LECTURE #5

PHP (Hypertext Preprocessor)

- has evolved significantly since its creation in **1994** by **Rasmus Lerdorf**

Cookies

- Small pieces of data stored on the client's browser, used to remember information between requests.
- **setcookie()** function

Access Control

- Determining what resources a user can access and what operations they can perform

- **Sessions -** to maintain user state and data across multiple pages.
- **User Authentication** - verifying user credentials against stored data
- **Role-Based Access Control (RBAC)** - defining roles and assigning permissions to these roles

Frameworks like Laravel, Symfony, CodeIgniter

- streamlined the process of managing cookies and implementing robust access controls

USING COOKIES IN PHP

- **setcookie() -** this function should be called before any output is sent to the browser, as it modifies the HTTP headers.
- **$_COOKIE** - to access a cookie
- To delete a cookie, you set it with an expiration date in the past.

HTTP authentication

- Method to ensure that users are who they claim to be by verifying their credentials.

Salting

- Involves adding a random value to a password before hashing it.

Verifying Passwords

- When a user attempts to login, the stored hash must be compared to the hash of the provided password.
- **password_verify()**

Storing Usernames and Passwords

- Usernames are stored as plain text or lightly sanitized strings in the database.

USING SESSIONS

- Used to store and manage user data across multiple pages.
- **session_start()**

Ending a sessions

- Unsetting all sessions variables
- Destroying the session itself

Setting a session timeout

- Helps in automatically logging out users after a period of inactivity

Session Security

- To prevent attacks like session hijacking and fixation

Database Security

- Ensuring that data is protected against unauthorized access, corruption, or loss.

B**ackup and Recovery**

- Involves creating copies of the database at regular intervals and ensuring these copies can be restored when needed.

Table Maintenance

- Ensuring database performance and integrity

MySQL administration

- Involves managing users, securing connections, and monitoring database performance.

LECTURE # 7

Web Security

- Ensuring the security of web applications protects sensitive data, maintains user trust, and prevents malicious activities.

Importance of security

- Vital for protecting sensitive user data, maintaining the integrity of web applications, and preventing unauthorized access.

Validation and sanitization of user inputs

- Processed used to ensure that user inputs are safe and meet expected formats
- Validation checks if the input meets specific criteria
- Sanitization cleans the input to remove any harmful characters

Preventing SQL injection

- Involves using prepared statements and parameterized queries
- **SQL injection** - A technique used by attackers to manipulate SQL queries by injecting malicious code.

Preventing XSS

- Involves escaping user inputs before displaying them on the page
- Cross-Site Scripting(XSS) attacks inject malicious scripts into web pages viewed by other users.

Preventing remote execution

- Involves validating and sanitizing files uploads and commands
- Remote code execution allows attackers to execute arbitrary code on a server

Preventing Session Hijacking

- Includes regenerating session IDs and using secure cookies
- Session Hijacking involves stealing a user's session ID to impersonate them.