WEB Security: Secure Socket Layer

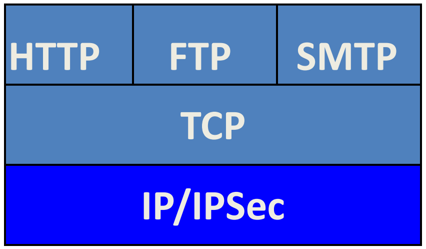
Cunsheng Ding

HKUST, Hong Kong, CHINA

Outline of this Lecture

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

Security Facilities in the TCP/IP Protocol Stack

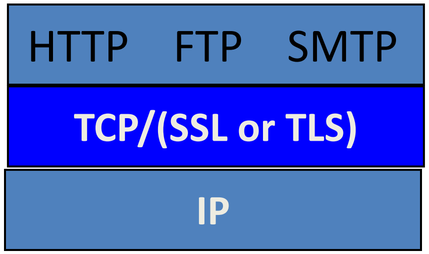


HTTP FTP SMTP

TCP

IP/IPSec

(a) Network level

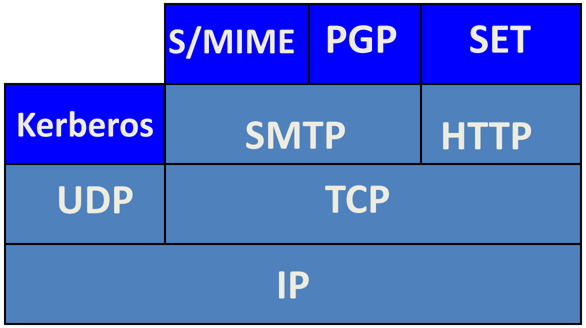


HTTP FTP SMTP

TCP/(SSL or TLS)

IP

(b) Transport level



S/MIME PGP SET

Kerberos SMTP HTTP

UDP TCP

IP

(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

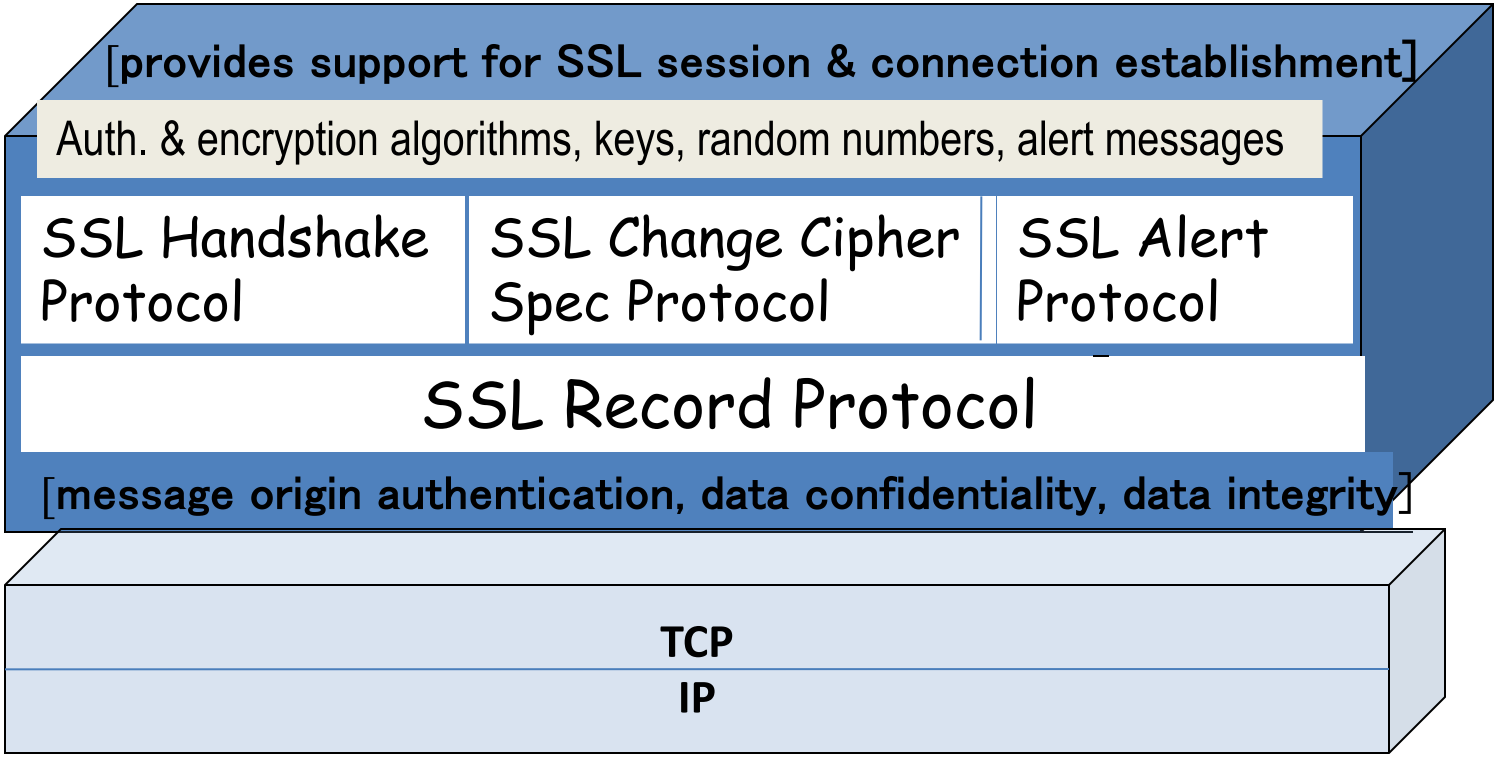
SSL Services

- 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.


