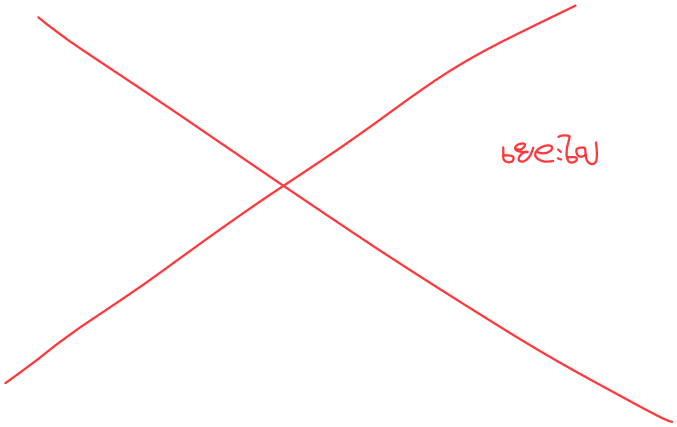
3. **(3 pts)** The hospital now combines digital signatures and CP-ABE encryption. Write the complete protocol equations (encryption + signing) that a physician performs before uploading (CT_M, CT_K), and the equations that a verifier performs after downloading. Use proper notation.



Case Study — Question C (6 points)

Requirement V states that physicians should log in via SSO using a web portal. The hospital's IT team proposes using **OpenID Connect (OIDC)** on top of **OAuth 2.0**.

1. **(2 pts)** Briefly describe the OAuth 2.0 Authorization Code Flow (steps 1–5 are sufficient) as it would apply when a physician logs into the MedChain portal using their hospital identity provider (IdP).



2. **(2 pts)** The OIDC **ID Token** contains the physician's identity information.

   (a) In what format is the ID Token typically encoded?

   (b) List two claims that would be particularly useful for MedChain's access control system (beyond `sub` and `iss`).

The ID Token is encoded as a **JWT (JSON Web Token)**.

Structure: Three Base64URL-encoded parts separated by dots:

`header.payload.signature`

Example ID Token:

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.
eyJpc3MiOiJodHRwczovL2lkcC5ob3NwaXRhbC5jb20iLCJzdWIiOiJwaHlzaWNpYW4xMjM...
SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c
```

Decoded payload example:

```
{
  "iss": "https://idp.hospital.com",
  "sub": "physician12345",
  "aud": "medchain_portal",
  "exp": 1738123456,
  "iat": 1738119856,
  "nonce": "n-0S6_WzA2Mj",
  "name": "Dr. Sarah Chen",
  "email": "s.chen@hospital.com",
  "role": "cardiologist",
  "department": "cardiology",
  "employee_id": "EMP-67890",
  "medical_license": "MD-12345-CA"
}
```

Beyond standard claims (`sub`, `iss`, `aud`, `exp`, `iat`), two particularly useful claims for MedChain:

1. `role` claim

- Value: `"cardiologist"`, `"oncologist"`, `"admin"`, etc.
- Usage: Map directly to CP-ABE attribute `role=cardiologist`
- Access control: Determines which types of records physician can access
- Example policy: `(role=cardiologist AND dept=cardiology) OR admin`

2. `department` claim

- Value: `"cardiology"`, `"oncology"`, `"emergency"`, etc.
- Usage: Map to CP-ABE attribute `dept=cardiology`
- Fine-grained control: Ensures physicians only access records in their department
- Cross-departmental collaboration: Some records may require multiple departments

3. **(2 pts) Do you agree** that OIDC alone (without CP-ABE) is sufficient to enforce MedChain's Requirement III ("access control must be enforced cryptographically, not just by the server")? Justify your answer in 3–5 sentences, explaining the fundamental difference between server-side access control and cryptographic access control.

ตัวเอง disagree ขี้เกียจเหตุผล 555

Aspect	**OIDC (Server-side AC)**	**CP-ABE (Crypto AC)**
Where data is decrypted	Server	Client (physician's device)
Server can read data?	Yes (plaintext stored)	No (only ciphertext)
What if server is hacked?	All data exposed	Data remains encrypted
What if admin is malicious?	Admin can read anything	Admin cannot read (no key)
Access control enforcement	Software policy (bypassable)	Cryptographic (mathematical)
Trust requirement	Must trust server	Only trust Attribute Authority