Overview

The purpose of this document is to show the different unit and integration tests run on the system. Tests will be broken down into the function, a description of what was tested, the expected result, and lastly an image of the actual result.

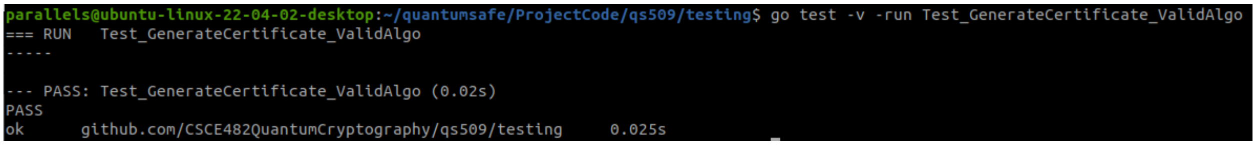
Unit Tests

Test_GenerateCertificate_ValidAlgo

Description: Generate a certificate with a signing algorithm supported by qs509

Expected Result: No error, certificate generated, key generated

Actual Result:



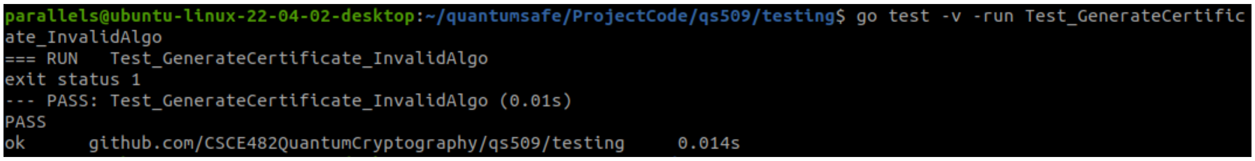
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_GenerateCertificate_ValidAlgo
=== RUN   Test_GenerateCertificate_ValidAlgo
-----

--- PASS: Test_GenerateCertificate_ValidAlgo (0.02s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.025s
```

Test_GenerateCertificate_InvalidAlgo

Description: Generate a certificate with a signing algorithm *not* supported by qs509

Expected Result: Throws error, no certificate generated, no key generated

Actual Result:



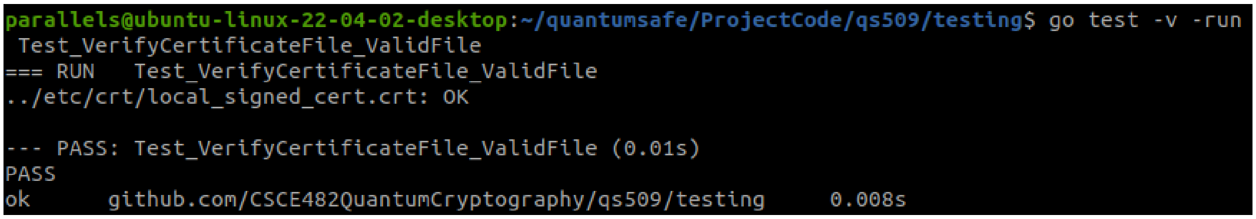
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_GenerateCertific
ate_InvalidAlgo
=== RUN   Test_GenerateCertificate_InvalidAlgo
exit status 1
--- PASS: Test_GenerateCertificate_InvalidAlgo (0.01s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.014s
```

Test_VerifyCertificateFile_ValidFile

Description: Verify a certificate was signed by a CA, using a valid certificate

Expected Result: Result is true

Actual Result:



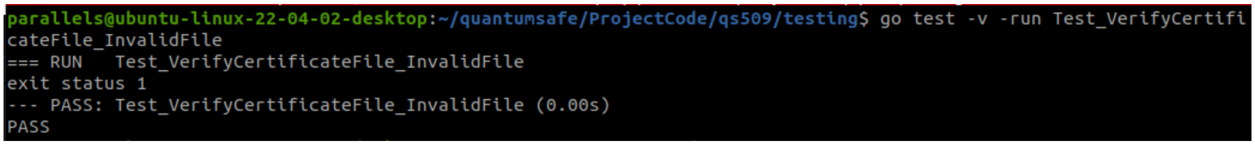
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run
 Test_VerifyCertificateFile_ValidFile
=== RUN   Test_VerifyCertificateFile_ValidFile
../etc/crt/local_signed_cert.crt: OK

--- PASS: Test_VerifyCertificateFile_ValidFile (0.01s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.008s
```

Test_VerifyCertificateFile_InvalidFile

Description: Verify a certificate was signed by a CA, using a non-existent certificate

Expected Result: Error thrown, returns false

Actual Result:



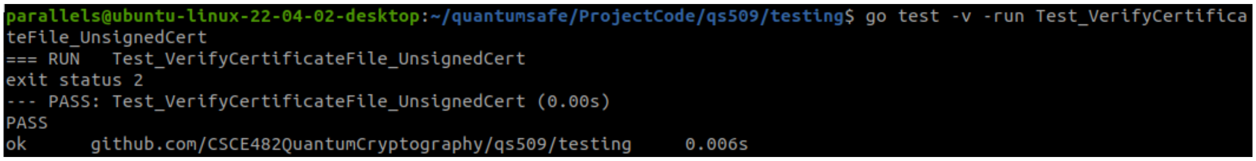
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_VerifyCertifi
cateFile_InvalidFile
=== RUN   Test_VerifyCertificateFile_InvalidFile
exit status 1
--- PASS: Test_VerifyCertificateFile_InvalidFile (0.00s)
PASS
```

Test_VerifyCertificateFile_UnsignedFile

Description:Verify a certificate was signed by a CA, using a certificate that wasn't signed

Expected Result: Error thrown, return false

Actual Result:



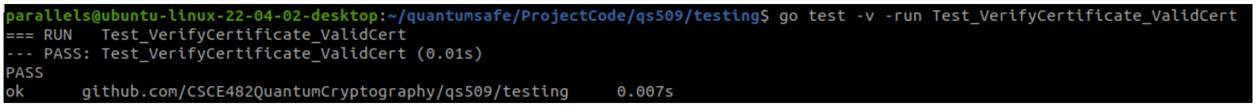
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_VerifyCertifica
teFile_UnsignedCert
=== RUN   Test_VerifyCertificateFile_UnsignedCert
exit status 2
--- PASS: Test_VerifyCertificateFile_UnsignedCert (0.00s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.006s
```

Test_VerifyCertificate_ValidCert

Description: Verify a certificate's bytes were signed by a CA, using a certificate that was signed

Expected Result: True

Actual Result:



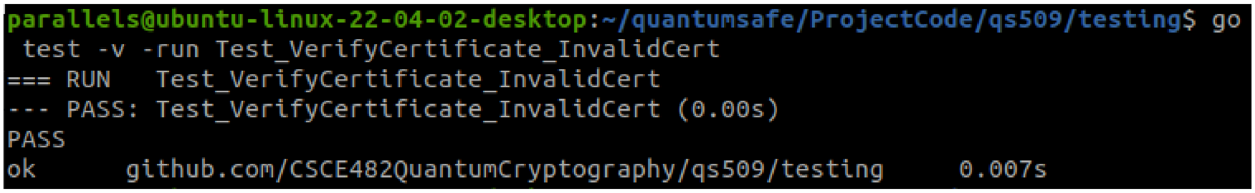
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_VerifyCertificate_ValidCert
=== RUN   Test_VerifyCertificate_ValidCert
--- PASS: Test_VerifyCertificate_ValidCert (0.01s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.007s
```

