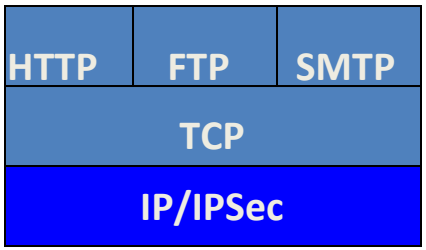
WEB Security: Secure Socket Layer

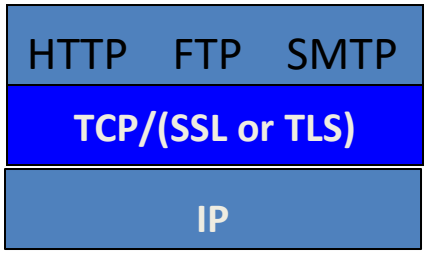
Outline of this Lecture

- Brief Information on SSL and TLS
- Secure Socket Layer (SSL)
- Transport Layer Security (TLS)
- Recommended Reading Materials

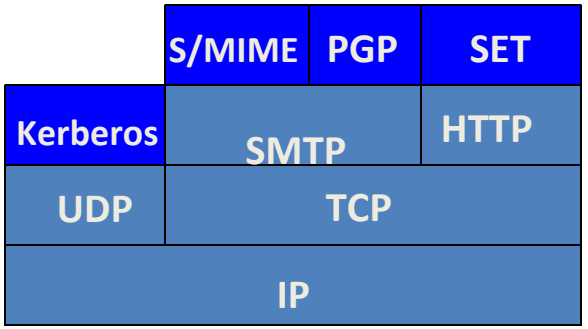
Security Facilities in the TCP/IP Protocol Stack



(a) Network level



(b) Transport level



(c) Application level

SSL and TLS: Information

- SSL was originated by Netscape, Version 1.0, 2.0, 3.0, 3.1
- TLS is an IETF protocol.
- TLS 1.0 (SSL 3.1), TLS 1.1 (SSL 3.2), TLS 1.2 (SSL 3.3), TLS 1.3 released 2018
- They are the most popular transport layer security protocols
- https: http over SSL or TLS (Web secu.)

SSL: Brief Introduction

- Based on connection-oriented and reliable service (e.g., TCP)
- Able to provide security services for any TCP-based application protocol, e.g., HTTP, FTP, TELNET, etc.
  - Application independent

