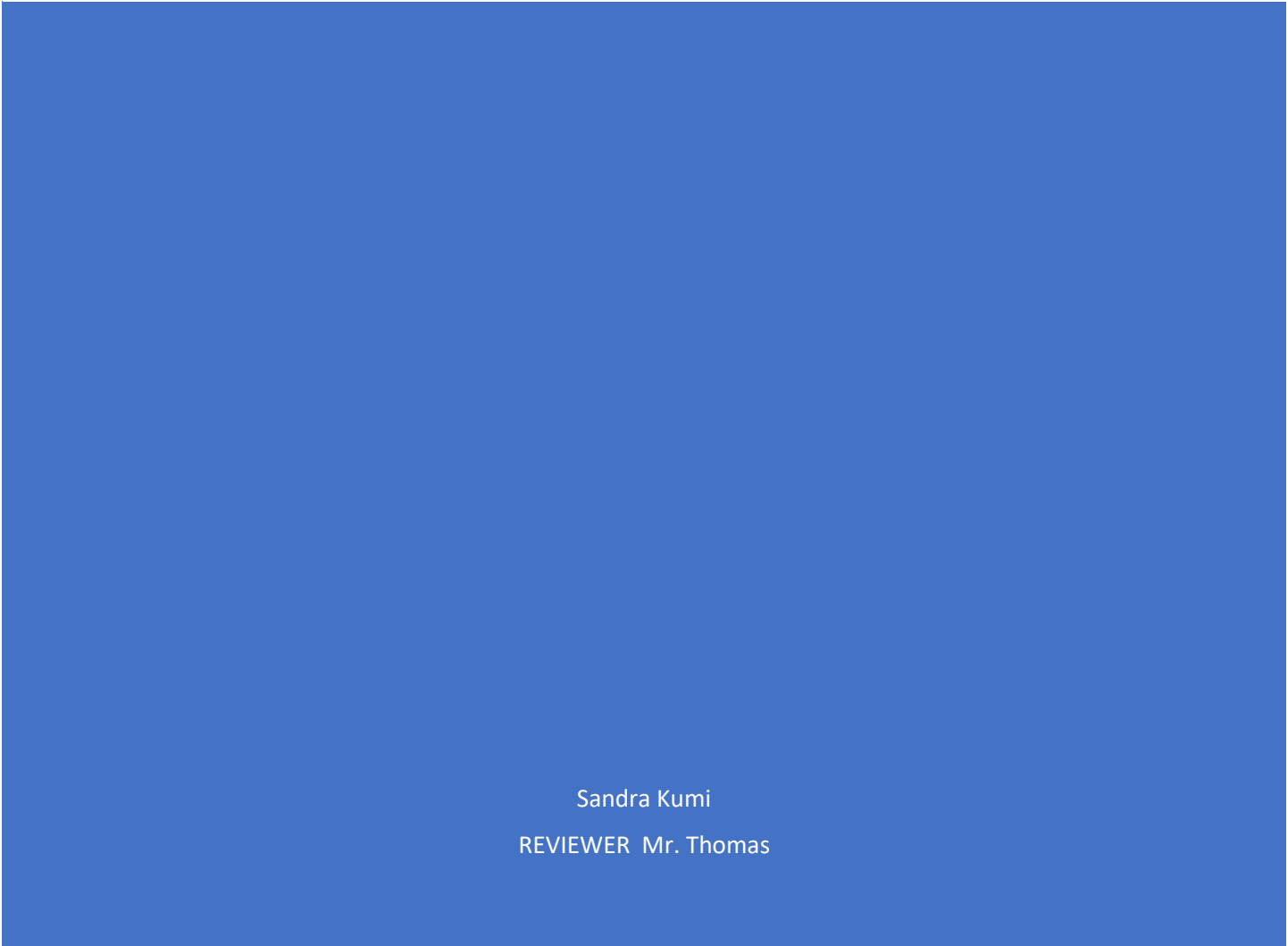




SUMMARY OF CSRF, OAUTH 2.0, OPENID CONNECT, AND SESSION MANAGEMENT



Sandra Kumi
REVIEWER Mr. Thomas

Summary of CSRF, OAuth 2.0, OpenID Connect, and Session Management

1. Cross-Site Request Forgery (CSRF)

Definition: Cross-Site Request Forgery (CSRF) is a security vulnerability that tricks a user into performing actions on a web application in which they are authenticated, without their consent. This can lead to unauthorized transactions or actions on behalf of the user.

Protection Mechanisms:

- **CSRF Tokens:** A unique token is generated and included in each HTTP request made by the client. This token is validated by the server to ensure that the request is legitimate.
- **SameSite Cookies:** Setting cookies with the SameSite attribute helps prevent the browser from sending cookies along with cross-site requests.

Implementation in our Project:

- CSRF tokens are implemented in forms and AJAX requests to ensure that only legitimate requests are processed by the server.
- Spring Security's CSRF protection is enabled by default, and custom configurations can be applied as needed.

2. OAuth 2.0

Definition: OAuth 2.0 is an authorization framework that allows third-party applications to obtain limited access to a user's resources on a server without exposing the user's credentials. It is widely used in web applications for secure access delegation.

Grant Types:

- **Authorization Code Grant:** Used in web and mobile applications to obtain access tokens after the user authenticates.
- **Client Credentials Grant:** Used by applications to authenticate and access resources on their own behalf.
- **Implicit Grant:** Used in single-page applications where the access token is directly returned without an authorization code.
- **Password Grant:** Allows the exchange of a username and password for an access token (not recommended due to security risks).

Implementation in our Project:

- OAuth 2.0 is implemented using the Authorization Code Grant flow. Users are redirected to an authorization server, authenticate, and grant access to our application. An authorization code is then exchanged for an access token.

3. OpenID Connect (OIDC)

Definition: OpenID Connect is an identity layer on top of OAuth 2.0, enabling clients to verify the identity of the end-user based on the authentication performed by an authorization server. It also provides basic profile information about the user.

Components:

- **ID Token:** A JSON Web Token (JWT) that contains user identity information.
- **UserInfo Endpoint:** An endpoint that returns additional user information.

Implementation in our Project:

- OpenID Connect is used to authenticate users and retrieve their profile information. The ID token is validated to ensure the authenticity of the user's identity.

4. Session Management

Definition: Session management is the process of securely managing user sessions in a web application. This includes handling authentication, maintaining session state, and ensuring the security of user data during a session.

Key Concepts:

- **Session Cookies:** Cookies are used to store session identifiers, which are essential for maintaining a user's session across multiple requests.
- **Session Expiration:** Sessions are configured to expire after a certain period of inactivity to prevent unauthorized access.
- **Session Storage:** Sessions can be stored in-memory, in a database, or using distributed caching mechanisms.

Implementation in our Project:

- Session management is implemented using Spring Security. Session cookies are used to maintain user sessions, and session expiration is configured to enhance security.