Test_VerifyCertificate_InvalidCert

Description: Verify a certificate's bytes were signed by a CA, using a certificate that wasn't signed

Expected Result: Error thrown, return false

Actual Result:



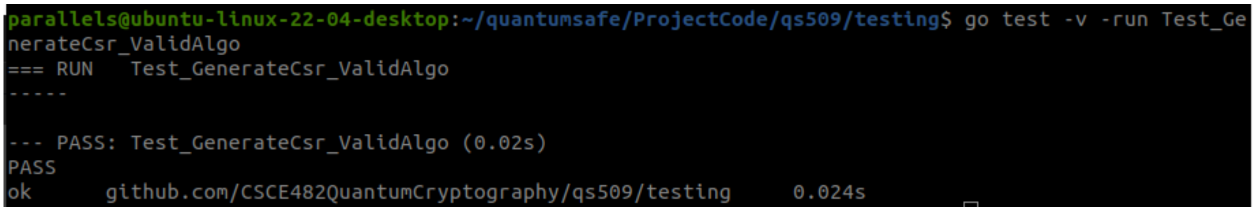
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go
 test -v -run Test_VerifyCertificate_InvalidCert
=== RUN   Test_VerifyCertificate_InvalidCert
--- PASS: Test_VerifyCertificate_InvalidCert (0.00s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.007s
```

Test_GenerateCsr_ValidAlgo

Description: Generate a CSR using a supported algorithm

Expected Result: No error, return true, csr file generated, key file generated

Actual Result:



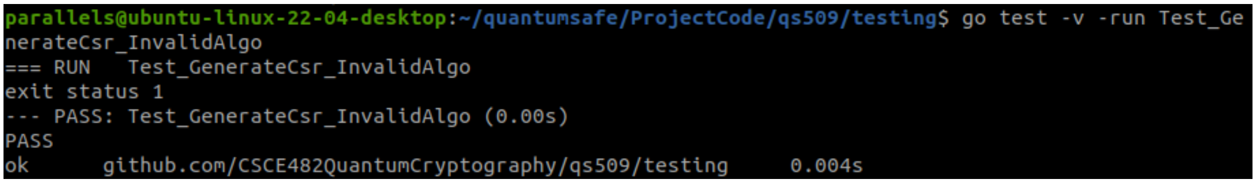
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Ge
nerateCsr_ValidAlgo
=== RUN   Test_GenerateCsr_ValidAlgo
-----

--- PASS: Test_GenerateCsr_ValidAlgo (0.02s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.024s
```

Test_GenerateCsr_InvalidAlgo

Description: Generate a CSR using a non-supported algorithm

Expected Result: Throw error, return false, no csr or key file generated

Actual Result:



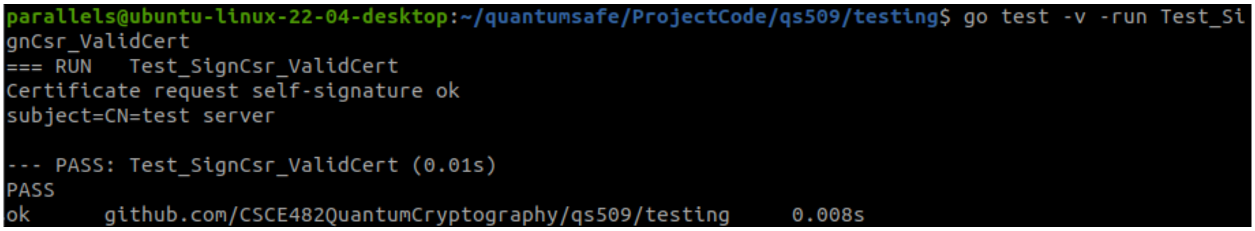
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Ge
nerateCsr_InvalidAlgo
=== RUN   Test_GenerateCsr_InvalidAlgo
exit status 1
--- PASS: Test_GenerateCsr_InvalidAlgo (0.00s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.004s
```

Test_SignCsr_ValidCert

Description: Take a valid CSR file and sign it with local CA

Expected Result: No error, return true, crt file generated

Actual Result:



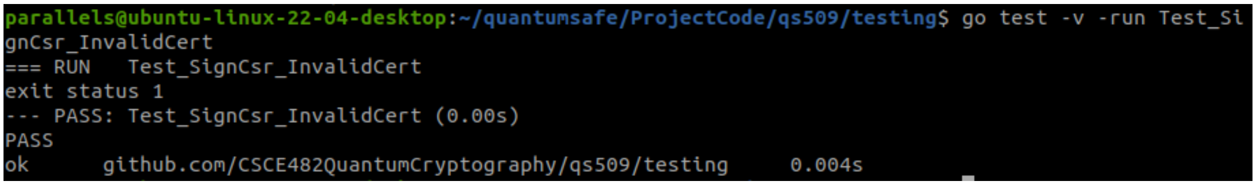
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Si
gnCsr_ValidCert
=== RUN   Test_SignCsr_ValidCert
Certificate request self-signature ok
subject=CN=test server

--- PASS: Test_SignCsr_ValidCert (0.01s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.008s
```

Test_SignCsr_InvalidCert

Description: Take an invalid csr and sign it with local CA

Expected Result: Throw error, return false, no certificate generated

Actual Result:



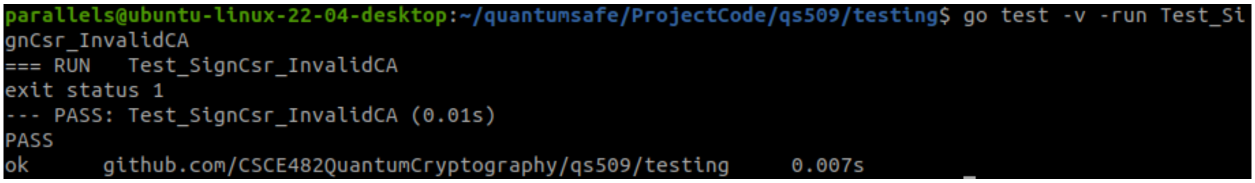
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Si
gnCsr_InvalidCert
=== RUN   Test_SignCsr_InvalidCert
exit status 1
--- PASS: Test_SignCsr_InvalidCert (0.00s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.004s
```

Test_SignCsr_InvalidCA

Description: Take a valid CSR and sign it with an invalid CA

Expected Result: Throw error, return false, no certificate generated

Actual Result:



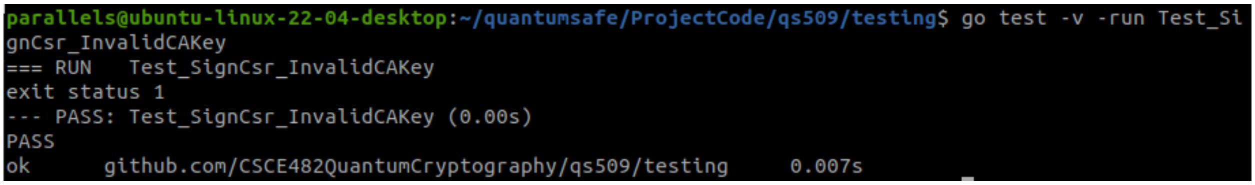
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Si
gnCsr_InvalidCA
=== RUN   Test_SignCsr_InvalidCA
exit status 1
--- PASS: Test_SignCsr_InvalidCA (0.01s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.007s
```

Test_SignCsr_InvalidCAKey

Description: Take a valid CSR and sign it with a valid CA, but with invalid CA key

Expected Result: Throw error, return false, no certificate generated

Actual Result:



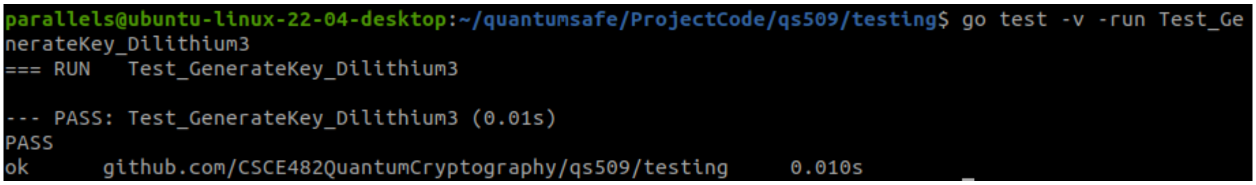
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Si
gnCsr_InvalidCAKey
=== RUN   Test_SignCsr_InvalidCAKey
exit status 1
--- PASS: Test_SignCsr_InvalidCAKey (0.00s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.007s
```

Test_GenerateKey_Dilithium3

Description: Generate a private key with dilithium 3 algorithm

Expected Result: return true, key file generated

Actual Result:



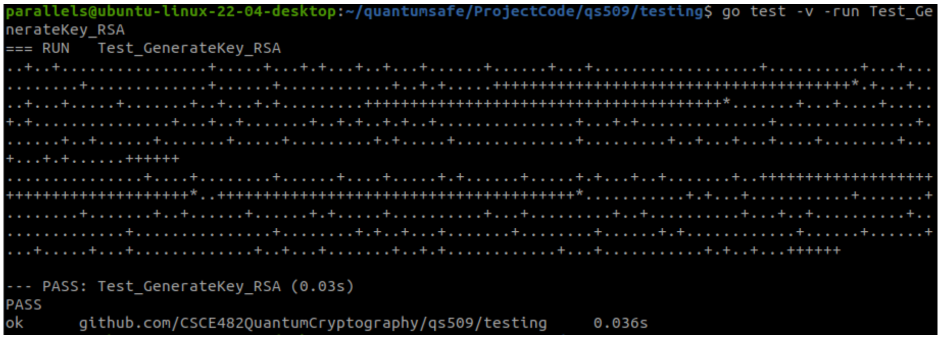
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Ge
nerateKey_Dilithium3
=== RUN   Test_GenerateKey_Dilithium3

--- PASS: Test_GenerateKey_Dilithium3 (0.01s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.010s
```

Test_GenerateKey_RSA

Description: Generate a private key with RSA algorithm

Expected Result: return true, key file generated

Actual Result:



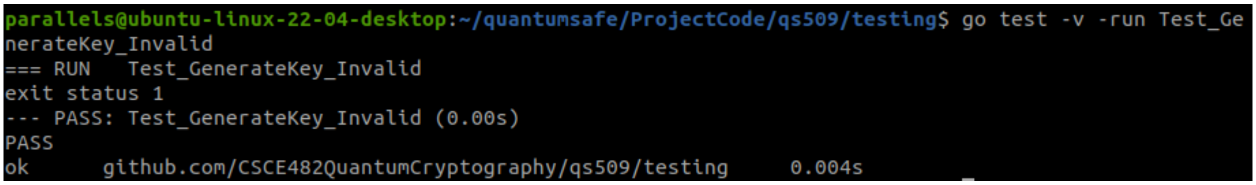
```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Ge
nerateKey_RSA
=== RUN   Test_GenerateKey_RSA
..+..+................+.....+...+.+...+..+...+......+......+...+..................+..........+...+...
........+.............+......+............+..+.+.....+++++++++++++++++++++++++++++++++++++++*.+...+..
..+...+.....+.......+..+...+.+.........+++++++++++++++++++++++++++++++++++++++*.......+...+....+.....
+.+...............+...+..+.......+..+.+..+.+..+...............+...+.+..............+...............+.
......+..+......+.......+.....+.........+.+.....+.............+.........+..+...+...+....+........+...
+...+.+......++++++
...............+....+........+......+....+.....+.+......+.....+.+...+..+.......+..+++++++++++++++++++
++++++++++++++++++++*..+++++++++++++++++++++++++++++++++++++++*...........+.+...+...........+.......+
........+.......+..+......+......+.+.....+..........+...+.........+..+..........+...+..+..........+..
.............+...............+........+.+..+...+.......+........+......+.+............+......+......+
...+.....+...+.............+..+...+.......+..+.+............+...+...........+.+..+...++++++

--- PASS: Test_GenerateKey_RSA (0.03s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.036s
```

Test_GenerateKey_Invalid

Description: Generate a private key with an invalid algorithm

Expected Result: return false, key file *not* generated

Actual Result:



```
parallels@ubuntu-linux-22-04-desktop:~/quantumsafe/ProjectCode/qs509/testing$ go test -v -run Test_Ge
nerateKey_Invalid
=== RUN   Test_GenerateKey_Invalid
exit status 1
--- PASS: Test_GenerateKey_Invalid (0.00s)
PASS
ok      github.com/CSCE482QuantumCryptography/qs509/testing     0.004s
```