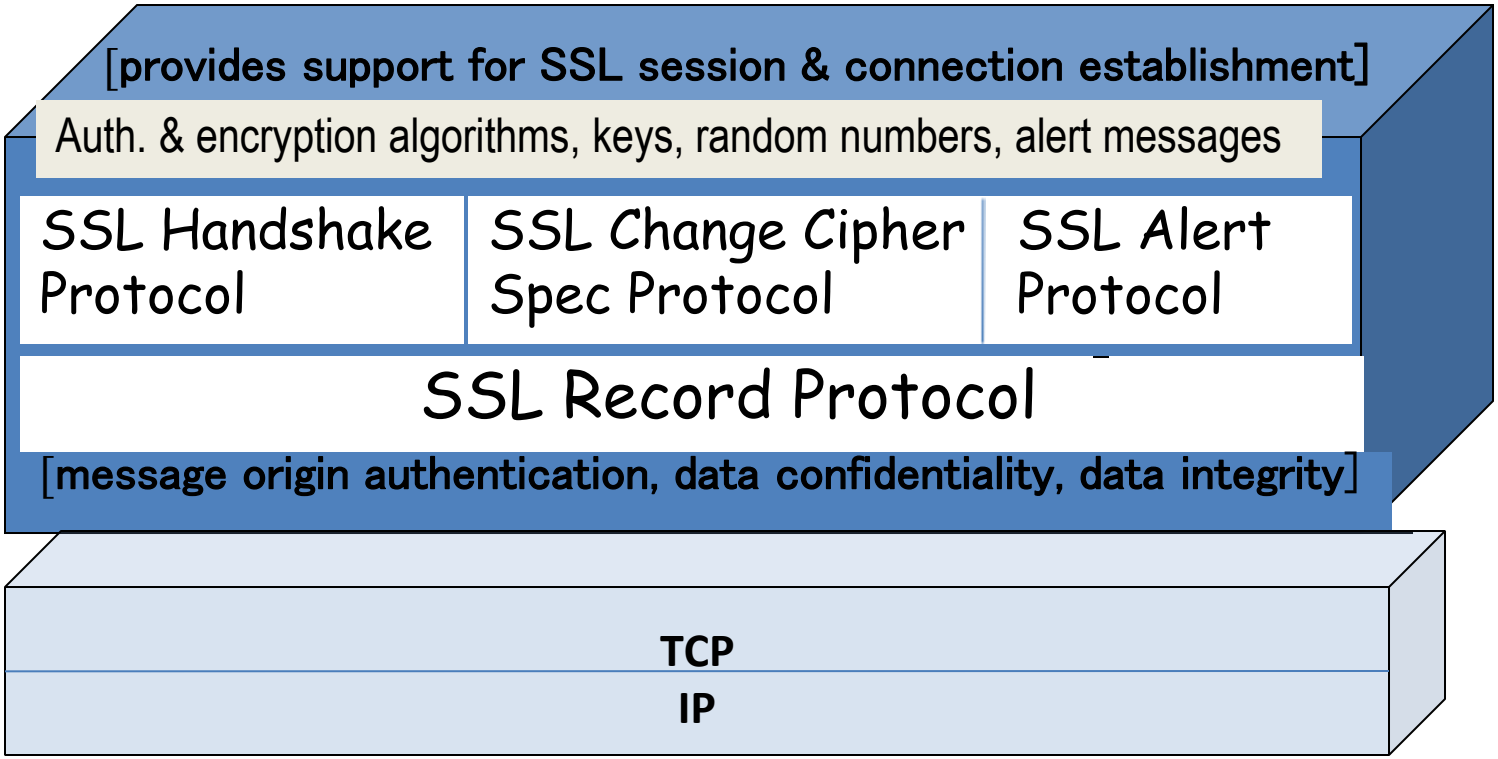
SSL Services

- Client- server authentication
- Data confidentiality
- Data origin authentication
- Data integrity

SSL Architecture

SSL Protocol Structure

It makes use of TCP to provide reliable end-to-end secure service.



SSL Protocol

Components:

- SSL Record Protocol
  - Layered on top of the connection-oriented and reliable transport layer service provided by TCP
  - Provides message origin authentication, data confidentiality, and data integrity
- SSL sub-protocols
  - Layered on top of the SSL Record Protocol
  - Provides support for SSL session and connection establishment

SSL Connection and Session

- **Connection:**
  - a transport (in the OSI layering model definition) that provides a suitable service.
  - For SSL, such connections are peer-to-peer relationships.
  - Every connection is associated with one "session".
- **Session:**
  - an association between a client and a server.
  - Defines a set of cryptographic parameters, which can be shared among multiple connections.
  - It is used to avoid the expensive negotiation of new security parameters for each connection.

SSL Session State: Its Elements

- **Session ID**: An arbitrary byte sequence chosen by the server to identify an active or resumable session state.
- **Peer certificate**: X.509v3 certificate of the peer
- **Compression method**: algorithm to compress data before encryption
- **Cipher spec**: specification of data encryption and Message Authentication Code (MAC) algorithms
- **Master key**: 48-byte secret key shared between client and server
- **Is resumable**: flag that indicates whether the session can be used to initiate new connections

SSL Sessions

- SSL session is stateful
  - SSL session state information is used by both sides
  - SSL protocol must initialize and maintain session state information on either side of the session
- SSL session is used for a number of connections (i.e., it has a lifetime)

