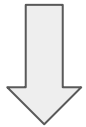
1, Hash function is not encryption
2, A hash code is a one-way function that produces random-like numbers

# Digital Signature:
# Using sender's private key
# And receiver uses sender's public key to decrypt



No-repudiation：The signer is hard to deny
Authenticity
Data Integrity

# Message Authention Codes (MACs)

**Question:** *“If you send a message to your friend over the internet, how can you be sure it hasn’t been tampered with? And how do you know it really came from your friend?”*

This introduces the concept of **data integrity** and **authentication**, leading to the need for **Message Authentication Codes (MACs)**.

**Message Authentication Code (MAC)**: A cryptographic code that ensures a message hasn’t been altered and verifies the identity of the sender.

- **Integrity**: Ensures the message wasn’t tampered with during transmission.
- **Authenticity**: Confirms the message came from a trusted sender.

**Step 1: Generating the MAC**

1. The sender has a message `M = "Hello"` and a shared secret key `K`.
2. Using a cryptographic algorithm (e.g., HMAC), the sender generates the MAC:
   **MAC = HMAC(K, M)**
3. The sender transmits the message along with the MAC:
   `("Hello", MAC)`

**Step 2: Verifying the MAC**

1. The receiver receives the message `M` and the MAC.
2. The receiver uses the same secret key `K` and algorithm to generate their own MAC: **MAC' = HMAC(K, M)**
3. They compare the two MACs:
    - If **MAC == MAC'**, the message is authentic and hasn't been tampered with.
    - If they don't match, the message may have been altered or isn't from a trusted source.

# Hash function
# Vs.
# HMAC function

**Hash Function (e.g., SHA-256)**:

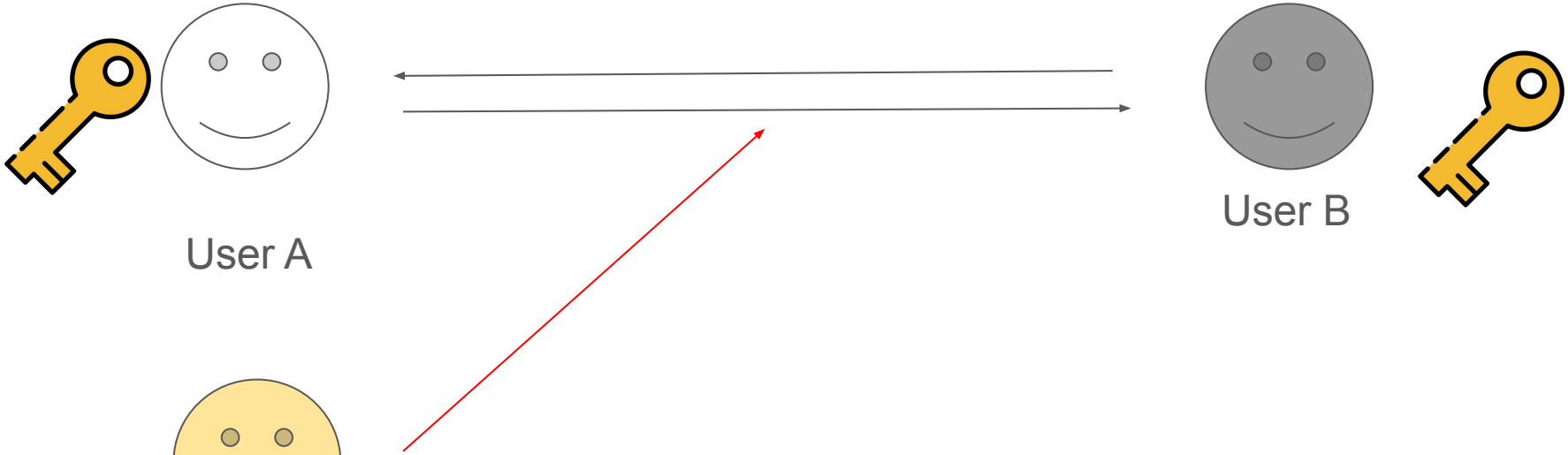
1. Input message: `"Hello"`
2. Apply hash function: `SHA256("Hello")`
3. Output: A fixed-size hash, like `2cf24dba5fb0...`

- **Key Point**: Anyone with `"Hello"` can compute the same hash.