[provides support for SSL session & connection establishment]
Auth. & encryption algorithms, keys, random numbers, alert messages
SSL Handshake Protocol
SSL Change Cipher Spec Protocol
SSL Alert Protocol
SSL Record Protocol
[message origin authentication, data confidentiality, data integrity]
TCP
IP

SSL Protocol

Components:

- SSL Record Protocol
  - Layered on top of a connection-oriented and reliable transport layer service
  - 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.
  - Is is used to avoid the expensive negotiation of new security parameters for each connection.

SSL State Information

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

SSL Session State Information Elements

- **Session ID**: An arbitrary byte sequence chosen by the server to identify an active or resumable session state.
- **Peer certificate**: X509.v3 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 secret**: 48-byte secret shared between client and server
- **Is resumable**: flag that indicates whether the session can be used to initiate new connections

More on SSL Session State

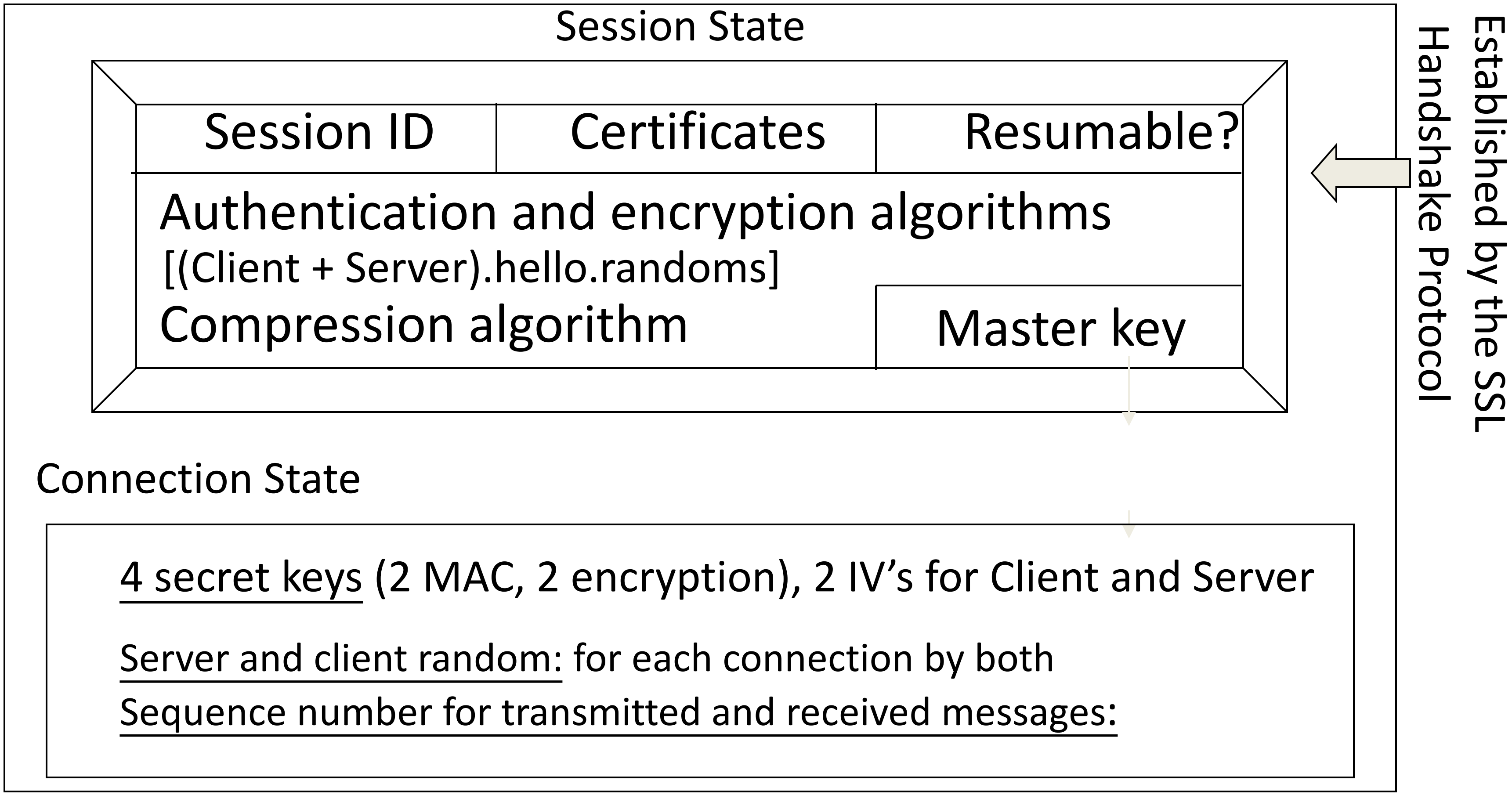
- A previous session may be resumed (use Session ID and its session cache)
- A new session may be negotiated

  (use Session ID and the Handshake Protocol)

SSL Connection State Information Elements

- Server and client random: byte sequences that are chosen by server and client for each connection.
- Server write MAC secret: secret used for MAC on data written by server
- Client write MAC secret: secret used for MAC on data written by client [**different from server write MAC key**]
- Server write key: key used for data encryption by server and decryption by client
- Client write key: key used for encryption by client and decryption by 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



Session State

Session ID

Certificates

Resumable?