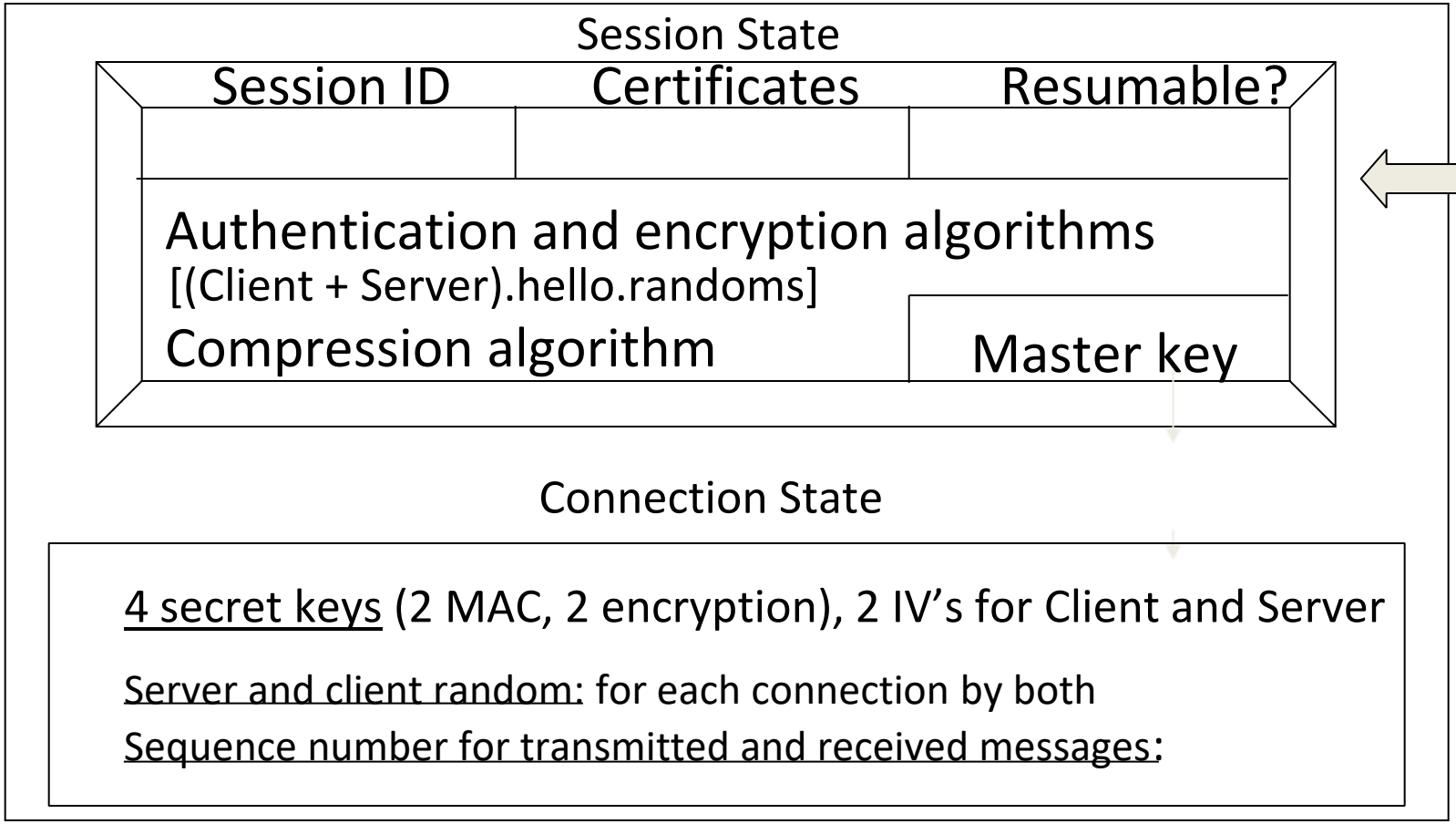
More on SSL Sessions

- A previous session may be resumed (use Session ID and its session cache)
- A new session may be negotiated (use the SSL Handshake Protocol)

Elements of SSL Connection State

- Server and client random: byte sequences that are chosen by the server and client for each connection.
- Server write MAC secret: secret used for MAC on data written by the server
- Client write MAC secret: secret used for MAC on data written by the client [**different from server write MAC key**]
- Server write key: key used for data encryption by server and decryption by the client
- Client write key: key used for encryption by client and decryption by the server [**different from server write key**]
- Initialization vectors: for CBC mode (**two are different**!)
- Sequence number: for both transmitted and received messages, maintained by each party.

Session & Connection State: Pictorial Description



Current and Pending Session State

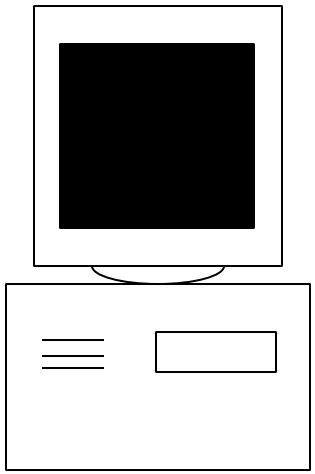
- **Current state**: At each side there is a current operating session state.
  - The current operating session state is void if it is the first time the client and the server use SSL.
- **Pending state**: After a successful Handshaking, the pending session state is created.
- **Updating**: After invoking the Change Cipher Spec Protocol by each side, the pending session state at the side becomes the current state.