**HMAC Function (e.g., HMAC-SHA256)**:

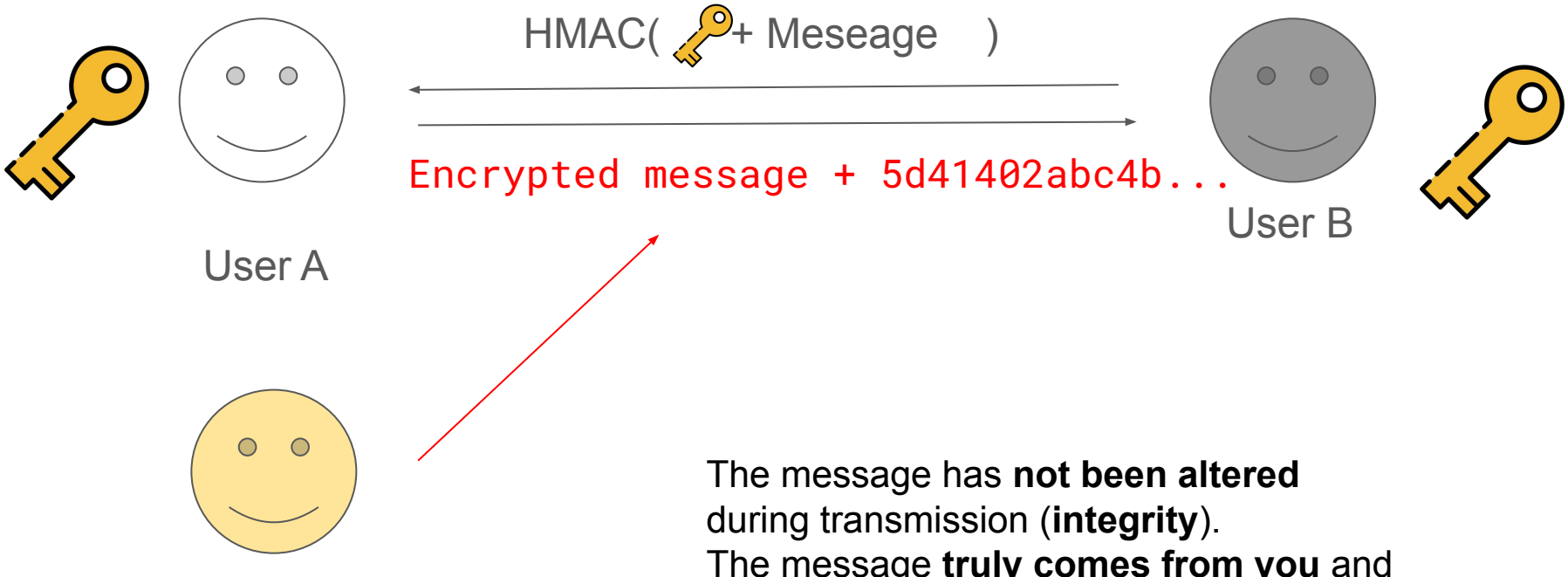
1. Input message: `"Hello"`
2. Secret key: `"my_secret_key"`
3. Apply HMAC: `HMAC_SHA256("my_secret_key", "Hello")`
4. Output: A unique HMAC, like `5d41402abc4b...`

- **Key Point**: Only those with `"my_secret_key"` can generate or verify this HMAC.




Membership attack

**Encryption alone doesn't guarantee the message hasn't been tampered with or that it actually came from you.** An attacker might not be able to read the message, but they could still modify it or even impersonate you.




The message has **not been altered** during transmission (**integrity**).
The message **truly comes from you** and not an attacker (**authenticity**).

# **CSC116** Access Control: Role-based Access Control, Fine-grained Access Control

# What is Access Control?

Definition: **Restricting unauthorized access to systems/data.**

Importance: Security, compliance, data protection

Data Privacy

# 5 Access Control Methods

- **Discretionary Access Control (DAC)**: The owner of the resource decides who gets access, e.g., file sharing permissions in Windows.
- **Mandatory Access Control (MAC)**: Access is determined by a central authority based on classifications. **Example: Military or government systems.**
- **Role-Based Access Control (RBAC)**: Access is based on the user's role within an organization. **Example: Admins have more privileges than regular users.**
- **Attribute-Based Access Control (ABAC)**: Access is granted based on a combination of attributes (e.g., time of access, location, user clearance). Also, it is known for providing **fine-grained access control.**

# Fine-grained access control

Fine-grained access control allows for **very detailed, specific rules** about who can access what, under precise conditions.

**“Only employees in the Finance department can access payroll data, but only during business hours and from a secure company device.”**

**“A doctor can access patient records, but only for patients assigned to them and only from within the hospital network.”**

**Fine-grained access control is part of ABAC**

**Question 1 : In a military database, documents are labeled as "Top Secret," "Confidential," or "Public." Only users with the appropriate security clearance can access certain documents, regardless of their role.**

*Which access control model is being used?*

**A)** Role-Based Access Control (RBAC)

**B)** Discretionary Access Control (DAC)

**C)** Mandatory Access Control (MAC)

**D)** Attribute-Based Access Control (ABAC)

**Question 2 : A hospital system allows doctors to access patient records only if they are the primary physician assigned to that patient, and only while they are inside the hospital premises.**

*Which access control model is being used?*

**A)** Discretionary Access Control (DAC)

**B)** Role-Based Access Control (RBAC)

**C)** Attribute-Based Access Control (ABAC)

**D)** Mandatory Access Control (MAC)

## **Question 3**：Google Docs Sharing Demo

Question: which kinds of access control model?