Authentication and encryption algorithms

[(Client + Server).hello.randoms]

Compression algorithm

Master key

Established by the SSL Handshake Protocol

Connection State

4 secret keys (2 MAC, 2 encryption), 2 IV's for Client and Server

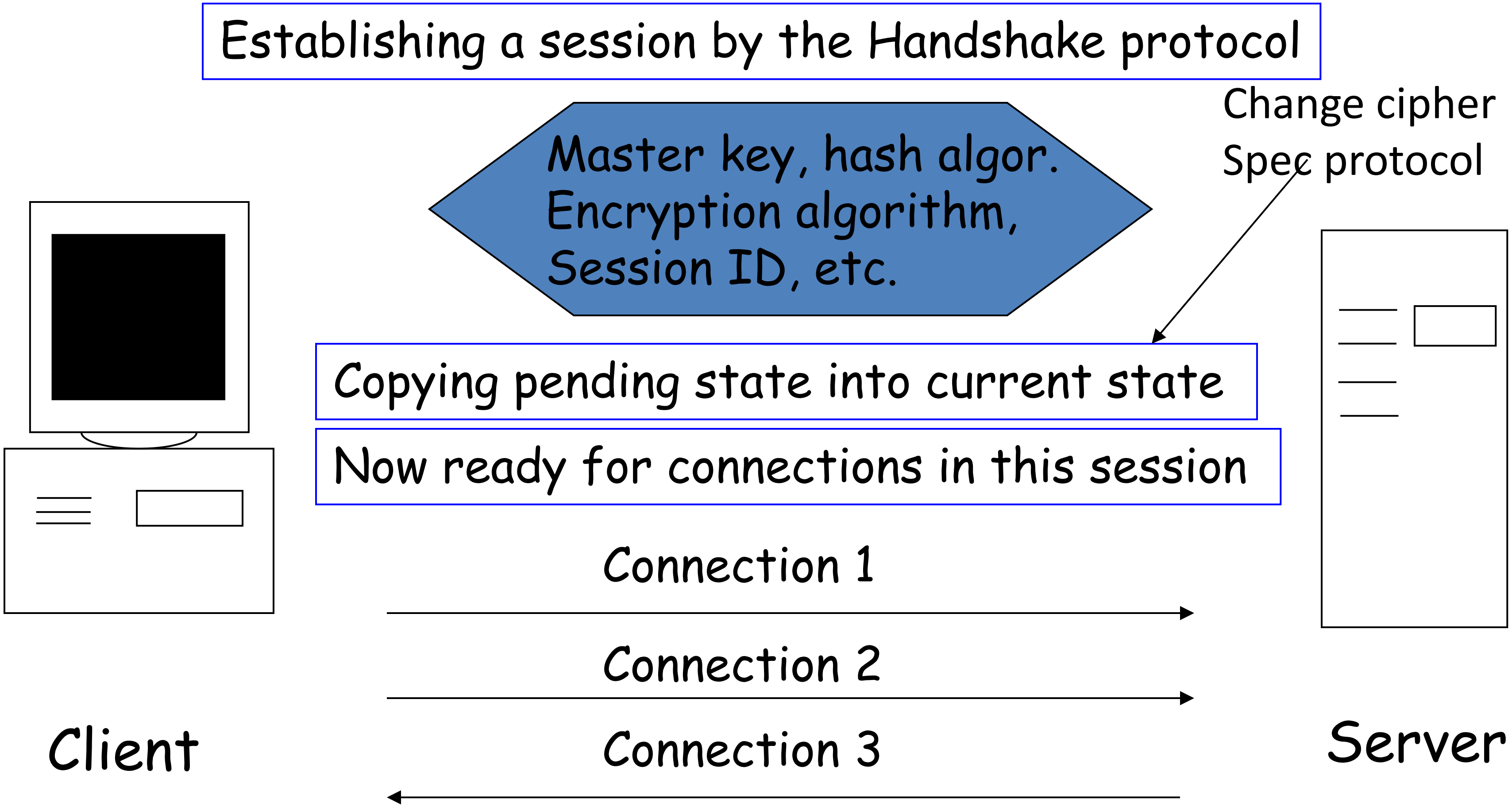
Server and client random: for each connection by both

Sequence number for transmitted and received messages:

Current and Pending State

- **Current state**: There is a current operating state for both read and write (i.e., receive and send).
- **Pending state**: In addition, during the Handshake Protocol, pending read and write states are created.
- **Updating**: Upon successful conclusion of the Handshake protocol, the pending states become the current states.