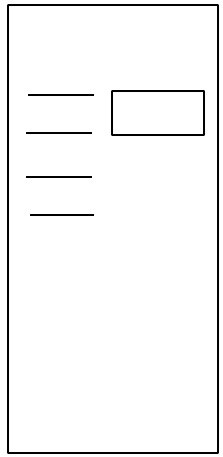
Connection and Session

Establishing a session by the Handshake protocol

Master key, hash algor.
Encryption algorithm,
Session ID, etc.

Change cipher
Spec protocol





Copying pending state into current state

Now ready for connections in this session

Connection 1

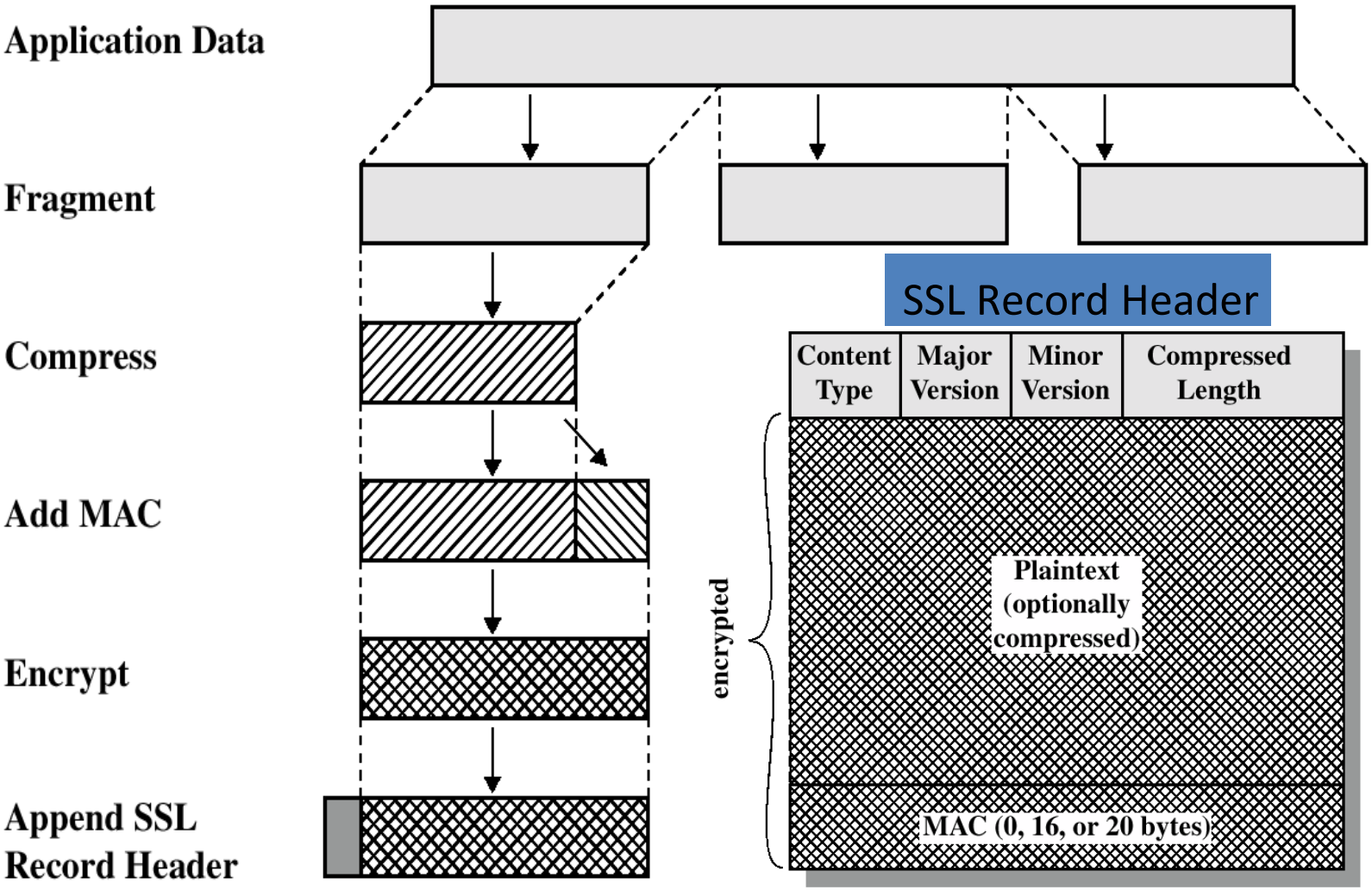
Connection 2

Client

Server

SSL Record Protocol

SSL Record Protocol Operation



SSL Record

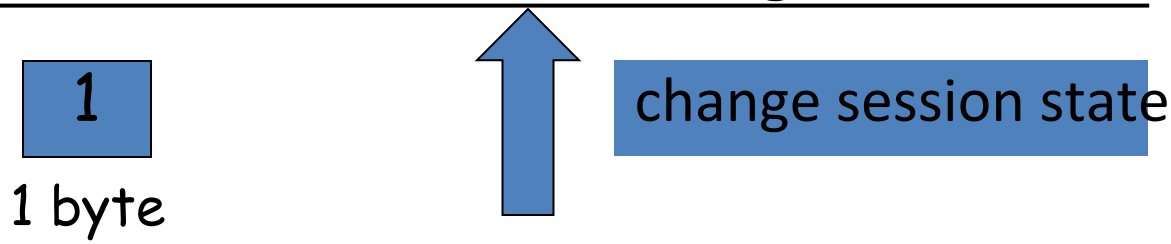
SSL Record Content

- **Content type (8 bits)**
  - Defines higher layer protocol that must be used to process the payload data (which may be handshake, alert, or change_cipher_spec messages).
- **Protocol version number (major & Minor) (8 bits)**
  - Defines SSL version in use. (3, 0) for SSLv3
- **Length (16 bits):** length in bytes of (compressed) plaint.
- **Data payload**
  - Optionally compressed and encrypted
  - Encryption and compression requirements are defined during SSL handshake
