CS 4480
PA 3 A - Design Document
Rob Johansen
u0531837

I will be using Java's built-in cryptography libraries contained in the javax.crypto and java.security packages to implement my secure messaging application for PA 3.

Before I begin writing any code, I will use openssl commands to generate keypairs for Alice, Bob, and the Certificate Authority. For example:

```
openssl genrsa -out aliceprivate.pem 1024
openssl rsa -in aliceprivate.pem -pubout > alicepublic.pem
```

I will then begin Bob's application by designing it to accept a connection from Alice's application on a port allowed by CADE lab machines (in the 6000-8000 range, I believe).

I will then design Alice's application to do the following:

1. Accept Bob's IP address as input using the -a option.
2. Prompt the user to provide a message on the command line.
3. Generate a signature of the input message using the java.security.Signature class.
4. Append the signature to the input message.
5. Generate a 3DES symmetric key using the javax.crypto.Cipher class and use it to encrypt the signed input message.
6. Read Bob's public key from the file system and verify it using the Certificate Authority public key (both of these keys will be provided in the tar ball for Alice's application).
7. Encrypt the 3DES symmetric key to Bob's public key.
8. Connect to the IP address passed in via -a and send the encrypted message and symmetric key to Bob.
9. Prompt the user for another message to send.

Finally, I will finish Bob's application so that it can decrypt the message using his private key, verify the signature using Alice's public key (using the same Java libraries), and print to stdout the cleartext message sent by Alice's program.

I will also design both programs to support the -v option for displaying verbose output of each step in the process, and the -h option for printing out a help message.