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:

```
java CipherListen -p 6000 -v

============================
    Starting CipherListen
============================

Waiting for a secure message from Alice...
```

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:
308227821030d692a864886f7d11150482262308225e210281810c0b8e4331aedfc32f5
14656a588bae86e8ffe5924d18356c6ce0e91fe65e8ea3fc3327a5eb8f88e65063673f8
2b4237cf1382982cb2ef2226e831d287e99b11f5fb59e937f554d13ed17e6baa15eabe4
bdfe6edae173a809ba1aa185939ca06ac1c3165f19bd1ad6239a33ee77f5d3d7acf5976
713d3aa46b9c874ba7b23101281810ae6e6a34f46082d4c4087be66ebafb98e68133d11
704352589594cb7a22beea694d25ced81238279dee944043b51f1fe11a494487c094c97
ba8877f5699059922741ae1c9f464385d84357ac5c8fff863497cc2722ed4a2c42daff0
cee1f946317613ddca0ebfb65b38c62d7f86533fb2ff7d1141a167aa8112410debfd18e
f9c34ca9ae0e9f3ea1baf33de6a5191687ac58555e1e80409b79b9cd3e7bbe16a36e48b
ce920ebcd2792e2561d7342a0b4ab81edecb87d99c09ec32410dd7d9d75d2ef6efd7582
686e7fb096355b807648c32333c5b1b6c4a86ce5a347b3c7a4797ca8c67a89b63a1a1f4
57f178d5cd288babb9d2724528fd29e9240522040fff2ab721f5d191dc8eb7735d966f2
39ded51ab2a562dc2fbcb3955f3cecfa639f4333807edaffcd3e83465af68e2f8f867a4
029863e8458ec1a1724109d247ee177a2577e885e974c428183a6be8a81d4ea0d09bedb
ad42ef378cf60e022632b838a7307245fc2cb2d5d6f7b99d8eb3228d1ed4895342ef62e
924108825c65d1278aa806fb0bf7ff985f09fcf32f52b16ea69a5bcbf8a1c6bde853134
9fe7be67a18560b19e185227d4dd395e423bde2bdfd03ca8d50a95a9963
```

```
Signing the message digest using Alice's private key

Alice's signature of the message digest in hex is:
66a4161751d4ee3432d8f8a6739981698c578b9e47959d8eda420de601219d3a5a72d2e
2d77e4a455ebfcdd19af6aa4ef47c123c4edbd28cffed4207a244793b1cf0703c4252e1
8349c54c481a038924f7413d7af10f64725aea34e50bbc34eac4ed60f8ae29881dfcfc2
95eec7f26a7a4f45370c9f16776d48834a420

Loading the CA's public key

The CA's public key in hex is:
30819f30d692a864886f7d111503818d0308189281810eb3665f4b73cc413e7b0dc4d37
588b5bdc6835a302fc0648db09dd2cd7313f6c66f12787b9f0751a856efa551b5bb7a23
02e296d32bdd28ae4a86588fe5e7a10bfaa62520ae78b3af4229e2122fad71431a1df64
ab7ede28761cf218f368123d296ca479ad1364afb0c3b236dc6a45fdb5dfe29ebf393c5
aede723101

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

The CA's signature of Bob's public key in hex is:
33f8ccd8c47386c56f6a88930609928852d1f7a9747d88e6e9692897944370152df95fc
478f96154da135c638a51b95ca6c954cba79ada5eca4f868d5c3cbca551a3811b31d65f
cea163c31cc5a5a2a23e72176eddfc6716a24fa58025718af8a2ddeb12b8010f9958323
254d21d6b9c7d416eff9c93a8dd9cd11c7

Loading Bob's public key

Bob's public key in hex is:
30819f30d692a864886f7d111503818d0308189281810c46a31e1a117058b571d2953ad
43d49cd4efbad2f10ab99fac82bd571e7b870ee35e31c2a1588aea1eeb9bd2ac88e7255
8867b463dc243feaa981187f097643d539c64a7ff8acf17d2070191da618f89bf295c73
3963cb5146b55833764a5aff7fe29bd4db6e53dc1c2bedcbd487d368e8c38648ad3054c
c75afd723101