- **MAC (0, 16, or 20 bytes)**
  - Appended for each record for message origin authentication and data integrity verification

Change Cipher Spec Protocol

Change Cipher Spec Protocol

- It is one of the three SSL-specific protocols that use the SSL Record Protocol.
- It consists of a single message, which consists of a single byte with value 1.
- The sole purpose of this message is to cause the pending session state to be copied into the current session state, which updates the cipher suite to be used in the coming connection.



Alert Protocol

Alert Protocol

- Used to transmit alerts to peer entity via the SSL Record Protocol.
  - Alert message: (alert level, alert description)
  - Alert messages are *compressed, authenticated and encrypted*, by the SSL Record Protocol.
  - Format of the alert message in this protocol:

Level Alert

1 byte	1 byte

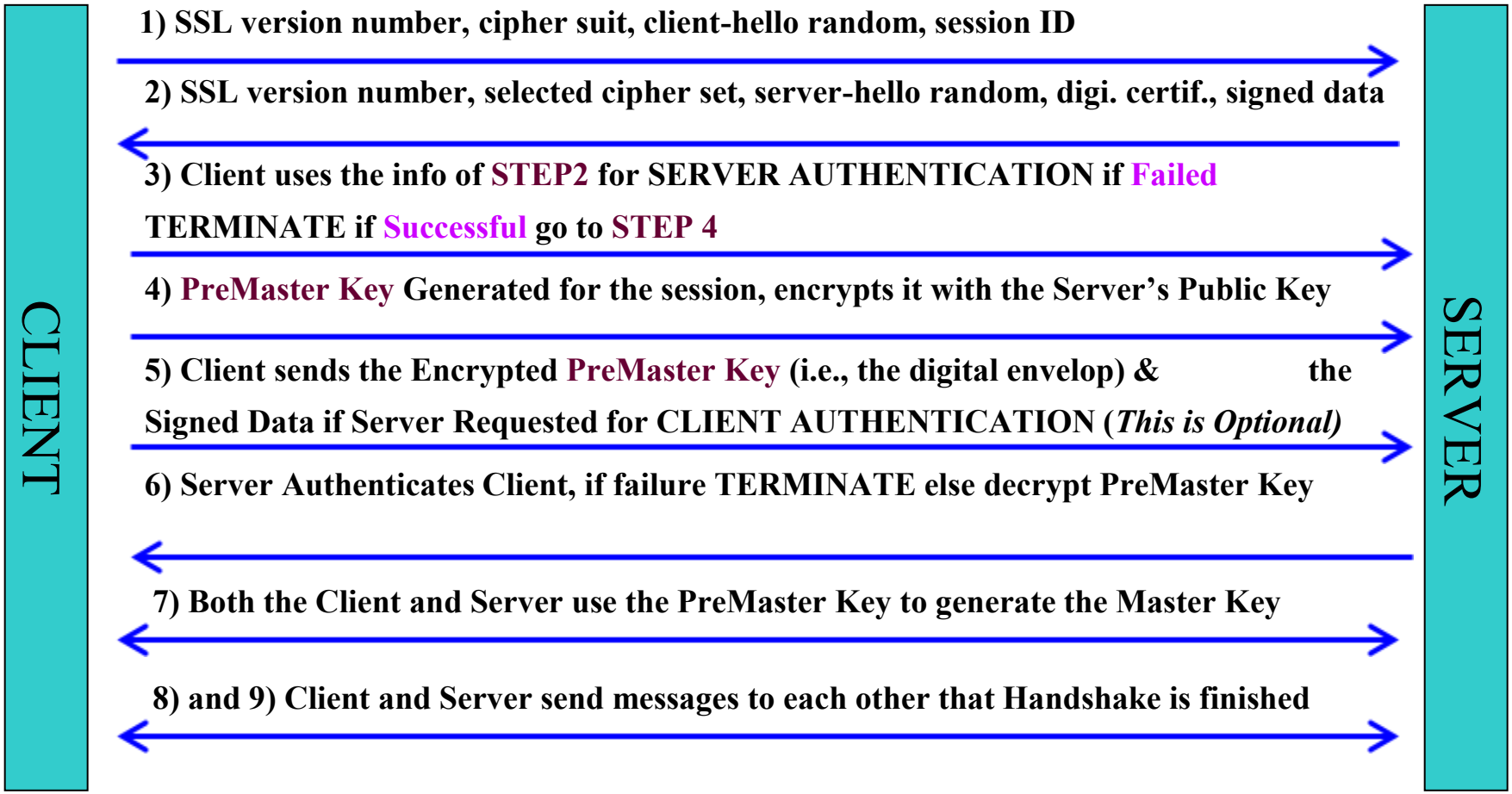
<==> errors occurred during handshaking
<=== errors occurred during processing at the sever

