SSL (1995-99) / TLS (1999-)

SSL was invented by Netscape in the 1990s to enable secure web transactions. It has two components to it:

- Handshake Protocol: to establish authenticity and a shared master secret.
- Record protocol: secure communication given a shared master secret.

Right now we are only looking at authentication protocols, so we look at the handshake only.
Handshake (key agreement)

Client to Server: ClientHello

highest version supported
client random nonce
session_id (0 if new, else resume old)
crypto_suite (list of enc/mac algorithms)
compression_method (0 if none)
Handshake (key agreement)

Server to Client: ServerHello

chosen version
server random nonce
session_id
chosen crypto_suite
chosen compression_method
Options in ServerHello

ServerCertificate (technically an option, but normally done)
ServerKeyExchange (if DH-based crypto_suite chosen)
CertificateRequest (If server wants to authenticate client)
Handshake (key agreement)

Client to Server: ClientFinished

ClientCertificate (if requested)
ClientKeyExchange (DH, or shared secret encrypted using server cert)
CertificateVerify (signed data, if server has client certificate)
ClientFinished (MAC of everything sent during handshake)
Handshake (key agreement)

Server to Client: ServerFinished

ServerFinished (MAC of everything sent during handshake)
Handshake (key agreement)

Most TLS/HTTPS connections:

— Server sends certificate
— Client does not send certificate
— Client authenticates self via password or credit card

Client confidence comes from ClientKeyExchange and/or ServerKeyExchange which use server's certificate for authentication.
OpenSSL

BIO abstraction allows reading and writing data of various types using a single abstraction.

```c
BIO *bio = BIO_new_connect("hostname:port");
BIO_do_connect(bio);
BIO_read(bio, buf, len);
BIO_write(bio, buf, len);
```
OpenSSL

OpenSSL tries to make TLS just as easy.

```c
SSL_CTX  ctx = SSL_CTX_new(TLS_client_method());
SSL_CTX_load_verify_locations(ctx, "/path/to/TrustStore.pem", NULL));
BIO *bio = BIO_new_ssl_connect(ctx);
BIO_set_conn_hostname(bio, "hostname:port");
BIO_do_connect(bio);
BIO_read(bio, buf, len);
BIO_write(bio, buf, len);
```