# Authorization and Access Control

1. **Access Control**
   This is a broader concept that refers to the entire process of **managing and restricting access to resources**. It includes:

- **Authentication**: Verifying the identity of a user, such as through passwords, biometrics, or multi-factor authentication.
- **Authorization**: After authentication, determining what resources the user is allowed to access and what actions they can perform.
- **Audit**: Monitoring and recording access activities to detect and respond to anomalies.

2. **Authorization**
   **Authorization is a subset of Access Control**, focusing specifically on **determining permissions** after a user's identity has been verified. For example:

- An employee may be authorized to read company documents but not edit them.
- An administrator may have full access to system settings, while regular users have limited permissions.

## Access Control

- **Authentication:** When a user tries to log into the EHR system, they must verify their identity using their university credentials, such as a username and password, or even multi-factor authentication (e.g., using a secure token).
- **Authorization:** Once authenticated, the system checks what specific permissions the user has. For example:
    - Medical students may be authorized to view patient records but cannot edit or delete them.
    - Professors and licensed physicians may have authorization to both view and update patient records.
- **Auditing:** Detecting unauthorized attempts or suspicious activities.

**Access Control** is the overall process that includes verifying identities, assigning permissions, and monitoring activities. (Much broader)

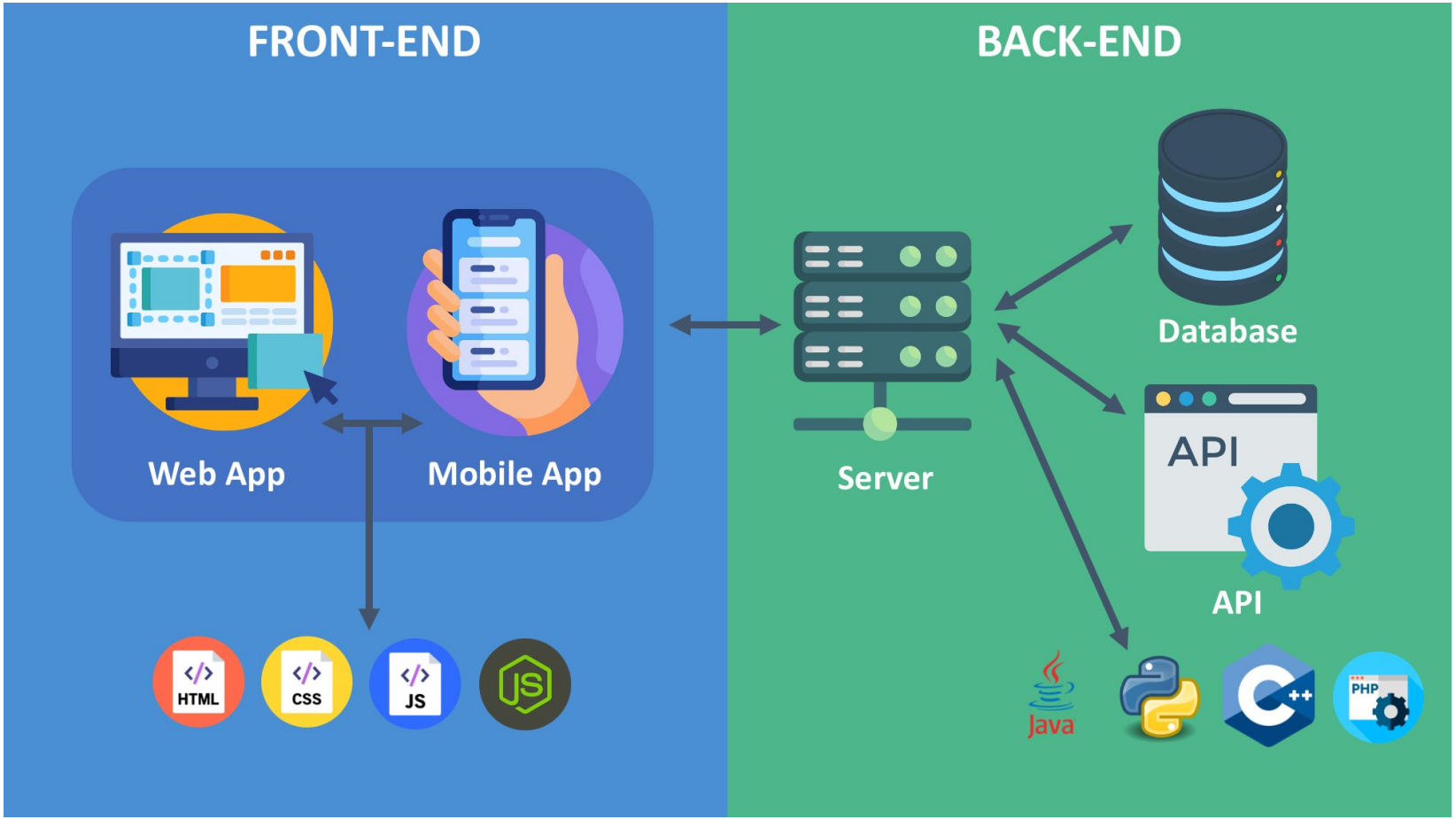
**Authorization** is the step that **determines what resources and actions** an authenticated user can access within the system.