Connection and Session



Establishing a session by the Handshake protocol

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

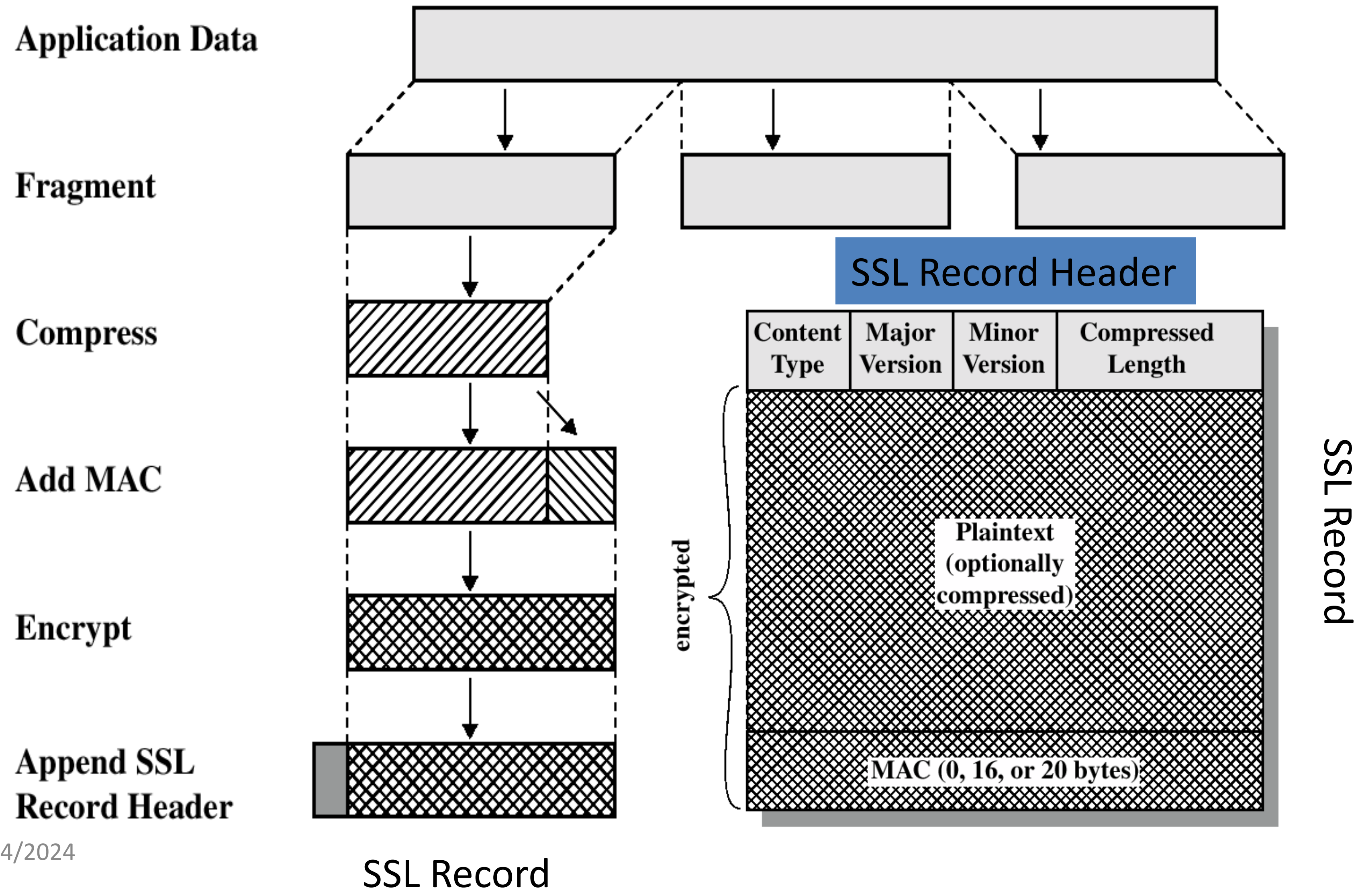
Change cipher
Spec protocol

Copying pending state into current state

Now ready for connections in this session

SSL Record Protocol

SSL Record Protocol Operation



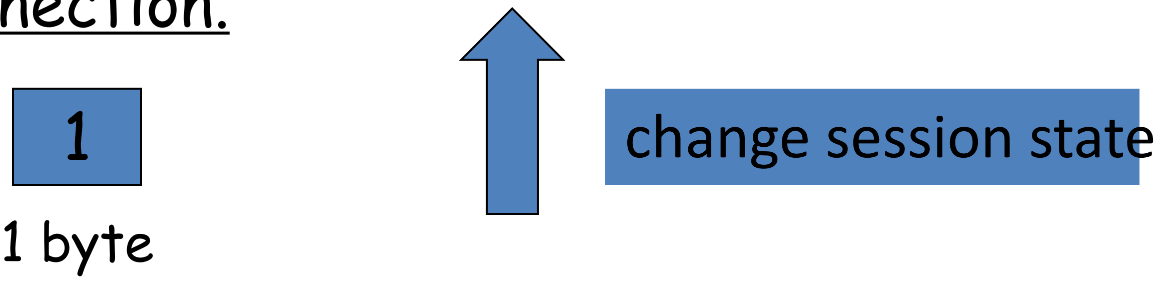
SSL Record Content

- **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 state to be copied into the current state, which updates the cipher suite to be used on this connection.



