4. [20 marks] TLS Key Kraziness

As we've talked about in class a single TLS connection involves *a lot* of keys. In this question you will enumerate *all* of the keys used in a single TLS connection to *google.ca*. This includes all public, private, and secret keys exchanged or generated during a TLS handshake.

Deliverables. Place answers in a directory Q4. First list the ciphersuite your browser connected under. The file then should have 3 columns: a description of the key (*e.g.,* Google's public ECDHE key), the name of the entities who know the key (*e.g.,* everyone), and a sentence or two describing the purpose of the key (*e.g.,* used for public key agreement)

1. GeoTrust Global CA signing key, used to self-sign GeoTrust's certificate, and to sign Google Internet Authority G2's certificate. Only Geo trust knows it.

2. GeoTrust Global CA verification key, used to verify the signatures on GeoTrust's certificate, and Google Internet Authority G2's certificate. Everyone knows it.

3. Google Internet Authority G2's signing key, used to sign Google.ca's certificate. Only Google Internet Authority G2 knows it.

4. Google Internet Authority G2's verification key, used to verify signature on google.ca's certificate. Everyone knows it.

5. Google.ca's signing key, used to sign Google.ca's ephemeral elliptic-curve Diffie-Hellman (ECDHE) public key. Only google.ca knows it.

6. Google.ca's verification key, used to verify the signature on google.ca's ECDHE public key. Everyone knows it.

7. Google.ca's ECDHE public key, sent to client for key agreement. Everyone knows it.

8. Google.ca's ECDHE private key, used to generate the corresponding public key, and to apply to client's public key to form shared secret. Only Google.ca knows it.

9. Client's ECDHE public key, sent to google.ca for key agreement. Everyone knows it.

10. Client's ECDHE private key, used to generate the corresponding public key, and to apply to google.ca's public key to form shared secret. Only client knows it.

11. Diffie-Hellman shared secret. It is the result of ECDHE key exchange, also called the pre-master secret. Only client and server know it.

12. Master secret. Derived from pre-master secret and client/server random values, and used to derive the following symmetric key. Only client and server know it.

13. Client-write symmetric encryption key. Used by the client to encrypt messages. Used by the server to decrypt messages. Only client and server know it.

14. Server-write symmetric encryption key. Used by the server to encrypt messages. Used by the client to decrypt messages. Only client and server know it.

15. Client-write MAC key. Used by client to generate a MAC tag. Used by server to verify MAC tag. Only client and server know it.

16. Server-write MAC key. Used by server to generate a MAC tag. Used by client to verify a MAC tag. Only client and server know it.