Handshake Protocol

Handshake Protocol

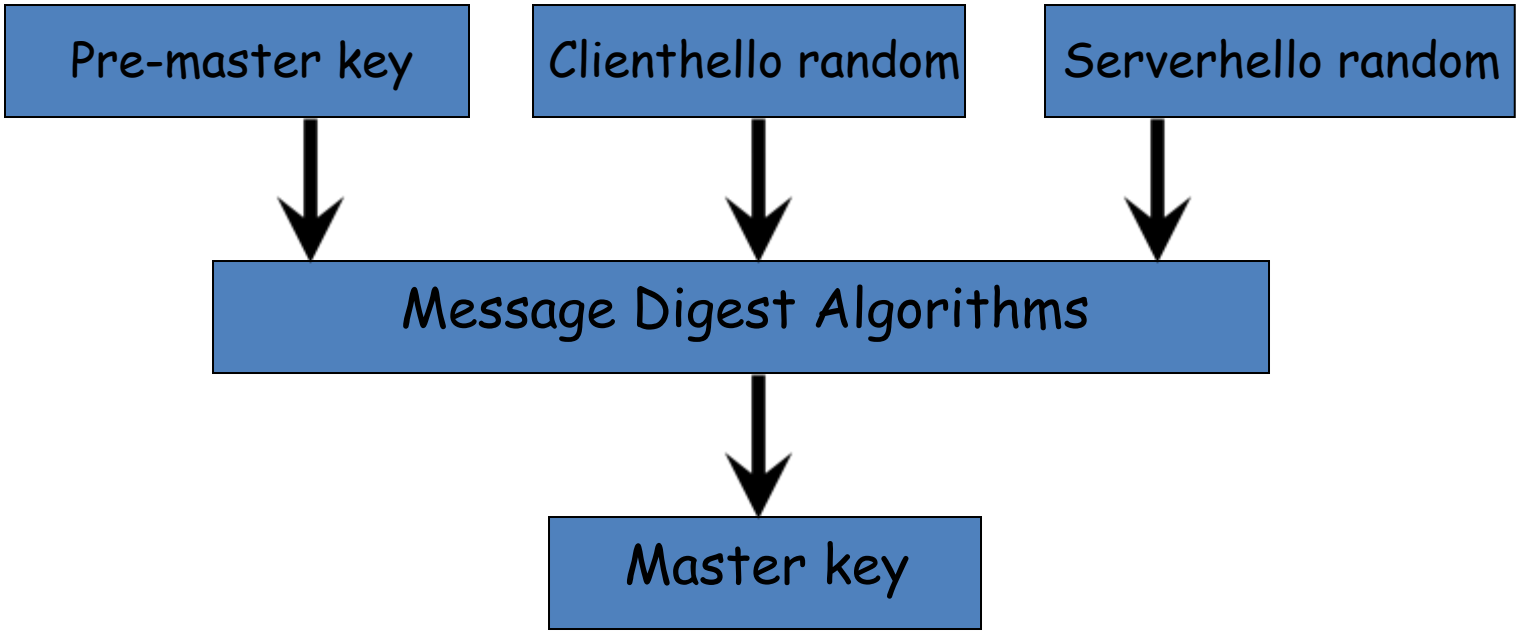
- The most complex part of SSL.
- Allows the server and client to authenticate each other.
- Negotiate encryption and MAC algorithm and a master key.
- Used before any application data is transmitted.

