UNC CHARLOTTE
College of Computing and Informatics

TLS and Certificate Authorities

IT Infrastructure and Security

Transport Layer Security

- Security Protocol
  - Formerly provided by Secure Socket Layer
    - SSL
- Provides:
  - Authentication
  - Confidentiality
  - Integrity
- Widely used on the Internet

TLS history

- SSL originally developed at Netscape
- SSL Version 2.0 was first public release (1995)
- SSL Version 3.0 soon followed (1996)
  - Corrected various security flaws of 2.0
- TLS first defined in 1999
  - Not backwards compatible with SSL

Modes of Operation

- TLS has two modes of operation
  - Implicit — aka “by protocol”
  - Explicit — aka “by port”

Implicit Mode

- Runs on a separate port from non-encrypted traffic
  - Deprecated from many protocols
- e.g. HTTP (80/tcp) vs. HTTPS (443/tcp)

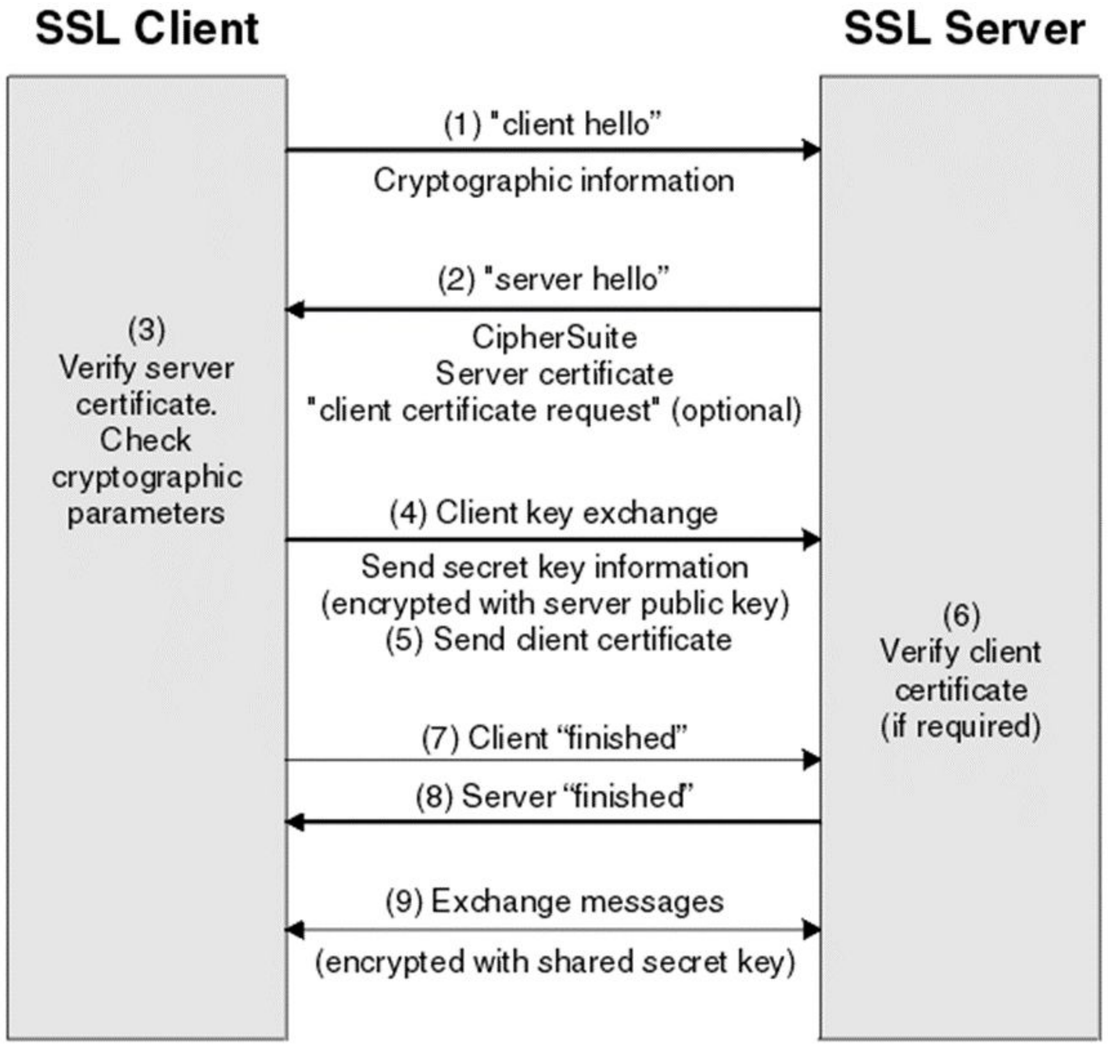
Explicit Mode

- Requires application be TLS aware
- One port to rule them all
- Communications start unencrypted
  - Client sends a ‘STARTTLS’ to initiate encrypted session
- e.g. IMAP, LDAP, POP3, SMTP

TLS Handshake

- Client opens connection to server
- Client and server agree on protocol version
- Negotiate cryptographic algorithms to use
- Client authenticates server’s digital certificate
  - Server can optionally authenticate a client’s certificate
- Asymmetric encryption used to share session key
  - Session key is symmetric
  - Symmetric encryption is faster than asymmetric

TLS Handshake



TLS: Trust

- Trust is handled by Certificate Authorities (CA)
- CAs act as a trusted third party
- Verify your identity and issue a signed certificate
- SSL clients are usually pre-loaded with trusted CAs
  - e.g. Verisign
- Certificates are verified by walking the certificate chain to a trusted certificate authority
- Root certificates for the CAs are just self-signed certificates that are marked as trusted
  - Usually by browsers and OS developers

TLS: Implementations

- OpenSSL is de facto standard on Linux
  - Has indispensable command line utility
  - Note the Heartbleed vulnerability
    - Heartbleed in itself is not “dangerous”
    - The danger is in other programs that are not securely written
      - E.g. those that do not clear memory of sensitive information after it is not needed anymore.\
- Supports connecting to any TLS Socket
  - STARTTLS support for FTP, IMAP, POP3, SMTP
- GnuTLS is an alternative