Java SSL with P12 Certificate Project

Saifullah Shaukat

September 25, 2025

Java SSL with P12 Certificate

Java SSL implementation with P12 certificate support for client authentication and SSL proxy functionality.

Features

- **TLS 1.2/1.3** - Modern SSL/TLS protocols
- **P12 Certificates** - PKCS#12 certificate authentication
- **SSL Proxy** - Multi-threaded SSL proxy server
- **Client Auth** - Mutual TLS with client certificates
- **Strong Encryption** - AES-256-GCM cipher suites

Quick Start

Requirements

- Java 8+
- Included P12 certificate: `badssl.com-client.p12` (password: `badssl.com`)

Setup

```
git clone https://github.com
cd java-ssl
./run.sh compile
```

Usage

Easy Commands

```
./run.sh client     # Test SSL client with P12 certificate
./run.sh proxy      # Start SSL proxy server (port 8444)
./run.sh test       # Run proxy tests
./run.sh clean      # Remove compiled files
```

Manual Commands

```
# SSL Client with P12 Certificate
java P12SSLClient
```

```
# SSL Proxy Server
java P12SSLProxy

# Advanced SSL Proxy
java AdvancedSSLProxy

# Basic SSL Proxy
# Basic SSL Proxy
java SSLProxy
```

Project Structure

```
java-ssl/
├── SSLProxy.java              # Basic SSL proxy
├── AdvancedSSLProxy.java      # Advanced SSL proxy with client certs
├── P12SSLProxy.java           # P12 certificate-enabled proxy
├── P12SSLClient.java          # SSL client with P12 authentication
├── ProxyTestClient.java       # Proxy test client
├── ProxyTestSuite.java        # Test suite
├── badssl.com-client.p12      # P12 certificate file
├── run.sh                     # Build and run script
└── README.md                  # This file
```

Code Examples

P12 Certificate Loading

```
KeyStore keyStore = KeyStore.getInstance("PKCS12");
keyStore.load(new FileInputStream("badssl.com-client.p12"),
        "badssl.com".toCharArray());
```

SSL Context Creation

```
SSLContext sslContext = SSLContext.getInstance("TLS");
sslContext.init(keyManagerFactory.getKeyManagers(), trustManagers, new
        SecureRandom());
```

SSL Server Setup

```
SSLServerSocket serverSocket = (SSLServerSocket)
        factory.createServerSocket(8444);
serverSocket.setWantClientAuth(true);
```

Configuration

- **Port**: 8444 (SSL Proxy)
- **Protocols**: TLS 1.2, TLS 1.3
- **Ciphers**: AES-256-GCM, AES-128-GCM
- **Certificate**: BadSSL Client Certificate (valid until 2027)

Testing

```
./run.sh test    # Run proxy tests
./run.sh suite   # Run complete test suite
```

```
##  Technical Implementation

java-ssl/ ├── SSLProxy.java # Basic SSL proxy implementation ├──
AdvancedSSLProxy.java # Advanced SSL proxy with client cert support ├──
P12SSLProxy.java # P12 certificate-enabled SSL proxy ├── P12SSLClient.java # SSL
client with P12 certificate authentication ├── ProxyTestClient.java # Test client for
proxy functionality ├── ProxyTestSuite.java # Comprehensive test suite ├──
badssl.com-client.p12 # P12 certificate file ├── CLIENT_DELIVERY_GUIDE.md #
Client delivery instructions ├── P12_SSL_GUIDE.md # P12 certificate integration
guide ├── PROJECT_OVERVIEW.md # Technical project overview ├──
PROXY_README.md # Proxy-specific documentation └── README.md # Main
project documentation


### Key Features

#### SSL/TLS Implementation
```java
// Create SSL context with P12 certificate
SSLContext sslContext = SSLContext.getInstance("TLS");
sslContext.init(keyManagerFactory.getKeyManagers(), trustManagers, new
SecureRandom());
```

P12 Certificate Loading

```java
// Load P12 keystore
KeyStore keyStore = KeyStore.getInstance("PKCS12");
keyStore.load(new FileInputStream("badssl.com-client.p12"),
         "badssl.com".toCharArray());
```

SSL Proxy Server

```java
// Create SSL server socket with client authentication
SSLServerSocket serverSocket = (SSLServerSocket)
         factory.createServerSocket(8444);
serverSocket.setWantClientAuth(true);
```

Configuration

SSL Settings

- **Default Proxy Port**: 8444
- **Supported Protocols**: TLS 1.2, TLS 1.3
- **Cipher Suites**: AES-256-GCM, AES-128-GCM
- **Client Authentication**: Optional/Required

Certificate Configuration

- **P12 File**: `badssl.com-client.p12`
- **Password**: `badssl.com`
- **Certificate Subject**: CN=BadSSL Client Certificate
- **Validity**: September 24, 2025 - September 24, 2027

Testing

The project includes comprehensive testing:

```
# Test basic proxy functionality
```
```

```
java ProxyTestClient

# Run complete test suite
java ProxyTestSuite
```

Test Results

- SSL handshake validation
- Client certificate authentication
- TLS 1.2/1.3 protocol support
- Strong cipher suites

Security

- **Mutual TLS** - Client & server certificate validation
- **Strong Encryption** - AES-256-GCM cipher suites
- **Secure Storage** - PKCS#12 password-protected certificates

License

MIT License - see LICENSE file for details.

Author

Saifullah Shaukat - Saifullah Shaukat

Secure SSL connections with Java