SSL Handshake



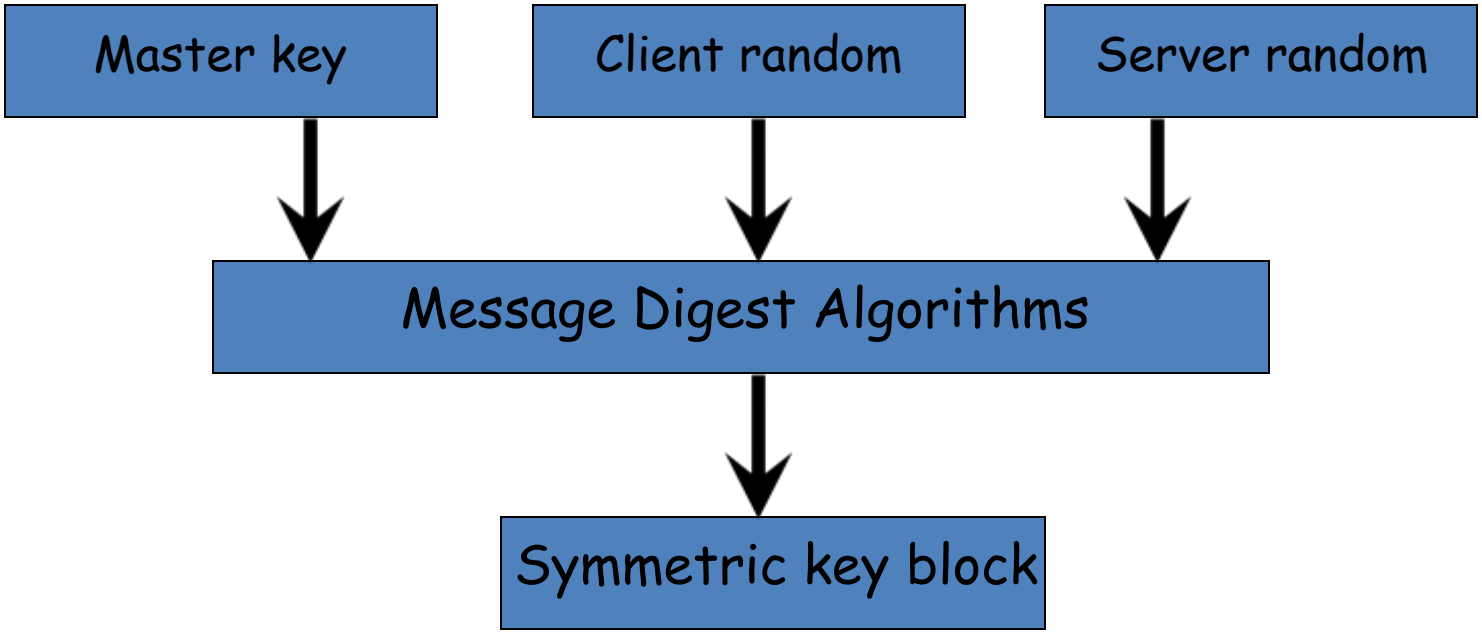
Computing the master key from the Pre-master key

The three words “A”, “BB” and “CCC” are also given as input values here



Computing other keys and IV's from the master key

The three words “A”, “BB” and “CCC” are also given as input values here



Symmetric key block = client write MAC secret, server write MAC secret, client write key, server write key, client write IV, and server write IV

