This sequence diagram describes the SSL processing and the basic cryptography concepts need to understand SSL operation.

SSL is a sophistication encryption scheme that does not require the client and the server to arrange for a secret key to be exchanged between the client and server BEFORE the transaction is started. SSL uses public/private keys to provide a flexible encryption scheme that can be setup at the time of the secure transaction.

In typical encryption schemes the client and server would be required to use a secret key that has been preconfigured in the client and the server machines. In such a scheme, the client would use the secret key to encrypt the data. The server would use the same secret key to decrypt the data. Same logic applies in the server to client direction. This type of preconfigured secret keys are not suitable for Web based secure services that involve millions of users who have no prior secret key arrangement with the secure server.

SSL solves this problem by using asymmetric keys. These keys are defined in pairs of public and private keys. As the name suggests the public key is freely available to anybody. The private key is known only to the server. The keys have two important properties:

(1) Data encrypted by the client using the pubic key can be decrypted only by the server's private key. Due to this property of the keys, the client is able to send secure data that can be understood only by the server.

(2) Data encrypted to by the server's private key can only be decrypted using the public key. This property is useful in a client level authentication of the server. If the server sends a known message (say the name of the server), the client can be sure that it is talking to the authentic server and not an imposter if it is successfully able to decrypt the message using the public key.

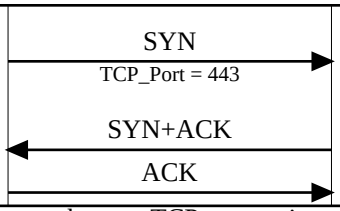
Note that property (1) allows us to use conventional secret keys. A secret key can be sent by the client as data that has been encrypted using the public key. This secret key can be decrypted only by the server. Once the server gets the key, the client and the server are able to communicate using this secret key.

The public/private key based encryption is used only for handshaking and secret key exchange. Once the keys have been exchanged the symmetric secret keys are used. This is done for two reasons:

(1) Public/private key based encryption techniques are computationally very expensive thus their use should be minimized.

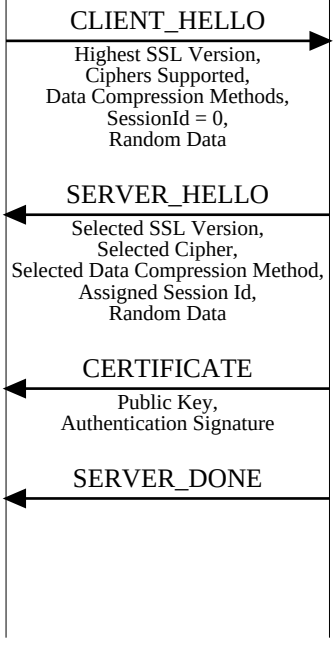
(2) The secret key mechanism is needed for server to client communication.

User clicks on a URL starting with https://www.mybank.com



A secure connection is needed for this session. The browser establishes a TCP connection on the HTTPS TCP Port 443

SSL Handshake on the new TCP connection

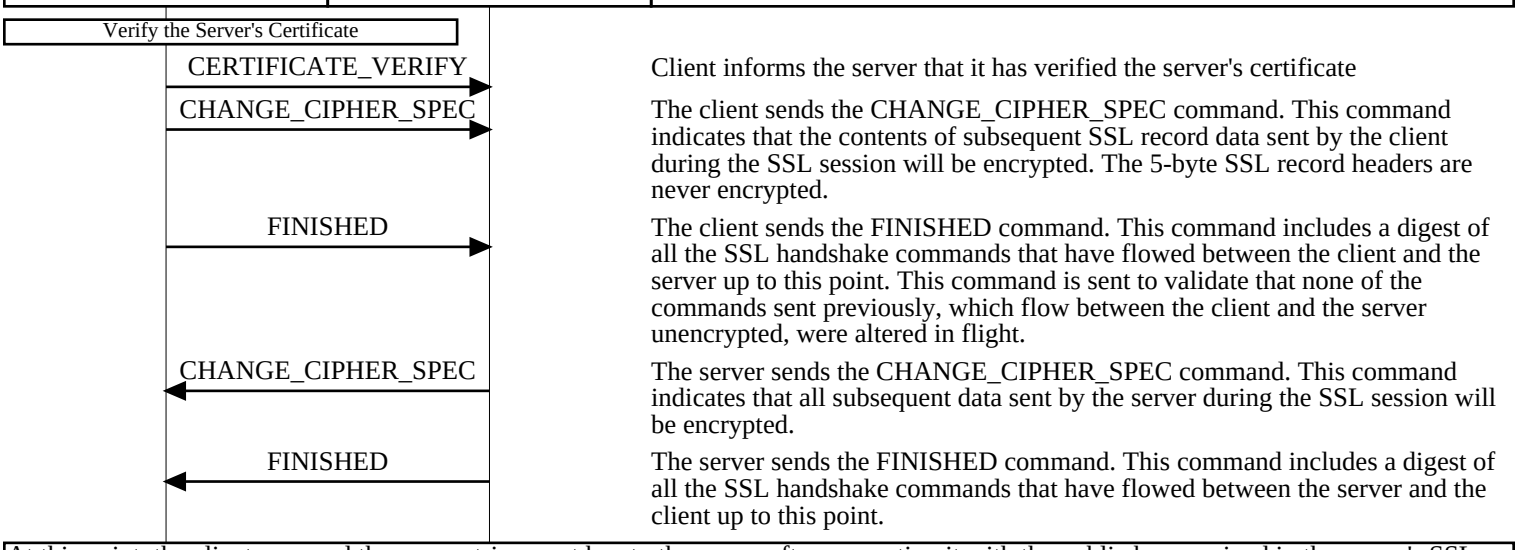


The client sends a CLIENT_HELLO command to the server, which includes: The highest SSL and TLS version supported by the client. Ciphers supported by the client. The ciphers are listed in order of preference. Data compression methods that are supported by the client. The session ID. If the client is starting a new SSL session, the session ID is 0. Random data that is generated by the client for use in the key generation process.

The server sends a SERVER_HELLO command to the client, which includes: The SSL or TLS version that will be used for the SSL session. The cipher that will be used for the SSL session. Data compression method that will be used for the SSL session. SSL Version 3 and TLS have no compression algorithms defined at this time. The session ID for the SSL session. Random data that is generated by the server for use in the key generation process.

The server sends the CERTIFICATE command. This command includes the server's certificate and, optionally, a chain of certificates beginning with the certificate of the certificate authority (CA) that assigned the server's certificate.

The server sends the SERVER_DONE command. This command indicates that the server has completed this phase of the SSL handshake. If client authentication will be performed, additional commands flow between the client and the server following the SERVER_DONE command.



At this point, the client can send the symmetric secret key to the server after encrypting it with the public key received in the server's SSL certificate. This encrypted secret key can only be decrypted using the private key. Thus only the server is able to decrypt the message