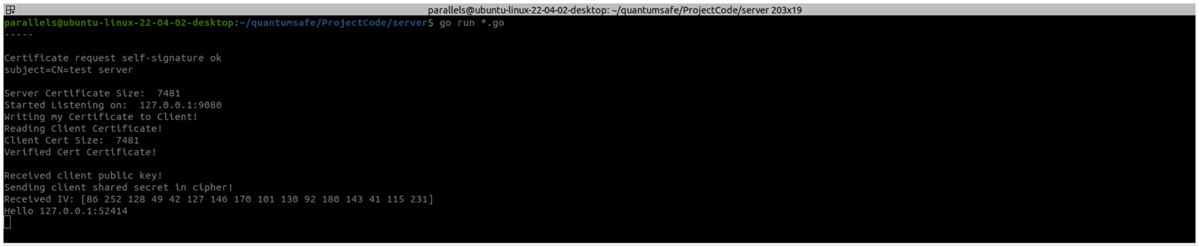
Integration Tests

Dilithium3 SA, Kyber512 KA, Valid CA

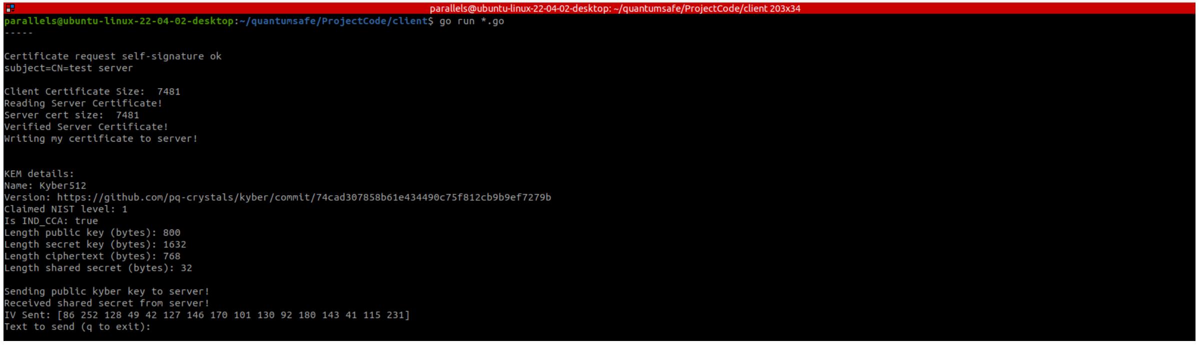
Description: Open client and server with same sa, ka, and ca

Expected Result: Client and server authenticate, exchange secret, and tunnel is formed

Actual Result:


parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x19

```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go
-----

Certificate request self-signature ok
subject=CN=test server

Server Certificate Size:  7481
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
Client Cert Size:  7481
Verified Cert Certificate!

Received client public key!
Sending client shared secret in cipher!
Received IV: [86 252 128 49 42 127 146 170 101 130 92 180 143 41 115 231]
Hello 127.0.0.1:52414
```


parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x34

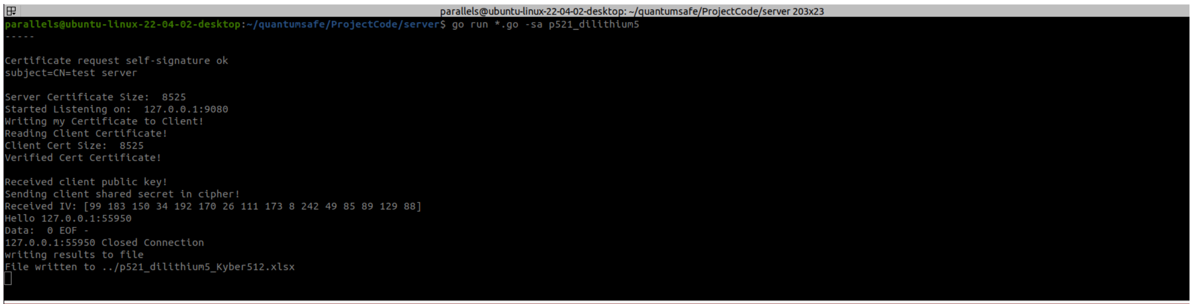
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go
-----

Certificate request self-signature ok
subject=CN=test server

Client Certificate Size:  7481
Reading Server Certificate!
Server cert size:  7481
Verified Server Certificate!
Writing my certificate to server!


KEM details:
Name: Kyber512
Version: https://github.com/pq-crystals/kyber/commit/74cad307858b61e434490c75f812cb9b9ef7279b
Claimed NIST level: 1
Is IND_CCA: true
Length public key (bytes): 800
Length secret key (bytes): 1632
Length ciphertext (bytes): 768
Length shared secret (bytes): 32

Sending public kyber key to server!
Received shared secret from server!
IV Sent: [86 252 128 49 42 127 146 170 101 130 92 180 143 41 115 231]
Text to send (q to exit):
```

P521_Dilithium5 SA, Kyber512 KA, Valid CA

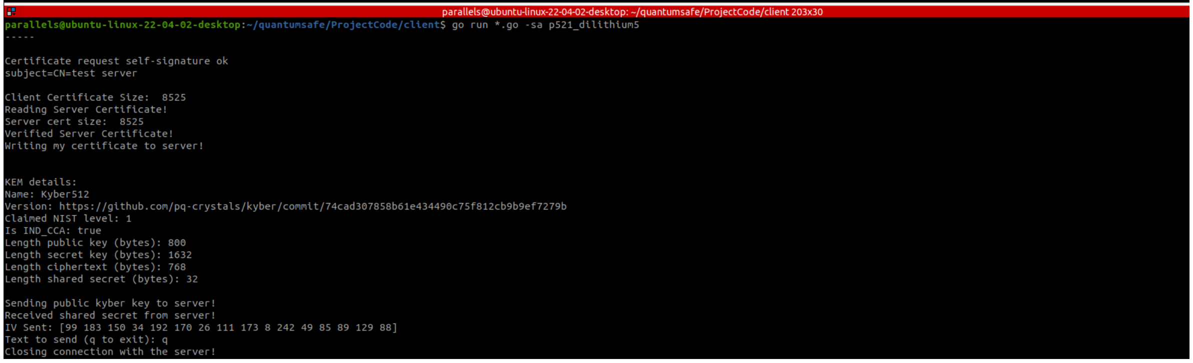
Description: Open client and server with same sa, ka, and ca

Expected Result: Client and server authenticate, exchange secret, and tunnel is formed

Actual Result:


parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x23

```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go -sa p521_dilithium5
-----

Certificate request self-signature ok
subject=CN=test server

Server Certificate Size:  8525
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
Client Cert Size:  8525
Verified Cert Certificate!

