**W3 TNE30024 Lab Notaions**

**Create Intermediate and Final Certificates**

**Step 1: Create the First Intermediate Certificate**

**# Generate a private key for the first intermediate certificate**

openssl genpkey -algorithm RSA -out keys/IntCertA1.key

**# Generate a certificate signing request (CSR) for the intermediate certificate**

openssl req -new -key keys/IntCertA1.key -out req1.csr -subj "/CN=IntCertA1"

**# Sign the CSR to generate the intermediate certificate IntCertA1.crt**

openssl x509 -req -in req1.csr -set\_serial 10 -CA certs/root.crt -CAkey keys/root.key -passin pass:challenge -days 365 -extfile v3.ext -out certs/IntCertA1.crt

**# Verify the intermediate certificate**

openssl verify -verbose -CAfile certs/root.crt certs/IntCertA1.crt

**Step 2: Create the Second Intermediate Certificate**

**# Generate a private key for the second intermediate certificate**

openssl genpkey -algorithm RSA -out keys/IntCertA2.key

**# Generate a CSR for the second intermediate certificate**

openssl req -new -key keys/IntCertA2.key -out req2.csr -subj "/CN=IntCertA2"

**# Sign the CSR to generate the intermediate certificate IntCertA2.crt**

openssl x509 -req -in req2.csr -set\_serial 20 -CA certs/IntCertA1.crt -CAkey keys/IntCertA1.key -passin pass:challenge -days 365 -extfile v3.ext -out certs/IntCertA2.crt

**# Concatenate IntCertA1.crt and IntCertA2.crt into a single file**

cat certs/IntCertA1.crt certs/IntCertA2.crt > certs/combinedCerts.crt

Notation for Oral test: CONCATENATE root.crt at this stage alongside IntCertA1 and IntCertA2 can ensure all of the 3 certs IntCertA1, IntCertA2 and FinalCertA being verified at the end from Step 3.

**# Verify the concatenated certificates**

openssl verify -verbose -CAfile certs/root.crt certs/combinedCerts.crt

**Step 3: Create the Final Certificate**

**# Generate a private key for the final certificate**

openssl genpkey -algorithm RSA -out keys/FinalCertA.key

**# Generate a CSR for the final certificate**

openssl req -new -key keys/FinalCertA.key -out req3.csr -subj "/CN=FinalCertA"

**# Sign the CSR to generate the final certificate FinalCertA.crt**

openssl x509 -req -in req3.csr -set\_serial 20 -CA certs/IntCertA2.crt -CAkey keys/IntCertA2.key -passin pass:challenge -days 365 -out certs/FinalCertA.crt

**# Add FinalCertA.crt to the combined certificates file**

cat certs/FinalCertA.crt >> certs/combinedCerts.crt

**# Move into the certs directory and verify all certificates**

cd certs

openssl verify -verbose -CAfile root.crt IntCertA1.crt IntCertA2.crt FinalCertA.crt

**Step 4: Create an Additional Final Certificate (FinalCertB)**

**# Generate a private key for another final certificate**

openssl genpkey -algorithm RSA -out keys/FinalCertB.key

**# Generate a CSR for the new final certificate**

openssl req -new -key keys/FinalCertB.key -out req4.csr -subj "/CN=FinalCertB"

**# Sign the CSR using IntCertA1 to generate the final certificate FinalCertB.crt**

openssl x509 -req -in req4.csr -set\_serial 21 -CA certs/IntCertA1.crt -CAkey keys/IntCertA1.key -passin pass:challenge -days 365 -extfile v3.ext -out certs/FinalCertB.crt

**# Add FinalCertB.crt to the combined certificates file**

cat certs/FinalCertB.crt >> combinedCerts.crt

**# Verify the new certificate chain**

openssl verify -verbose -CAfile root.crt combinedCerts.crt

**REFLECTIONS:**

Output shows such as:  
student@rule103:~/lab3 % openssl verify -verbose -CAfile certs/root.crt certs/IntCertA1.crt

certs/IntCertA2.crt certs/FinalCertA.crt

certs/IntCertA1.crt: OK

CN = IntCertA2

error 20 at 0 depth lookup: unable to get local issuer certificate error

certs/IntCertA2.crt: verification failed

CN = FinalCertA error 20 at 0 depth lookup: unable to get local issuer certificate error

certs/FinalCertA.crt: verification failed

**Understanding the Issue**

1. **Chain of Trust**: For a certificate chain to be verified:
   * Each certificate must be directly signed by the next authority in the chain.
   * OpenSSL must be able to trace this chain from the certificate being verified up to a trusted root entity (in this case, root.crt).
2. **Error 20 - Unable to Get Local Issuer Certificate**:
   * This error occurs when the verifying authority’s certificate is not found or not specified correctly during the verification. For IntCertA2.crt and FinalCertA.crt, OpenSSL couldn't verify them directly against root.crt because they are not signed directly by the root CA but by intermediate CAs.

**Solutions:  
Step 1: Concatenate the Intermediate Certificates for Verification**

cat certs/IntCertA1.crt certs/IntCertA2.crt > certs/intermediate\_chain.crt

**Step 2: Verify Using Intermediate Certificates as Chain**

openssl verify -verbose -CAfile certs/root.crt -untrusted certs/intermediate\_chain.crt certs/IntCertA2.crt

openssl verify -verbose -CAfile certs/root.crt -untrusted certs/intermediate\_chain.crt certs/FinalCertA.crt

**Explanation:**

-CAfile: Specifies the root certificate. This is the top of the trust chain and must be trusted absolutely.

-untrusted: Specifies intermediate certificates that help form the chain of trust from the root to the target certificate but are not themselves trusted like the root.

**Step 3: Update Your Verification Process for All Certificates**

cat certs/IntCertA1.crt certs/IntCertA2.crt > certs/full\_chain.crt

openssl verify -verbose -CAfile certs/root.crt -untrusted certs/full\_chain.crt certs/IntCertA1.crt certs/IntCertA2.crt certs/FinalCertA.crt

**ORAL PREPERATIONS:**

openssl genpkey -algorithm RSA -out private\_key.pem -pkeyopt rsa\_keygen\_bits:2048

Alternative command to generate private key:  
openssl genrsa -out private\_key.pem 2048