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:
```

a756a47576c3666c8ee239e914fa3db3a5cc2a5a88afb6e9353a6dcf2b4521f685a1c9b
b2e1aaa8e69920bd5568a90b4ae239550677218bbc2d7983b488ce67f236585c673b72c
58979d14938dae60c4537d4c81b3bf3930be78f95d974429dd9e6a2d05a9de8dcfe9db0
b97e3d19f72cee7b4ecc740454665f792cbfff1ad5654b57fb9cd357f64cb9c186a9d18
07a65c57a9829ba512391d3728ea3b6b1121f5ad03e

Encrypting the 3DES key to Bob's public key

The encrypted 3DES key in hex is:
3a20a07d54601bc965284d72a75c3e8d81b8657b19bf45fc3ff9d3a9728d822abe416f4
032d5c1a8cd3ecb3ca4e8496214ecafe5656f11af609019cd772b7b42c813d9d3e814a9
b68c98a6ae9eb8272cd1b39618874a6eafec23ad8055b4bacbbe2bfb11f91fa710eaf64
15231d62e5db3199f6a1a7b9c8d226c1

Combining all the ciphertext bytes
Sending the ciphertext to Bob...

On Bob's Computer:

Now receiving a secure message from Alice

The encrypted 3DES key in hex is:
af528ea9861f86797f6067782ff3d8e3cc50f580ebf6cb1dbc97775fcd714cc81443d05
97985c8587744534890938c1d59fbb211ab038ccf51b263167fa1884bec
7a75e3478bff496c16adbf571326d55eb923720afde93365d903bee6f84f1fd4174edc8
0bcea870179dba29fb3d2ca777242e7ec5e9397d6d14fa80

Decrypting the 3DES key with Bob's private key

Bob's private key in hex is:
308227721030d692a864886f7d11150482261308225d210281810c46a31e1a117058b57
1d2953ad43d49cd4efbad2f10ab99fac82bd571e7b870ee35e31c2a1588
aea1eeb9bd2ac88e72558867b463dc243feaa981187f097643d539c64a7ff8acf17d207
0191da618f89bf295c733963cb5146b55833764a5aff7fe29bd4db6e53d
c1c2bedcbd487d368e8c38648ad3054cc75afd72310128181090da92547eba68c75bc6d
ecba52d52198d4f8568b985a4f8d3d9a7ffea62173f4714a877cdbd1a6f
155a05245d673a12e09357b942653d12f935aca8de2cae813a7cb2ab90e1357176ebe97
d16b238fa9c15b3bdf1bdad778cc24c52c686c36317af269937e49df85c
5c839fb91dbc937719e942f2787de69a5f25092410f12c2fd6f58adf27331a375b7362b
1d5484aea81a920f839017336811239b31af3318e2cd24b07eef924a2cb
d61b2fcff88bd78feab7ce37f1a4cb1eb6b2410d07d5f2f70adee66840652c567209a63
954c535be20665931647e5e8da0b4b6a794b0f19dc4af4f6061cfba7e47
e2587b75c77c8e4ed3f19afa85d6eb445240219fa877cea7cc209e40d06a4c5c84f341e
251bb6c2eafa54784848d40be7717a1367d84b9572128415c62eceb2589
9acbe2fa142e84bc11e861c317e8424b24011414f0f6e13f7171ab7b9a7e1bfadad197d

```
1d545a70e0873be0d7765e1571386eb1779e7c4cf7d924c8614b93ed9cc
d8eaf687a87f7dab1158838bb6d2410d0478f654a19f4f2a77bff7ccce9e729536e593e
f6ffb39045c6288effdb627b3d04e15479d4d12b34917b71ef67540a5c8
13fd07e178af7328d48


The decrypted 3DES key in hex is:
cb804f76a251e3883891f61e0d57f25e04dc75b551c846

Decrypting the ciphertext using the 3DES key

The ciphertext in hex is:
f1a2be9c41c68d56b93c7a729198b59d69272a3c275a238e38352d750e8ec9b09d3188b
f781d565d655450dbbed351921c259652cdad7b73aa2b7439f93734bc8a
89995935b4b5d1de8d7fd1089c07a939a1dec982d5dc7d6fafdf154b69e563472ee6d79
6dbaf4b1ebc36256ad0d873e1977164382fa191eaa13e4d8f768457b543
e6a1f3e9cb1c22f83aee4f4d652d6124c57540aff6d2f4c1beabf232f287ff5fe3

Checking the signature of the message.

Alice's public key in hex is:
30819f30d692a864886f7d111503818d0308189281810c0b8e4331aedfc32f514656a58
8bae86e8ffe5924d18356c6ce0e91fe65e8ea3fc3327a5eb8f88e650636
73f82b4237cf1382982cb2ef2226e831d287e99b11f5fb59e937f554d13ed17e6baa15e
abe4bdfe6edae173a809ba1aa185939ca06ac1c3165f19bd1ad6239a33e
e77f5d3d7acf5976713d3aa46b9c874ba7b23101


Alice's signature of the message digest in hex is:
66a4161751d4ee3432d8f8a6739981698c578b9e47959d8eda420de601219d3a5a72d2e
2d77e4a455ebfcdd19af6aa4ef47c123c4edbd28cffed4207a244793b1c
f0703c4252e18349c54c481a038924f7413d7af10f64725aea34e50bbc34eac4ed60f8a
e29881dfcfc295eec7f26a7a4f45370c9f16776d48834a420

The signature is verified. This message originated from Alice.

The message digest in hex is:
d165c26625b9937279fafe31d5ed1184c793777

Checking the integrity of the message.
The message digest is correct. This message was not altered in transit.

The plaintext message from Alice is:

This is secure!
```