Project Report: Authentication and PKI Lab

1. Lab Objectives

The objective of this lab was to implement various security features using Spring Security, including:

- Basic and token-based authentication.
- Public Key Infrastructure (PKI) setup.
- HTTPS configuration for secure communication.

2. Lab Exercises

Exercise 1: Authentication Methods

The application implements two types of authentication methods:

- **Basic Authentication**: Utilizes the built-in form-based login with user credentials stored in an in-memory database. The `UserDetailsService` provides user details for authentication, configured with roles such as `USER` and `ADMIN`.
- **Token-Based Authentication**: Uses JWT (JSON Web Token) to secure API endpoints. The custom `JwtAuthenticationFilter` is added to the security filter chain to intercept and validate JWT tokens in incoming requests.

Java code
```
http.addFilterBefore(jwtAuthenticationFilter,
UsernamePasswordAuthenticationFilter.class);
```

Exercise 2: PKI Setup

To ensure secure communication, PKI (Public Key Infrastructure) was set up by generating SSL certificates. These certificates are configured in the Spring Boot application to enable HTTPS communication.

Certificates and Keys Generation:

- Certificates were generated using a keytool or OpenSSL and stored in a `.p12` file. This is configured in `application.properties` to use the `PKCS12` keystore type.

Configuration:

application.properties

```
server.ssl.key-store=classpath:local-ssl.p12
server.ssl.key-store-password=12345678
server.ssl.key-store-type=PKCS12
server.ssl.key-alias=local_ssl
```

Exercise 3: HTTPS Configuration

The application enforces HTTPS by configuring the `SecurityFilterChain` to require secure channels for all requests:

Java code
```
.requiresChannel(channel -> channel.anyRequest().requiresSecure());
```

This ensures all communications are encrypted, preventing man-in-the-middle attacks and ensuring data integrity and confidentiality.

3. Conclusion

The lab successfully demonstrated the integration of Spring Security for both basic and token-based authentication, the setup of PKI for secure communications, and the enforcement of HTTPS for all requests. This comprehensive security setup provides robust protection for web applications against common security threats.