# We need Principle of Least Privilege

# Principle of Least Privilege

The **Principle of Least Privilege (PoLP)** is a fundamental cybersecurity concept that states **users, applications, and systems should be granted the minimum level of access—or permissions—necessary to perform their specific tasks** and nothing more.

# How to implement Role-based Access Control？


FRONT-END
BACK-END
Web App
Mobile App
Server
Database
API
API
HTML
CSS
JS
Java
PHP

# Step 1: Database Design

## Access Control Lists (ACL)

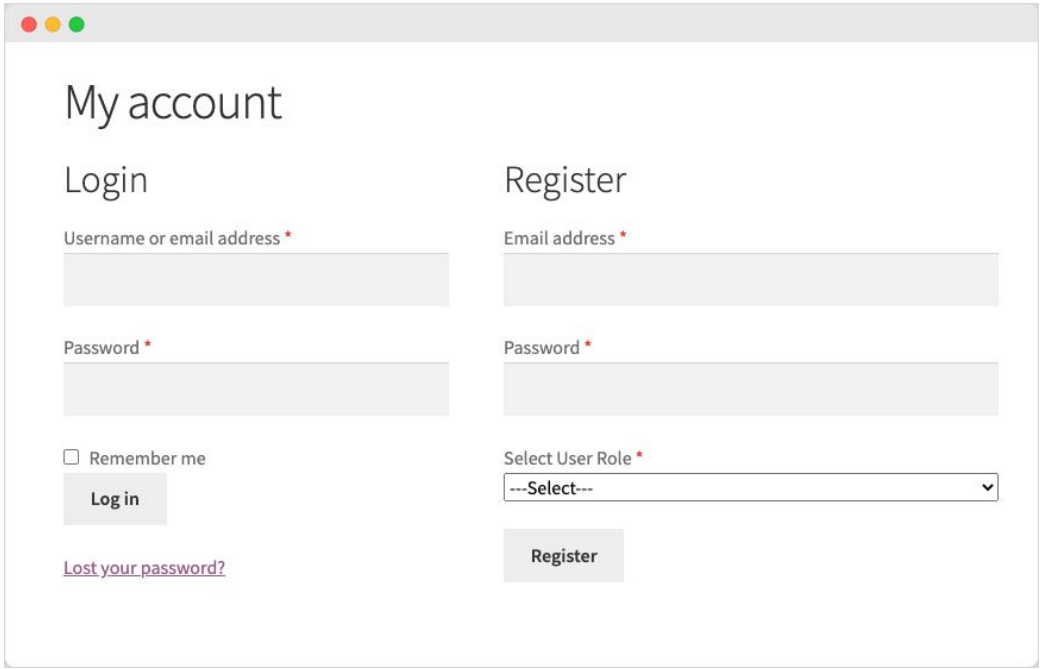
- **Users**: Store user details (e.g., `id`, `username`, `hash_password`).
- **Roles**: Define roles (e.g., `admin`, `patients`, `nurses`, `doctors`).
- **Permissions**: List actions (e.g., `read`, `write`, `update`, `remove`).
- **User_Roles**:
- **Role_Permissions**:

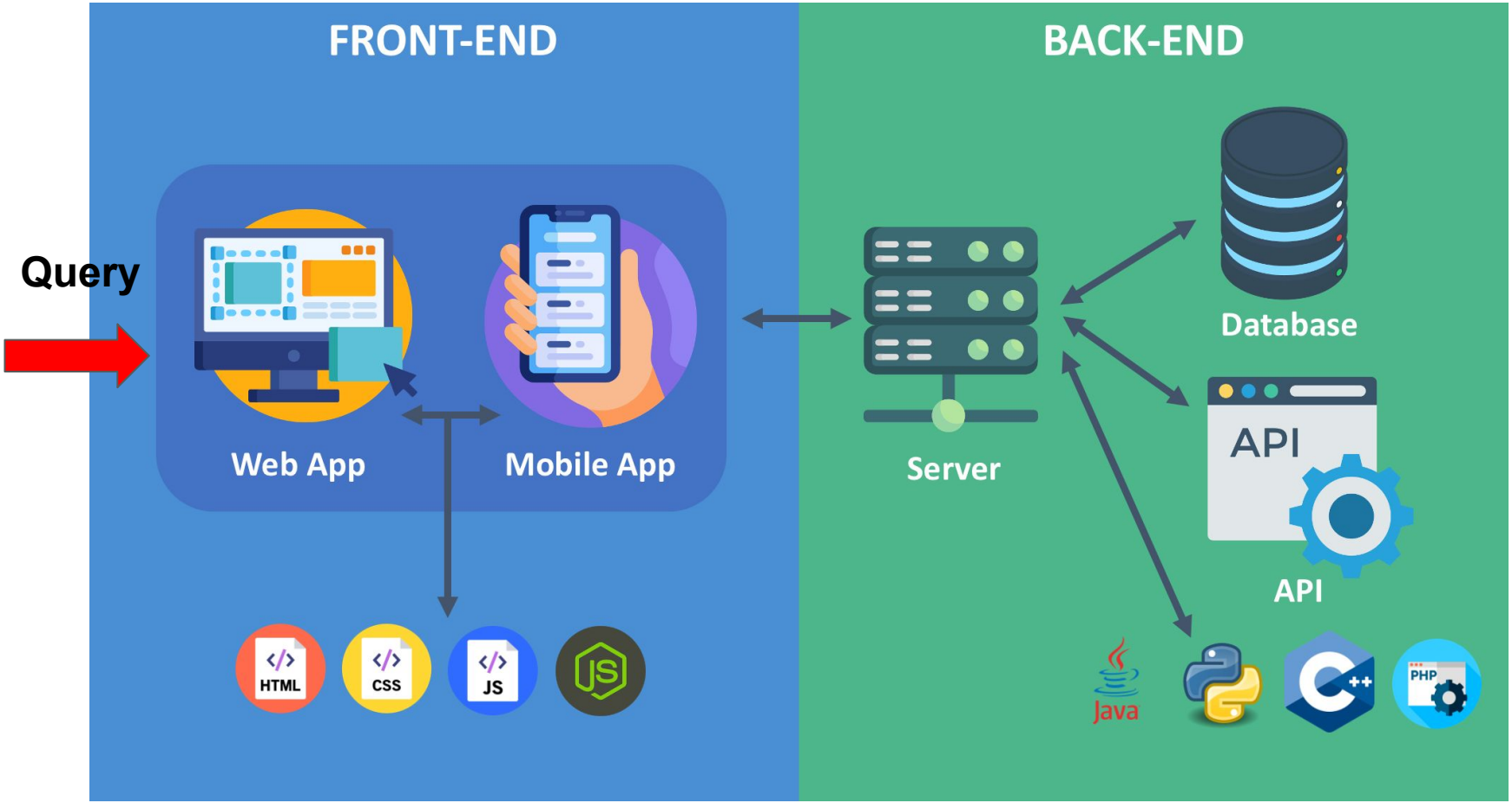
```
CREATE TABLE roles (id INT, name VARCHAR(255));
CREATE TABLE permissions (id INT, name VARCHAR(255));
CREATE TABLE user_roles (user_id INT, role_id INT);
CREATE TABLE role_permissions (role_id INT, permission_id INT);
```

# Step 2: Backend Development

API developments.

# Step 3: Frontend Development




FRONT-END
BACK-END
Query
Web App
Mobile App
Server
Database
API
API
HTML
CSS
JS
Java
PHP

Question：

If a web system with low quality access control, and the attackers modified the EHR patient records.

What is the next step if you are a developer?

# Data Backup and Disaster Recovery

**Data Backup and Disaster Recovery** are essential components of cybersecurity, particularly in environments like healthcare where data loss can directly impact patient care and organizational operations.

**Data backup** refers to the process of creating copies of important data and storing them in a separate, secure location. This ensures that if the original data is lost, corrupted, or compromised (due to hardware failure, human error, cyberattacks like ransomware, or natural disasters), you can restore it from the backup.

## Data backup and Local Storage

Local storage is a dedicated backup drive at your place of business where you make Full, Incremental, or Differential copies of your data.