Alert Protocol

Alert Protocol

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

Level	Alert
1 byte	1 byte

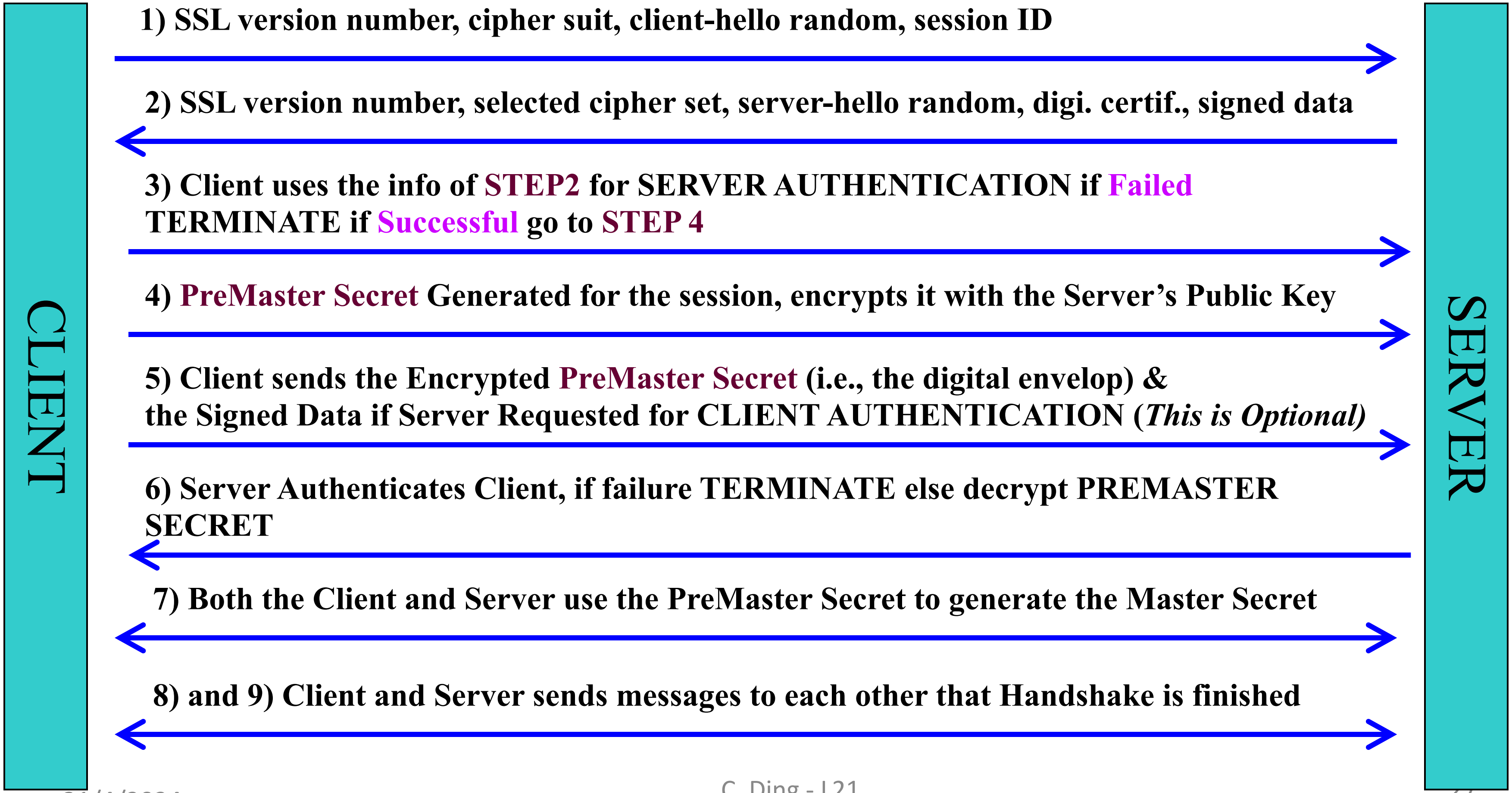
<==> errors occurred during handshaking
<=== errors occurred during processing at the sever

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



1) SSL version number, cipher suit, client-hello random, session ID

2) SSL version number, selected cipher set, server-hello random, digi. certif., signed data

3) Client uses the info of STEP2 for SERVER AUTHENTICATION if Failed TERMINATE if Successful go to STEP 4

4) PreMaster Secret Generated for the session, encrypts it with the Server's Public Key

5) Client sends the Encrypted PreMaster Secret (i.e., the digital envelop) & the Signed Data if Server Requested for CLIENT AUTHENTICATION (*This is Optional)*

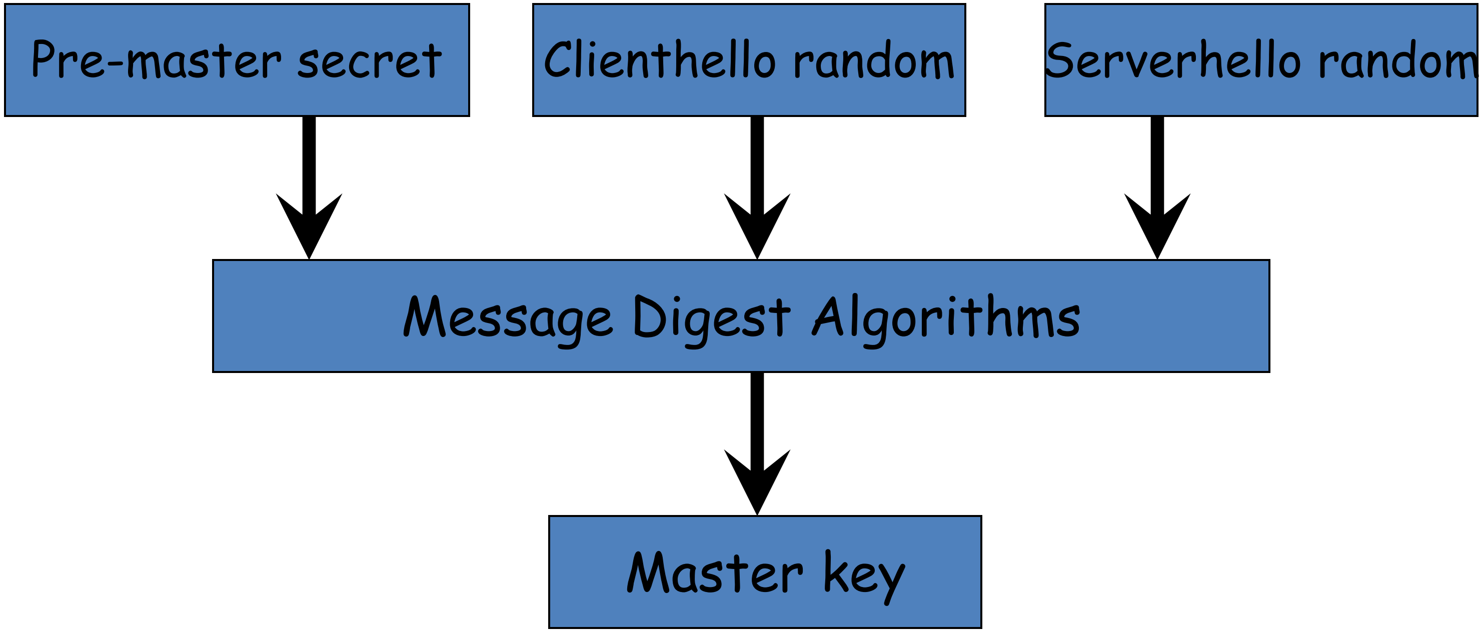
6) Server Authenticates Client, if failure TERMINATE else decrypt PREMASTER SECRET

