**CIS 5600 - Information Security Management Project Proposal**

**JAVA Security Basics**

**CIS 5600: Information Security Management**

**Instructor:** Dr. Jeremy Lanman

Bernice Templeman

Btempleman2015@my.fit.edu

# ABSTRACT:

Java is currently one of the most widely used programming languages around the world. With the increase in cyber-attacks, developers will need to protect their code. Different methods will be needed to protect their code. Java has included several security features in its platform. The security features are intended for developers, system administrators and users. Some of the security features include cryptography, authentication and access control, secure communications and public key infrastructure. Many employers are now seeking Java programmers with both knowledge and development experience with Java’s security platform. Cryptography is the most common and effective way to protect information. Cryptography can be used incorrectly resulting in unsecure systems and data. Even with Java’s built-in cryptography features, Java Cryptography Architecture (JCA) and Java Cryptography Extension (JCE), programmers need to know how to use them correctly. Secure Communications are based on cryptography and provides APIs for SSL/TLS. The Public Key Infrastructure (PKI) provides a framework for securing information that is exchanged on an insecure network. Java Authentication and Authorization Service (JAAS) is another Java package that adds to security by authenticating users. In this study, we can study how applications can be protected with the Java Security Platform and create java programs that use some of the basic Java security features. This will enhance our programming skills and prepare us for work as Java developers.

# KEYWORDS:

JAVA, Security, Cryptography, Authentication and Access Control, Public Key Infrastructure.

# SUMMARY:

Realizing the importance of both knowledge and development experience, this project will consist of two parts. The first part of the project will consist of an overview of Java Security. The first goal of the project is study Java’s security platform. This will include reading Java documentation and some tutorials. The plan is to focus mainly on cryptography (JCA & JCE), PKI and JAAS. The second part of the project will include creating programs using some of Java’s basic security features.

# MOTIVATION OF THE PROPOSED PROBLEM:

Java is one of the most popular program languages worldwide. Java security plays a valuable role as organizations worldwide are increasing their reliance on software to protect their systems and data. Java provides a variety of security APIs but programmers must know how to use them. Programmers with knowledge and experience with the Java security features are currently valuable to the worldwide job market.

# MAJOR ISSUES:

Understanding security concepts and how to use the Java APIs are important to protecting data. Using available documentation and tutorials will help programmers to learn how to protect the systems and data with the Java security platform.

## Platform Security:

There are several built-in security features in Java such as:

* Strong data typing
* Automatic memory management
* Bytecode verification
* Secure class loading

These provide a safe and secure platform for applications and data.

## Cryptography:

Java Cryptography is part of Java’s built-in security designed to protect programs and data from malicious programs. Java now includes both the Java Cryptography Architecture (JCA) and the Java Cryptography Extension (JCE) packages. Cryptography is the most common and effective way to protect information by encoding it. Cryptography is based on a cipher algorithm which transforms plain text to cipher text. A cipher key(s) is used to encrypt and decrypt the text. This requires a key exchange between the sender and receiver of the data. Even though cryptography is the most common and effective way to protect data, Cryptography can be used incorrectly resulting in unsecure systems and data. Even with Java’s built-in cryptography features, programmers need to know how to use them correctly.

## Authentication and Access Control:

The Java Authentication and Authorization Service (JAAS) includes:

* for *authentication* of users, to reliably and securely determine who is currently executing Java code, regardless of whether the code is running as an application, an applet, a bean, or a servlet
* for *authorization* of users to ensure they have the access control rights (permissions) required to do the actions performed.

## Secure Communications:

Java includes APIs to perform secure Communications. These are based on cryptography.

## Public Key Infrastructure:

## Java’s PKI includes tools to manage keys and certificates. It also supports:

## Certificates and Revocation Lists

## Certification Path Validators and Builders

## KeyStores

## Certificate Stores (Repositories)

# JUSTIFICATION:

It is important for programmers to know how to protect code, systems and data from malicious programs. Java has built-in security features but it is the programmer’s responsibility to learn them and use them correctly.

# ARCHITECTURE:

# The Java programs will be developed and tested on a Windows platform using NetBeans.

# PROPOSED TIMELINE:

|  |  |
| --- | --- |
| Task  Proposal Completed | Target Date Completed Date  5/28/05 |
| Proposal Approved | 10/31/2016 |
| Background Report Completed | 10/31/2016 |
| Tutorials and Design Completed | 11/14/2016 |
| Program Completed | 11/21/2016 |
| Testing of Program Completed | 11/21/2016 |
| Final Write Up and Summary Complete | 11/21/2016 |
| 15-20 minute YouTube Video summary and demonstration | 11/21/2016 |

# DELIVERABLES:

Final Report:

* Introduction
  + Discuss the background
  + Summarize the basis of your implementation
  + Summarize the design of your implementation.
  + Summarize the validation of your implementation (e.g., test, demonstration)
* Implementation details
  + Architecture description (not required for implementation of crypto algorithms).
  + Description of the basis of your implementation (e.g., algorithms, protocols, etc.)
  + Description of your implementation.
* Testing (or performance evaluation if you implemented crypto algorithms)
  + Description of the testing (or evaluation) environments
  + Description of the testing (or evaluation) process
  + Description of the testing (or evaluation) results
* Conclusions and possible future work
  + Summarize the conclusions of your survey.
* References
  + List all the citations referenced in your paper. You will lose points for each dangling reference (i.e., the reference not cited in the main text).
* 15-20 minute presentation to the class

# REFERENCES:

Workman, Michael; Phelps, Daniel; Gathegi, John. *Information Security for Managers.* Jones & Bartlett Learning, LLC. 2013.

Lokhande, P.S.; Meshram, B.B.. *E-Commerce Applications: Vulnerabilities, Attacks and Countermeasures.* February 2013[. http://ijarcet.org/wp-content/uploads/IJARCET-VOL-2-ISSUE-2-499-509.pdf](.%20http:/ijarcet.org/wp-content/uploads/IJARCET-VOL-2-ISSUE-2-499-509.pdf)

Java Security. <http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136007.html>

*Java Cryptography Architecture (JCA) Reference Guide*. <http://docs.oracle.com/javase/8/docs/technotes/guides/security/crypto/CryptoSpec.html>

Rubin, Brad. *Java security: Java security, Part 1: Crypto basics.* July 2002. [http://www.ibm.com/developerworks/java/tutorials/j-sec1/j-sec1.html](http://www.ibm.com/developerworks/java/tutorials/j-sec1/j-sec1.html%20)

Rubin, Brad. *Java security: Java security, Part 2: Authentication and Authorization.* July 2002. <http://www.ibm.com/developerworks/java/tutorials/j-sec2/j-sec2.html>

Java Security Overview. <https://docs.oracle.com/javase/8/docs/technotes/guides/security/overview/jsoverview.html>

JAAS Authorization Tutorial. <http://docs.oracle.com/javase/7/docs/technotes/guides/security/jaas/tutorials/GeneralAcnAndAzn.html>

Part II: Secure Communications using the Java SE Security API. <http://docs.oracle.com/javase/8/docs/technotes/guides/security/jgss/lab/part2.html>

Java PKI Programmer’s Guide. <http://docs.oracle.com/javase/7/docs/technotes/guides/security/certpath/CertPathProgGuide.html>