Received client public key!
Sending client shared secret in cipher!
Received IV: [99 183 150 34 192 170 26 111 173 8 242 49 85 89 129 88]
Hello 127.0.0.1:55950
Data:  0 EOF -
127.0.0.1:55950 Closed Connection
writing results to file
File written to ../p521_dilithium5_Kyber512.xlsx
```


parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x30

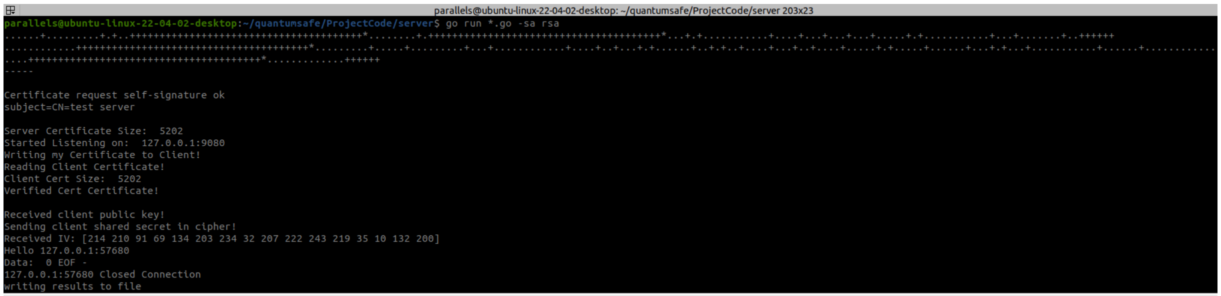
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go -sa p521_dilithium5
-----

Certificate request self-signature ok
subject=CN=test server

Client Certificate Size:  8525
Reading Server Certificate!
Server cert size:  8525
Verified Server Certificate!
Writing my certificate to server!


KEM details:
Name: Kyber512
Version: https://github.com/pq-crystals/kyber/commit/74cad307858b61e434490c75f812cb9b9ef7279b
Claimed NIST level: 1
Is IND_CCA: true
Length public key (bytes): 800
Length secret key (bytes): 1632
Length ciphertext (bytes): 768
Length shared secret (bytes): 32

Sending public kyber key to server!
Received shared secret from server!
IV Sent: [99 183 150 34 192 170 26 111 173 8 242 49 85 89 129 88]
Text to send (q to exit): q
Closing connection with the server!
```

RSA SA, Kyber512 KA, Valid CA

Description: Open client and server with same sa, ka, and ca

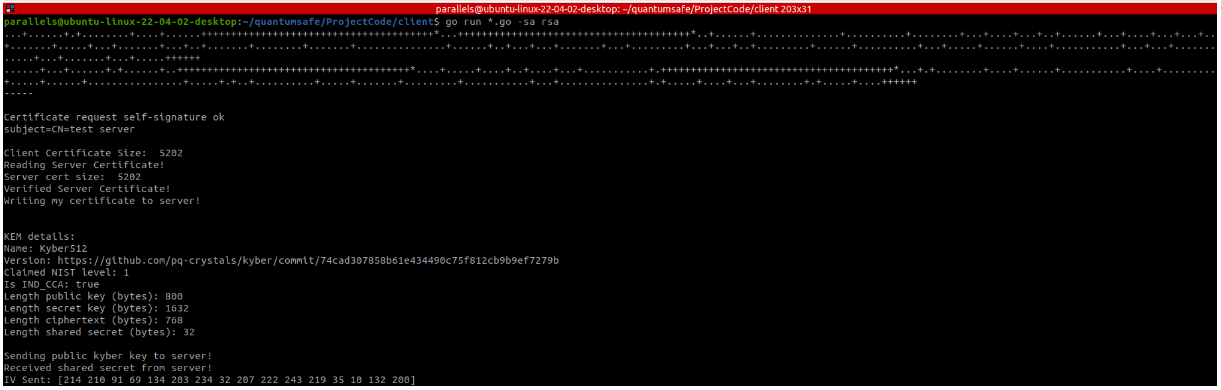
Expected Result: Client and server authenticate, exchange secret, and tunnel is formed

Actual Result:



parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x23

```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go -sa rsa
......+.........+.+..++++++++++++++++++++++++++++++++++++++++*........+.++++++++++++++++++++++++++++++++++++++++*...+.+...........+....+...+...+...+.....+.+...........+...+.......+..++++++
...........++++++++++++++++++++++++++++++++++++++++*.........+.....+.........+...+...........+....+..+...+..+...+.+......+..+.+..+....+...+..+....+.....+.+.....+......+...+.+...+...........+.......+...........
....++++++++++++++++++++++++++++++++++++++++*............++++++
-----

Certificate request self-signature ok
subject=CN=test server

Server Certificate Size:  5202
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
Client Cert Size:  5202
Verified Cert Certificate!

Received client public key!
Sending client shared secret in cipher!
Received IV: [214 210 91 69 134 203 234 32 207 222 243 219 35 10 132 200]
Hello 127.0.0.1:57680
Data:  0 EOF -
127.0.0.1:57680 Closed Connection
writing results to file
```



parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x31

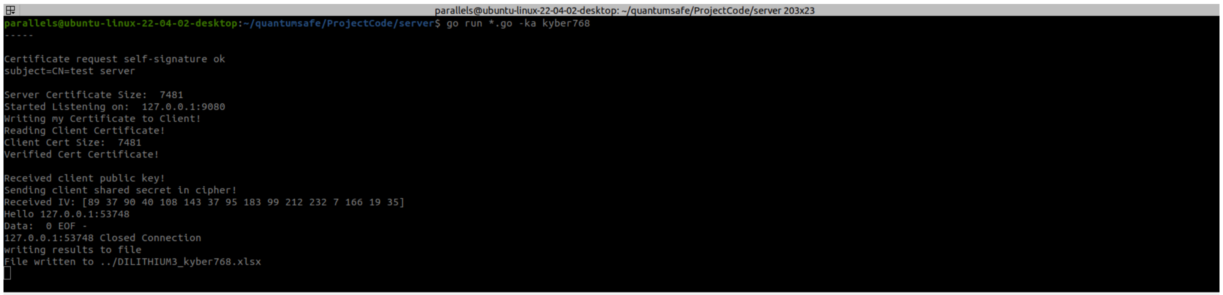
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go -sa rsa
...+......+.+........+....+......++++++++++++++++++++++++++++++++++++++++*...++++++++++++++++++++++++++++++++++++++++*..+......+..............+..........+........+...+....+...+..+.......+...+....+...+...+..
+.......+.....+...+.......+...+..+.......+........+.......+..............+......+..+...+...+.......+...+....+...+.........+.....+..........+...+.....+......+....+...........+...+...+.......
.....+...+.......+...+.....++++++
......+...+......+.+......+..++++++++++++++++++++++++++++++++++++++++*....+.....+....+..+....+...+..........+.++++++++++++++++++++++++++++++++++++++++*...+.+.........+....+.......+...........+....+.........
+.....+......+...............+.....+.+..+..........+.....+.......+.........+...........+...+...............+.+.....+....+...+.........+.+.....+....++++++
-----

Certificate request self-signature ok
subject=CN=test server

Client Certificate Size:  5202
Reading Server Certificate!
Server cert size:  5202
Verified Server Certificate!
Writing my certificate to server!