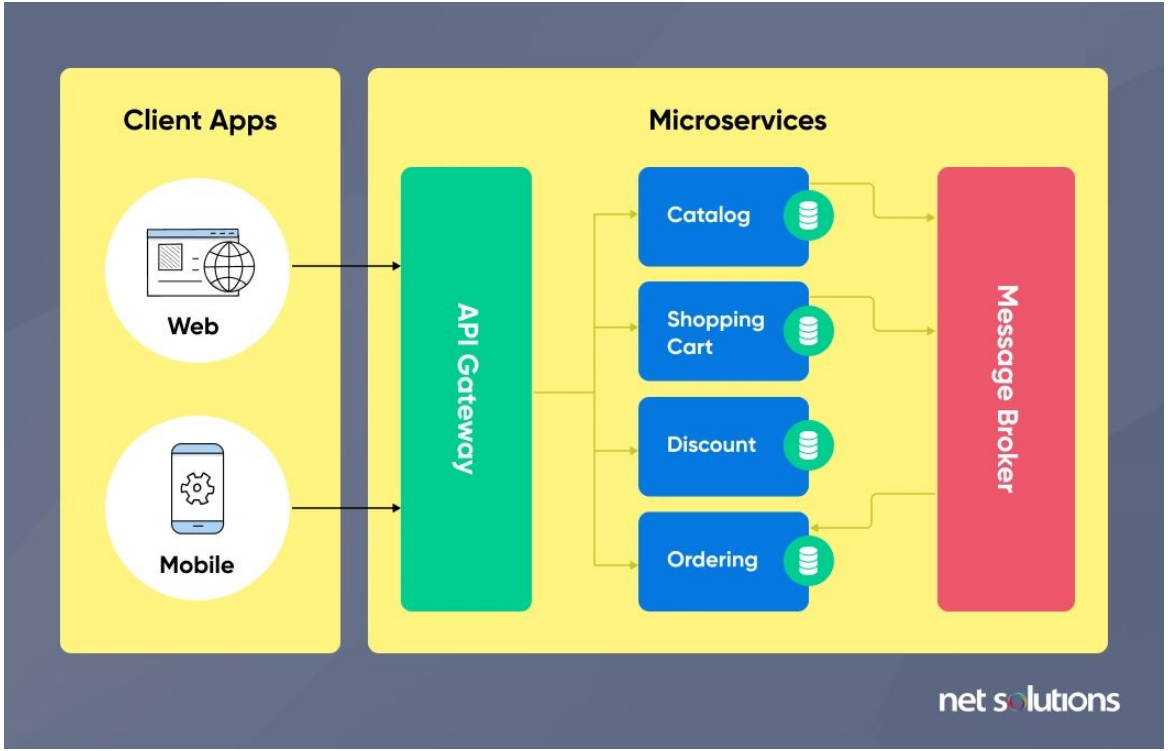
## Data backup and Off-site Storage

Off-site storage is an external storage media used to back up and store data at an off-site location. Because it’s stored off-site, it provides additional protection against catastrophic events. Methods of uploading your backup are the Internet or with a physical connection.