7) Both the Client and Server use the PreMaster Secret to generate the Master Secret

8) and 9) Client and Server sends messages to each other that Handshake is finished

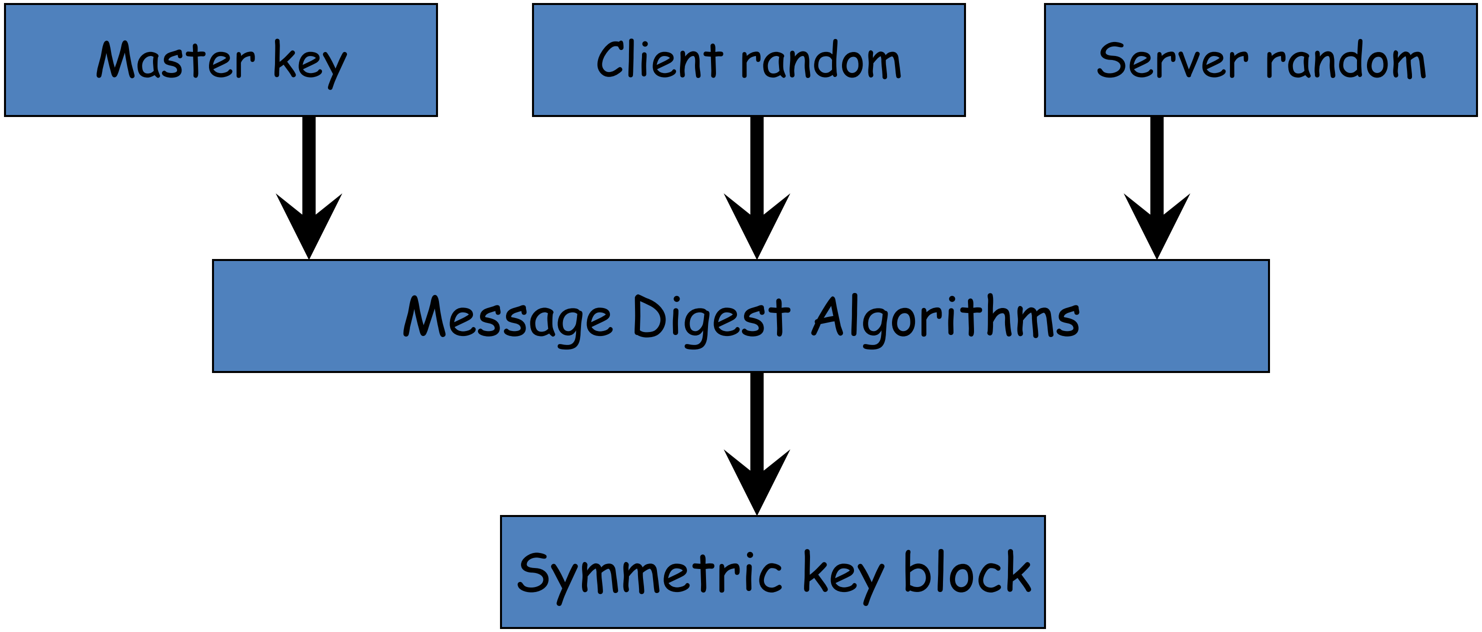
Pre-master secret, master secret, and symmetric key