KEM details:
Name: Kyber512
Version: https://github.com/pq-crystals/kyber/commit/74cad307858b61e434490c75f812cb9b9ef7279b
Claimed NIST level: 1
Is IND_CCA: true
Length public key (bytes): 800
Length secret key (bytes): 1632
Length ciphertext (bytes): 768
Length shared secret (bytes): 32

Sending public kyber key to server!
Received shared secret from server!
IV Sent: [214 210 91 69 134 203 234 32 207 222 243 219 35 10 132 200]
```

Dilithium3 SA, Kyber768 KA, Valid CA

Description: Open client and server with same sa, ka, and ca

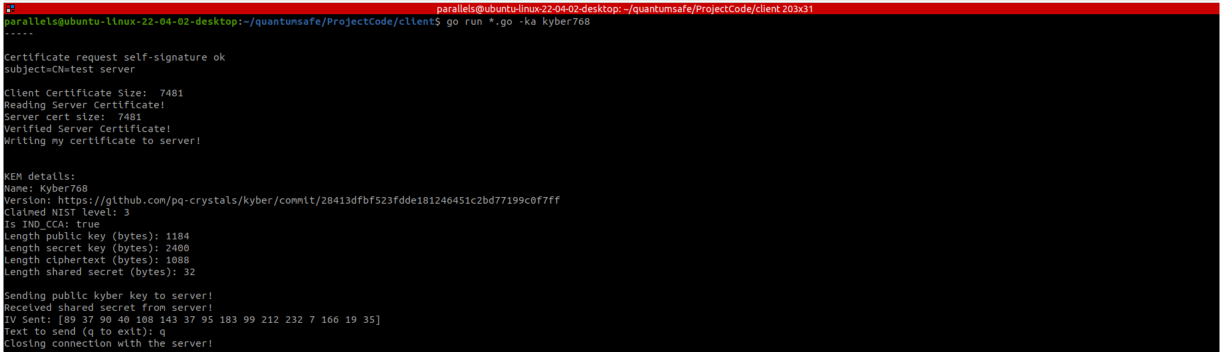
Expected Result: Client and server authenticate, exchange secret, and tunnel is formed

Actual Result:



parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x23

```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go -ka kyber768
-----

Certificate request self-signature ok
subject=CN=test server

Server Certificate Size:  7481
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
Client Cert Size:  7481
Verified Cert Certificate!

Received client public key!
Sending client shared secret in cipher!
Received IV: [89 37 90 40 108 143 37 95 183 99 212 232 7 166 19 35]
Hello 127.0.0.1:53748
Data:  0 EOF -
127.0.0.1:53748 Closed Connection
writing results to file
File written to ../DILITHIUM3_kyber768.xlsx
```



parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x31

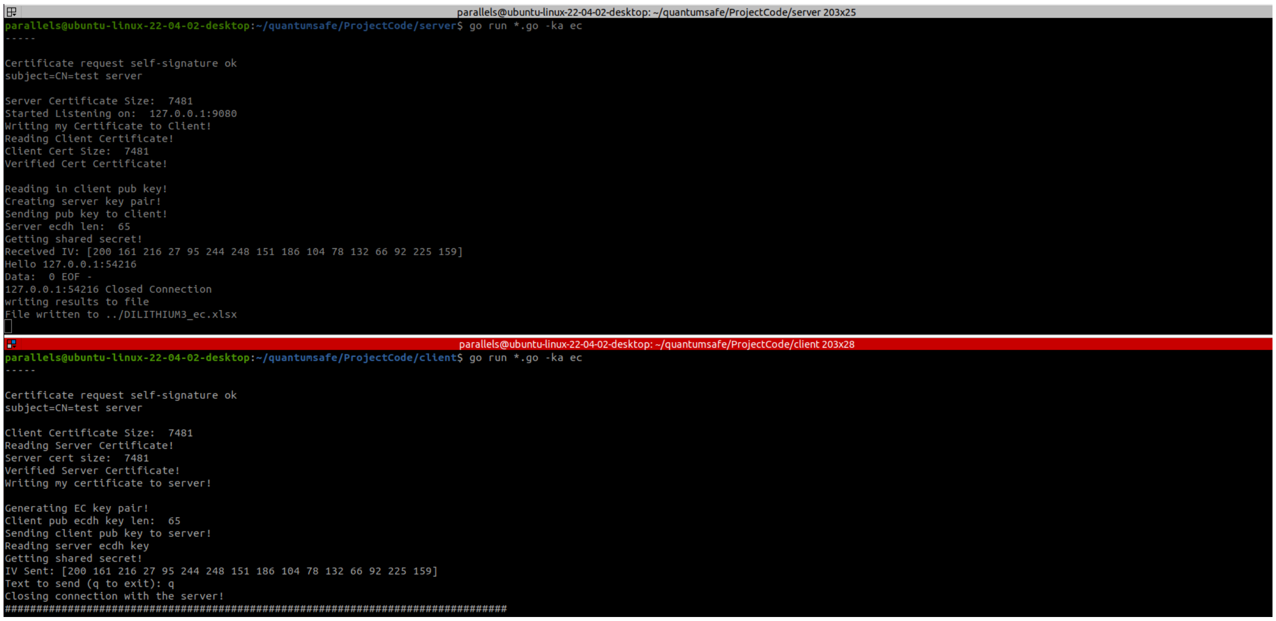
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go -ka kyber768
-----

Certificate request self-signature ok
subject=CN=test server

Client Certificate Size:  7481
Reading Server Certificate!
Server cert size:  7481
Verified Server Certificate!
Writing my certificate to server!


KEM details:
Name: Kyber768
Version: https://github.com/pq-crystals/kyber/commit/28413dfbf523fdde181246451c2bd77199c0f7ff
Claimed NIST level: 3
Is IND_CCA: true
Length public key (bytes): 1184
Length secret key (bytes): 2400
Length ciphertext (bytes): 1088
Length shared secret (bytes): 32

Sending public kyber key to server!
Received shared secret from server!
IV Sent: [89 37 90 40 108 143 37 95 183 99 212 232 7 166 19 35]
Text to send (q to exit): q
Closing connection with the server!
```

Dilithium3 SA, ECDH KA, Valid CA

Description: Open client and server with same sa, ka, and ca

Expected Result: Client and server authenticate, exchange secret, and tunnel is formed

