**Rob Johansen**

**u0531837**

**CS 4480 - PA 3 - Assignment Report**



***Design***

My implementation of the secure messaging application for PA 3 is written in Java. It is comprised of a CipherTalk class for Alice, and a CipherListen class for Bob. To send a secure message from Alice to Bob, you first execute Bob's program and specify the TCP port on which it will listen for a secure message from Alice. You then execute Alice's program, specifying Bob's IP address, the TCP port on which Bob's program is listening, and the message to deliver securely to Bob.

Both programs also support options for displaying verbose output and help messages which describe in detail how to use the programs. I highly recommend displaying verbose output.

One of my design tradeoffs is that Alice can only send Bob one secure message per program execution. After each secure message, Bob and Alice must both execute their programs again in order to send another secure message (this was done purely to decrease development time).

***Testing***

Most of my testing consisted of sending secure messages from Alice to bob from my MacBook Pro to my Windows computer, and between two different CADE machines.

However, since these programs depend on various files (e.g. public and private key files), I also tested without those files present to ensure that my programs would catch exceptions and display helpful error messages.

***Output***

Below is the output from my programs on two computers. This is the entire secure message exchange between Alice and Bob when verbose output is being displayed. The bold red text highlights where the output was obtained:

***On Bob's Computer:***

***On Alice's Computer:***

java CipherTalk -a 192.168.1.11 -p 6000 -m 'This is secure!' -v

========================================

Starting CipherTalk Transmission

========================================

The message from Alice to Bob is:

This is secure!

Creating a SHA-1 message digest

The message digest in hex is:

d165c26625b9937279fafe31d5ed1184c793777

Alice's private key in hex is:



Signing the message digest using Alice's private key

Alice's signature of the message digest in hex is:

66a4161751d4ee3432d8f8a6739981698c578b9e47959d8eda420de601219d3a5a72d2e2d77e4a455ebfcdd19af6aa4ef47c123c4edbd28cffed4207a244793b1cf0703c4252e18349c54c481a038924f7413d7af10f64725aea34e50bbc34eac4ed60f8ae29881dfcfc295eec7f26a7a4f45370c9f16776d48834a420

Loading the CA's public key

The CA's public key in hex is:

30819f30d692a864886f7d111503818d0308189281810eb3665f4b73cc413e7b0dc4d37588b5bdc6835a302fc0648db09dd2cd7313f6c66f12787b9f0751a856efa551b5bb7a2302e296d32bdd28ae4a86588fe5e7a10bfaa62520ae78b3af4229e2122fad71431a1df64ab7ede28761cf218f368123d296ca479ad1364afb0c3b236dc6a45fdb5dfe29ebf393c5aede723101

Loading the CA's signature of Bob's public key

The CA's signature of Bob's public key in hex is:

33f8ccd8c47386c56f6a88930609928852d1f7a9747d88e6e9692897944370152df95fc478f96154da135c638a51b95ca6c954cba79ada5eca4f868d5c3cbca551a3811b31d65fcea163c31cc5a5a2a23e72176eddfc6716a24fa58025718af8a2ddeb12b8010f9958323254d21d6b9c7d416eff9c93a8dd9cd11c7

Loading Bob's public key

Bob's public key in hex is:

30819f30d692a864886f7d111503818d0308189281810c46a31e1a117058b571d2953ad43d49cd4efbad2f10ab99fac82bd571e7b870ee35e31c2a1588aea1eeb9bd2ac88e72558867b463dc243feaa981187f097643d539c64a7ff8acf17d2070191da618f89bf295c733963cb5146b55833764a5aff7fe29bd4db6e53dc1c2bedcbd487d368e8c38648ad3054cc75afd723101

Verifying the CA signature on Bob's public key

Bob's public key is verified by the CA

Generating a 3DES symmetric key

The unencrypted 3DES key in hex is:

2fb0cb6edf7a70497c83fb46514c1071c1ccecda1941c85

Encrypting the hash, signature, and message to the 3DES key

The ciphertext in hex is:

a756a47576c3666c8ee239e914fa3db3a5cc2a5a88afb6e9353a6dcf2b4521f685a1c9bb2e1aaa8e69920bd5568a90b4ae239550677218bbc2d7983b488ce67f236585c673b72c58979d14938dae60c4537d4c81b3bf3930be78f95d974429dd9e6a2d05a9de8dcfe9db0b97e3d19f72cee7b4ecc740454665f792cbfff1ad5654b57fb9cd357f64cb9c186a9d1807a65c57a9829ba512391d3728ea3b6b1121f5ad03e

Encrypting the 3DES key to Bob's public key

The encrypted 3DES key in hex is:

3a20a07d54601bc965284d72a75c3e8d81b8657b19bf45fc3ff9d3a9728d822abe416f4032d5c1a8cd3ecb3ca4e8496214ecafe5656f11af609019cd772b7b42c813d9d3e814a9b68c98a6ae9eb8272cd1b39618874a6eafec23ad8055b4bacbbe2bfb11f91fa710eaf6415231d62e5db3199f6a1a7b9c8d226c1

Combining all the ciphertext bytes

Sending the ciphertext to Bob...