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



Pre-master secret, master secret, and symmetric key

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



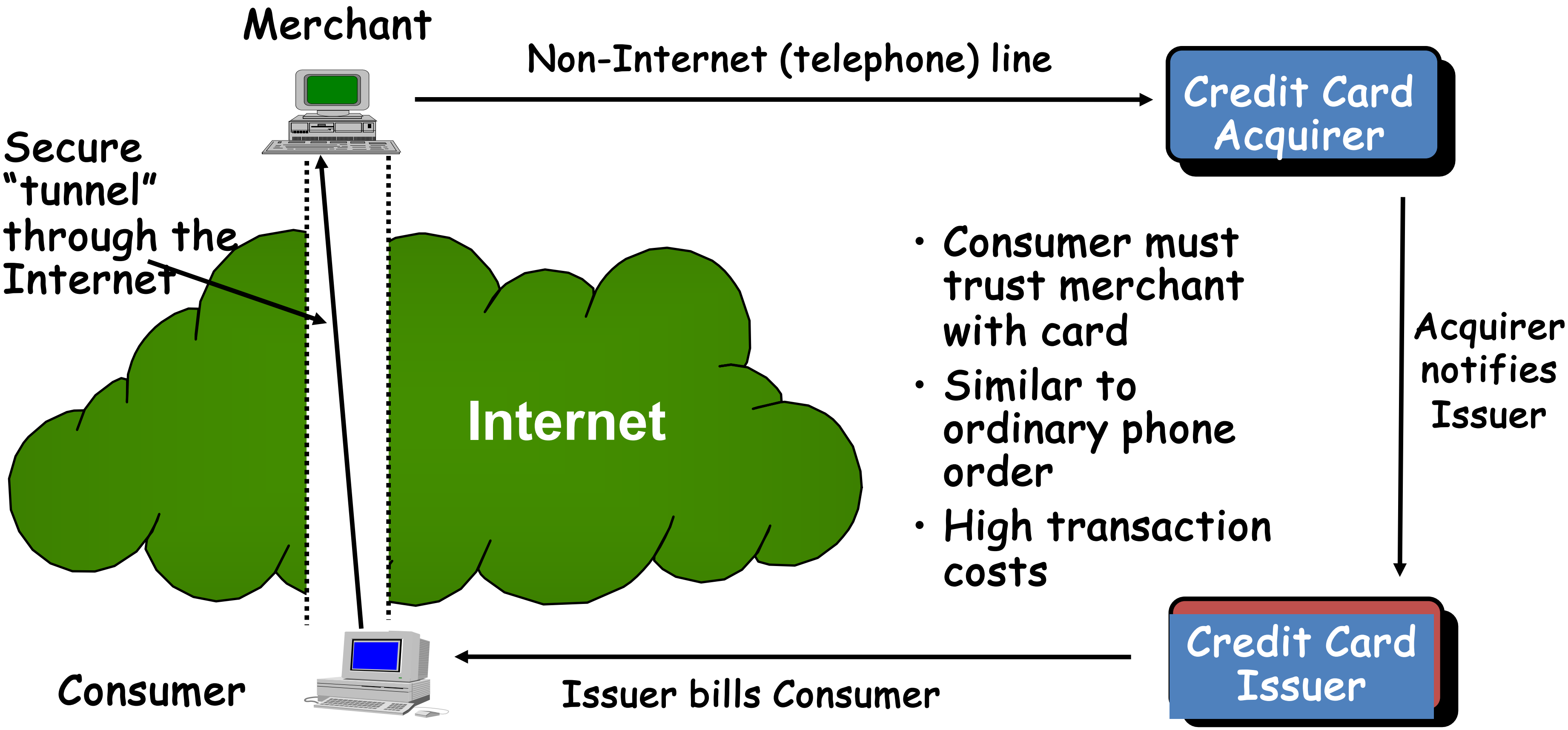
Symmetric key block = client write MAC secret, server write MAC secret, client write key, server write key, client write IV, and server write IV

Details Omitted in the Handshake Protocol

- Pre-master secret exchange methods:
  - RSA: A 48-byte pre-master key generated by client, and encrypted by 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 certificates)
- Client authentication: (using digital certificates)

SSL Applications

The Main Usage of SSL



Merchant

Non-Internet (telephone) line

Credit Card Acquirer

Secure "tunnel" through the Internet

Internet

- Consumer must trust merchant with card
- Similar to ordinary phone order
- High transaction costs