Actual Result:


parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x25

```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go -ka ec
-----

Certificate request self-signature ok
subject=CN=test server

Server Certificate Size:  7481
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
Client Cert Size:  7481
Verified Cert Certificate!

Reading in client pub key!
Creating server key pair!
Sending pub key to client!
Server ecdh len:  65
Getting shared secret!
Received IV: [200 161 216 27 95 244 248 151 186 104 78 132 66 92 225 159]
Hello 127.0.0.1:54216
Data:  0 EOF -
127.0.0.1:54216 Closed Connection
writing results to file
File written to ../DILITHIUM3_ec.xlsx
```

parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x28

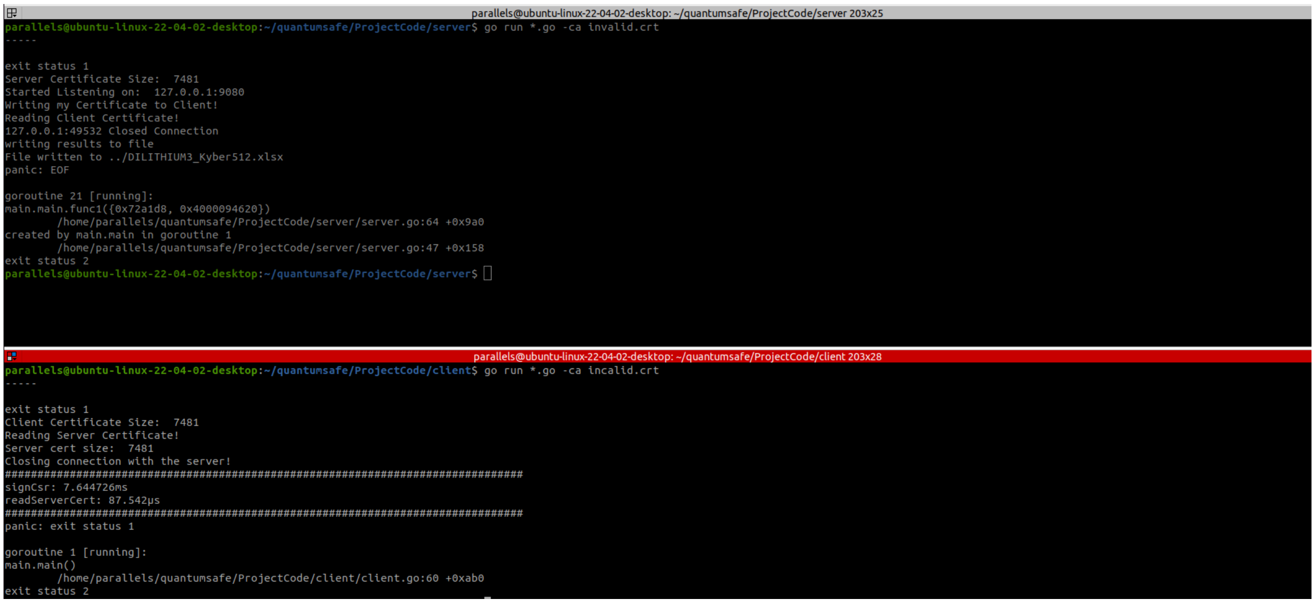
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go -ka ec
-----

Certificate request self-signature ok
subject=CN=test server

Client Certificate Size:  7481
Reading Server Certificate!
Server cert size:  7481
Verified Server Certificate!
Writing my certificate to server!

Generating EC key pair!
Client pub ecdh key len:  65
Sending client pub key to server!
Reading server ecdh key
Getting shared secret!
IV Sent: [200 161 216 27 95 244 248 151 186 104 78 132 66 92 225 159]
Text to send (q to exit): q
Closing connection with the server!
################################################################################
```

Dilithium3 SA, Kyber512 KA, Invalid CA

Description: Open client and server with same sa, ka, and an invalid ca

Expected Result: Client and server can’t authenticate, error thrown, connection closed

Actual Result:


parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x25

```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go -ca invalid.crt
-----

exit status 1
Server Certificate Size:  7481
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
127.0.0.1:49532 Closed Connection
writing results to file
File written to ../DILITHIUM3_Kyber512.xlsx
panic: EOF

goroutine 21 [running]:
main.main.func1({0x72a1d8, 0x4000094620})
        /home/parallels/quantumsafe/ProjectCode/server/server.go:64 +0x9a0
created by main.main in goroutine 1
        /home/parallels/quantumsafe/ProjectCode/server/server.go:47 +0x158
exit status 2
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ 
```

parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x28

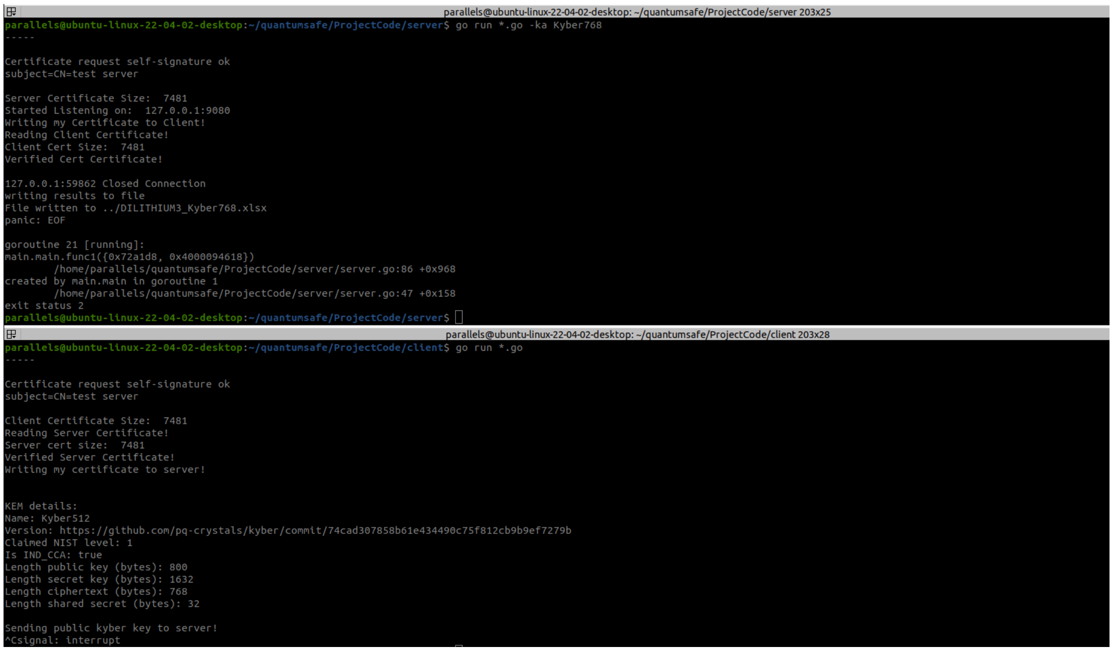
```
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go -ca incalid.crt
-----

exit status 1
Client Certificate Size:  7481
Reading Server Certificate!
Server cert size:  7481
Closing connection with the server!
################################################################################
signCsr: 7.644726ms
readServerCert: 87.542µs
################################################################################
panic: exit status 1

goroutine 1 [running]:
main.main()
        /home/parallels/quantumsafe/ProjectCode/client/client.go:60 +0xab0
exit status 2
```

Dilithium3 SA, Different KA between client and server, Valid CA

Description: Open client and server with same sa, client with Kyber512 ka and server with Kyber768 ka, and ca

Expected Result: Client and server authenticate, error exchanging secret, connection closed

Actual Result:



```
parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x25
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go -ka Kyber768
-----

Certificate request self-signature ok
subject=CN=test server

Server Certificate Size:  7481
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
Client Cert Size:  7481
Verified Cert Certificate!