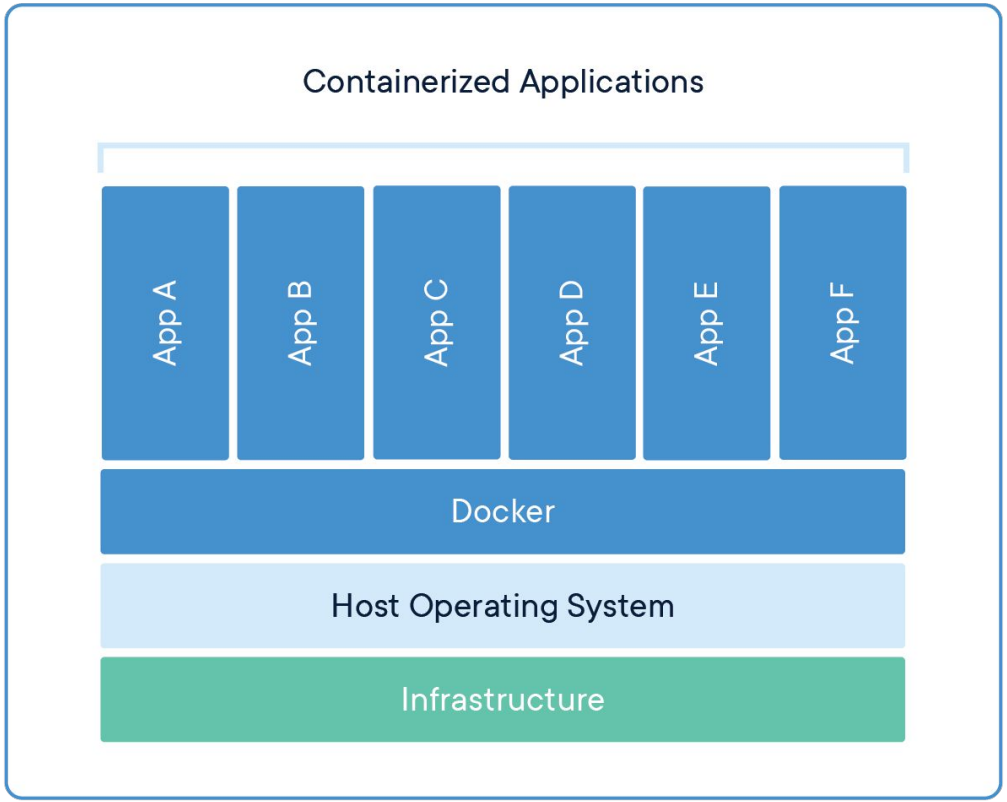
## Data backup and Cloud Storage

Cloud Storage is a virtual platform that provides off-site, scalable storage resources, dynamically provisioned per the business’s technical requirements.

# Distributed Storage Systems for Micro-Services

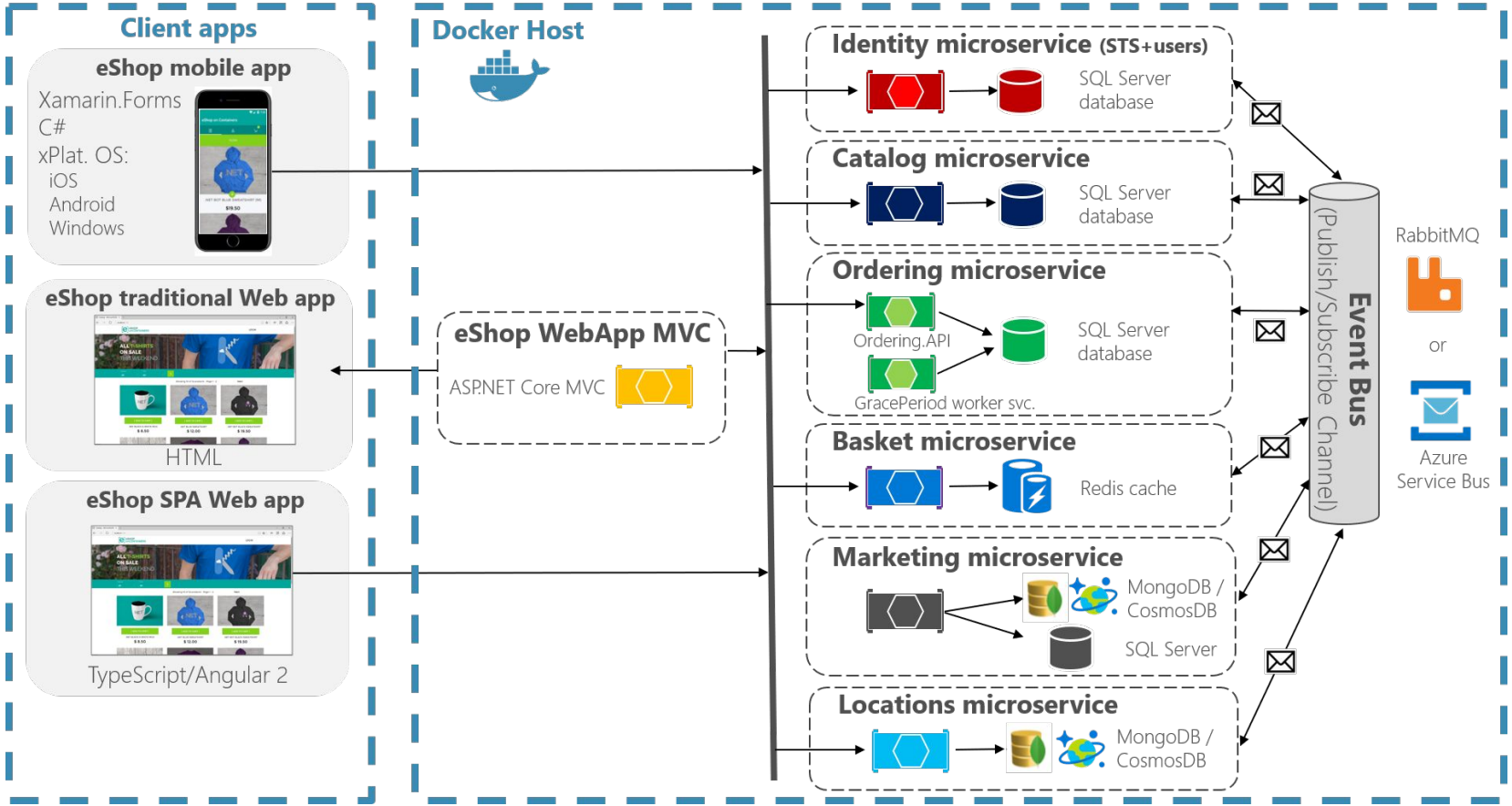


# Docker Container



# eShopOnContainers reference application

(Development environment architecture)



# Conclusions

Distributed storages can mitigate the whole databases be attacked.