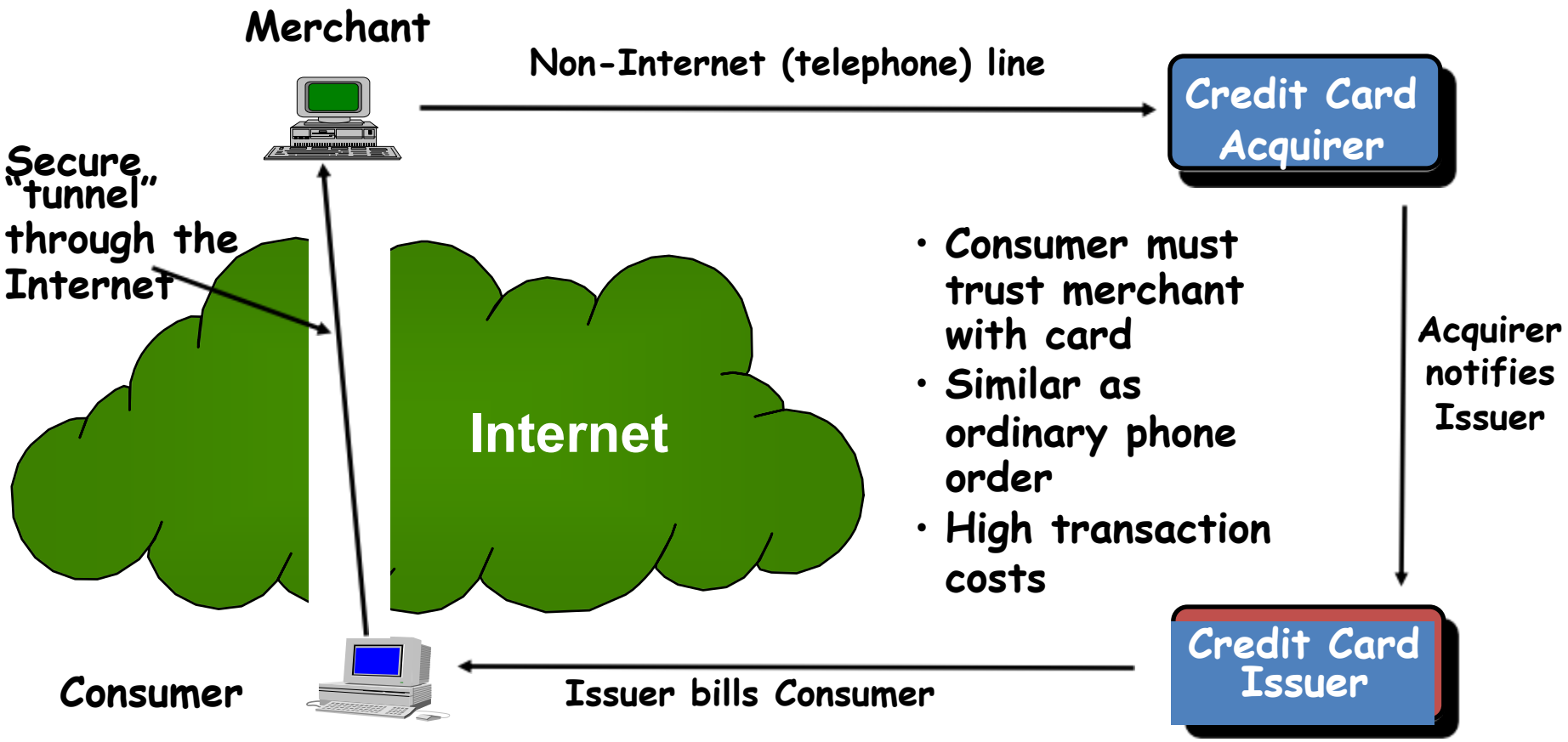
Details Omitted in the Handshake Protocol

- Pre-master key exchange methods:
  - RSA: A 48-byte pre-master key generated by client, and encrypted with the server's public key. The encrypted one is sent to server.
  - Diffie-Hellman: (three variants of DH) omitted.
- Cipher algorithm: RC4, RC2, DES, 3DES, AES, ...
- Server authentication: (using digital signature)
- Client authentication: (using digital signature)

SSL Applications

By design, SSL can be used to protect any TCP-based application protocol data.
In practice, SSL is mainly used to protect Web data.

The Main Usage of SSL



Transport Layer Security (Protocol)

- Similar as SSLv3.
- Differences in the:
  - Version number
  - Message authentication code
  - Alert codes
  - Cipher suites
  - Cryptographic computations
  - Padding
  - Etc.

Recommended Reading

- W. Stallings, Cryptography and Network Security, 7th Edition, Prentice Hall
- B.A. Forouzan, Cryptography and Network Security, McGraw-Hill.
- The SSL Protocol Version 3.0 Transport Layer Security Working Group RFC-2246
- The TLS Protocol Version 1.0 RFC 2246 https://datatracker.ietf.org/doc/rfc2246/
- OpenSSL website: www.openssl.org