127.0.0.1:59862 Closed Connection
writing results to file
File written to ../DILITHIUM3_Kyber768.xlsx
panic: EOF

goroutine 21 [running]:
main.main.func1({0x72a1d8, 0x4000094618})
        /home/parallels/quantumsafe/ProjectCode/server/server.go:86 +0x968
created by main.main in goroutine 1
        /home/parallels/quantumsafe/ProjectCode/server/server.go:47 +0x158
exit status 2
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ 
```

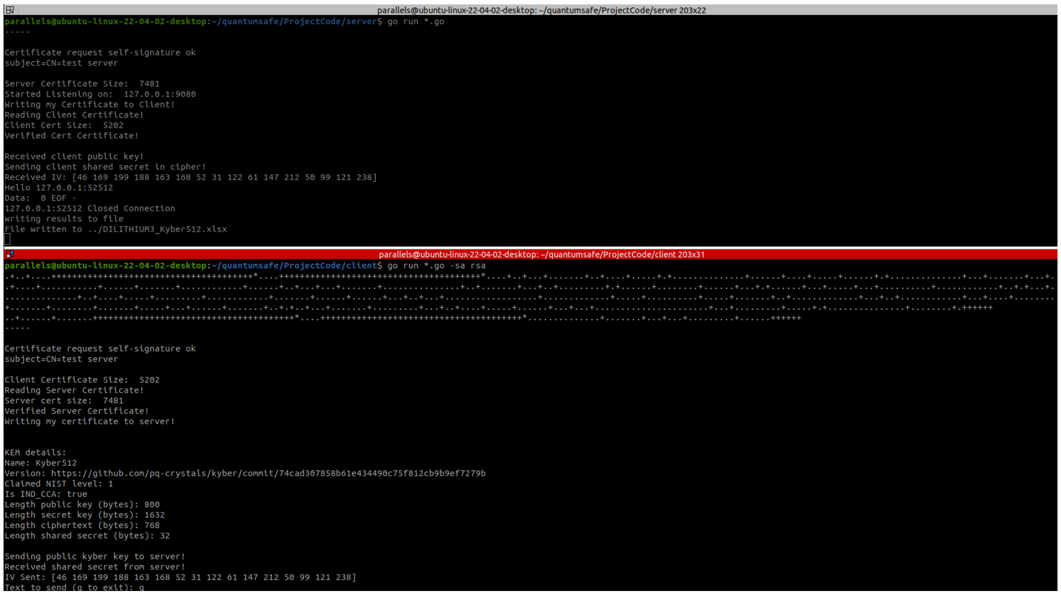
```
parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x28
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go
-----

Certificate request self-signature ok
subject=CN=test server

Client Certificate Size:  7481
Reading Server Certificate!
Server cert size:  7481
Verified Server Certificate!
Writing my certificate to server!


KEM details:
Name: Kyber512
Version: https://github.com/pq-crystals/kyber/commit/74cad307858b61e434490c75f812cb9b9ef7279b
Claimed NIST level: 1
Is IND_CCA: true
Length public key (bytes): 800
Length secret key (bytes): 1632
Length ciphertext (bytes): 768
Length shared secret (bytes): 32

Sending public kyber key to server!
^Csignal: interrupt
```

Different SA between client and server, Kyber512 KA, Valid CA

Description: Open client and server with different sa between server and client (server with dilithium3, client with RSA), same ka, and ca

Expected Result: Client and server authenticate, exchange secret, and tunnel is formed

Actual Result:



```
parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/server 203x22
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/server$ go run *.go
-----

Certificate request self-signature ok
subject=CN=test server

Server Certificate Size:  7481
Started Listening on:  127.0.0.1:9080
Writing my Certificate to Client!
Reading Client Certificate!
Client Cert Size:  5202
Verified Cert Certificate!

Received client public key!
Sending client shared secret in cipher!
Received IV: [46 169 199 188 163 168 52 31 122 61 147 212 50 99 121 238]
Hello 127.0.0.1:52512
Data:  0 EOF -
127.0.0.1:52512 Closed Connection
writing results to file
File written to ../DILITHIUM3_Kyber512.xlsx
```

```
parallels@ubuntu-linux-22-04-02-desktop: ~/quantumsafe/ProjectCode/client 203x31
parallels@ubuntu-linux-22-04-02-desktop:~/quantumsafe/ProjectCode/client$ go run *.go -sa rsa
.+..+....+++++++++++++++++++++++++++++++++++++++*....+++++++++++++++++++++++++++++++++++++++*....+..+...+......+..+...+........+..+....+.....+.+..............+.......+....+.....+......+.+..............+...+........+...+.
.+....+..........+......+.......+...........+......+..+...+...+.......+..............+..+.......+...+..+.....+...+..+.........+.+.......+.........+......+...+.+......+...+......+...+...........+............+..+.+...+.
.............+..+....+.....+.........+............+.......+......+......+...+..+...+.................+.............+............+.....+.........+..+...............+...+..+.............+...+....+........
+.......+.......+.......+.....+...+......+.......+..+.+..+...+.........+.........+...+..+....+.....+......+.....+...+...+.........................+...+.........+.....+.+...............+........+.++++++
..+......+.......+++++++++++++++++++++++++++++++++++++++*....+++++++++++++++++++++++++++++++++++++++*..............+.......+...+...+.........+......++++++
-----

Certificate request self-signature ok
subject=CN=test server

Client Certificate Size:  5202
Reading Server Certificate!
Server cert size:  7481
Verified Server Certificate!
Writing my certificate to server!


KEM details:
Name: Kyber512
Version: https://github.com/pq-crystals/kyber/commit/74cad307858b61e434490c75f812cb9b9ef7279b
Claimed NIST level: 1
Is IND_CCA: true
Length public key (bytes): 800
Length secret key (bytes): 1632
Length ciphertext (bytes): 768
Length shared secret (bytes): 32

Sending public kyber key to server!
Received shared secret from server!
IV Sent: [46 169 199 188 163 168 52 31 122 61 147 212 50 99 121 238]
Text to send (q to exit): q
```