It separates the whole services into different micro-services to improve the development speed.

# Generative AI Security and Privacy Issues

Generative AI (GenAI) is a type of artificial intelligence (AI) that can create new content, such as images, videos, text, and music. It can also learn from existing data and reuse it to solve new problems.

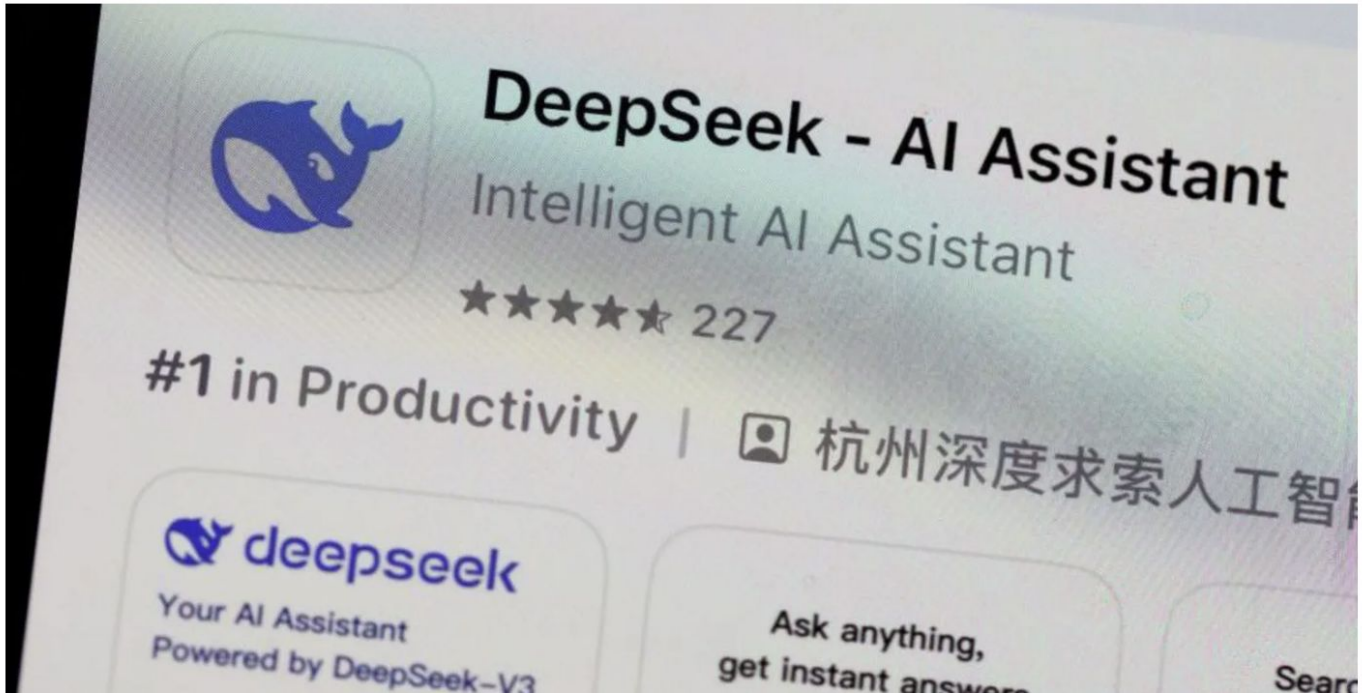

https://www.cbsnews.com/news/deepseek-ai-raises-national-security-concerns-trump/

# Be careful with DeepSeek, Australia says - so is it safe to use?

5 days ago

Share Save

**Tom Gerken**
Technology reporter



MATT BURGESS LILY HAY NEWMAN SECURITY JAN 27, 2025 5:10 PM

# DeepSeek's Popular AI App Is Explicitly Sending US Data to China

**Amid ongoing fears over TikTok, Chinese generative AI platform DeepSeek says it's sending heaps of US user data straight to its home country, potentially setting the stage for greater scrutiny.**

# Why it has privacy issues

**Inadequate Privacy Safeguards**: Some LLMs may collect user inputs to improve future model versions. If these inputs are not properly anonymized or secured, there’s a risk that sensitive information could be exposed or misused.