Acquirer notifies Issuer

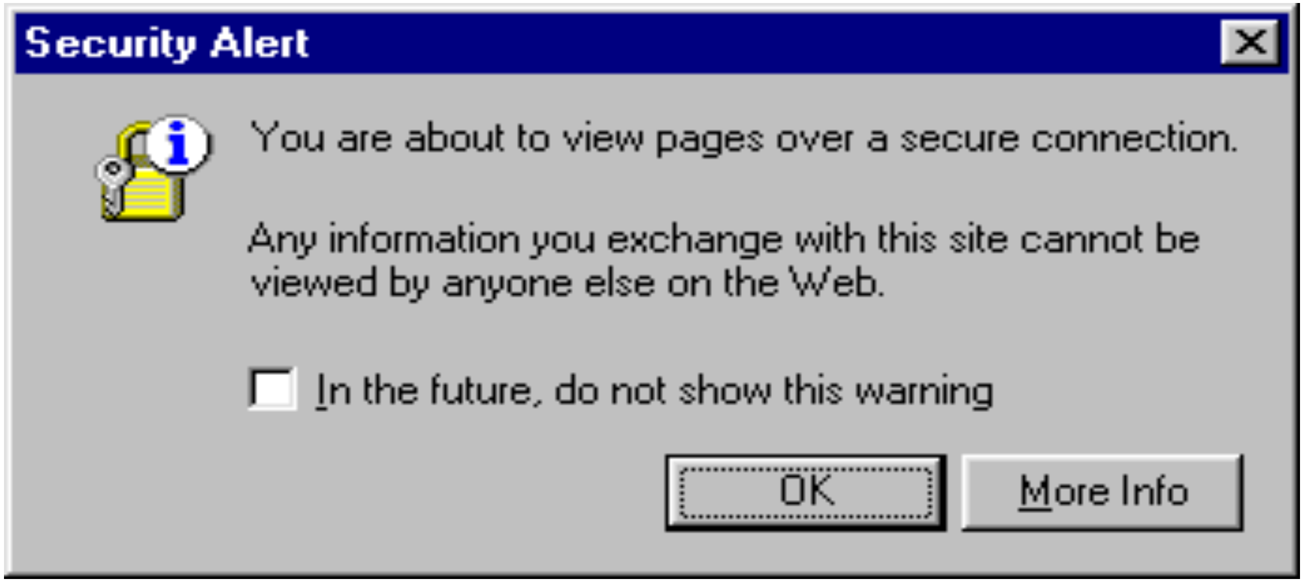
Credit Card Issuer

Consumer

Issuer bills Consumer

The Main Usage of SSL

After the SSL Handshaking



Security Alert

You are about to view pages over a secure connection.

Any information you exchange with this site cannot be viewed by anyone else on the Web.

□ In the future, do not show this warning

"If you want people to buy from your site, you must provide an order form with Secure Sockets Layer (SSL) encryption technology"

O'Brien (2000)

Transport Layer Security (Protocol)

- Similar to SSLv3.
- Differences in the:
  - version number
  - message authentication code
  - pseudorandom function
  - alert codes
  - cipher suites
  - client certificate types
  - certificate_verify and finished message
  - cryptographic computations
  - padding

Recommended Reading

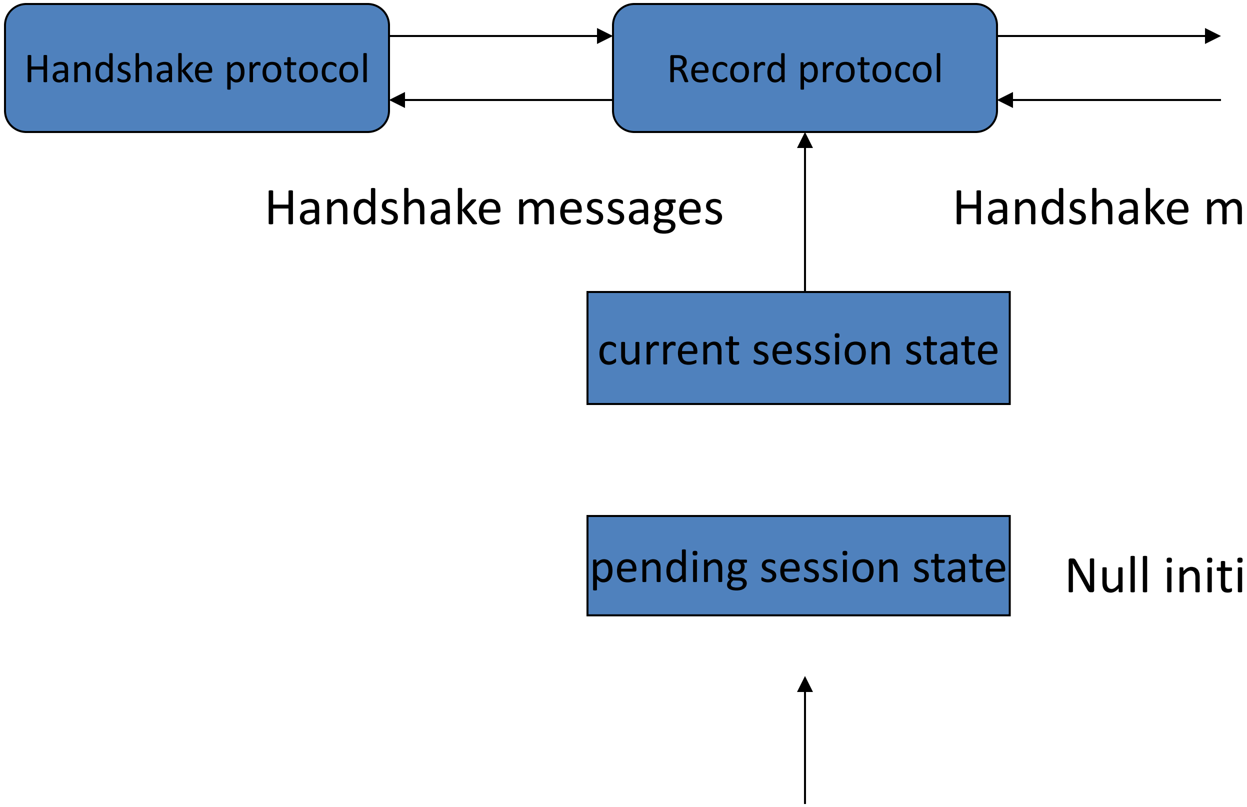
- W. Stallings, Cryptography and Network Security, 2nd, 3rd Edition, Prentice Hall
- B.A. Forouzan, Cryptography and Network Security, McGraw-Hill.
- Garfinkel, S., and Spafford, G. Web Security & Commerce. O'Reilly and Associates, 1997
- The SSL Protocol Version 3.0 Transport Layer Security Working Group RFC-2246
- http://wp.netscape.com/eng/ssl3/ssl-toc.html
- The TLS Protocol Version 1.0 RFC 2246

  https://datatracker.ietf.org/doc/rfc2246/
- OpenSSL website: www.openssl.org

Appendix

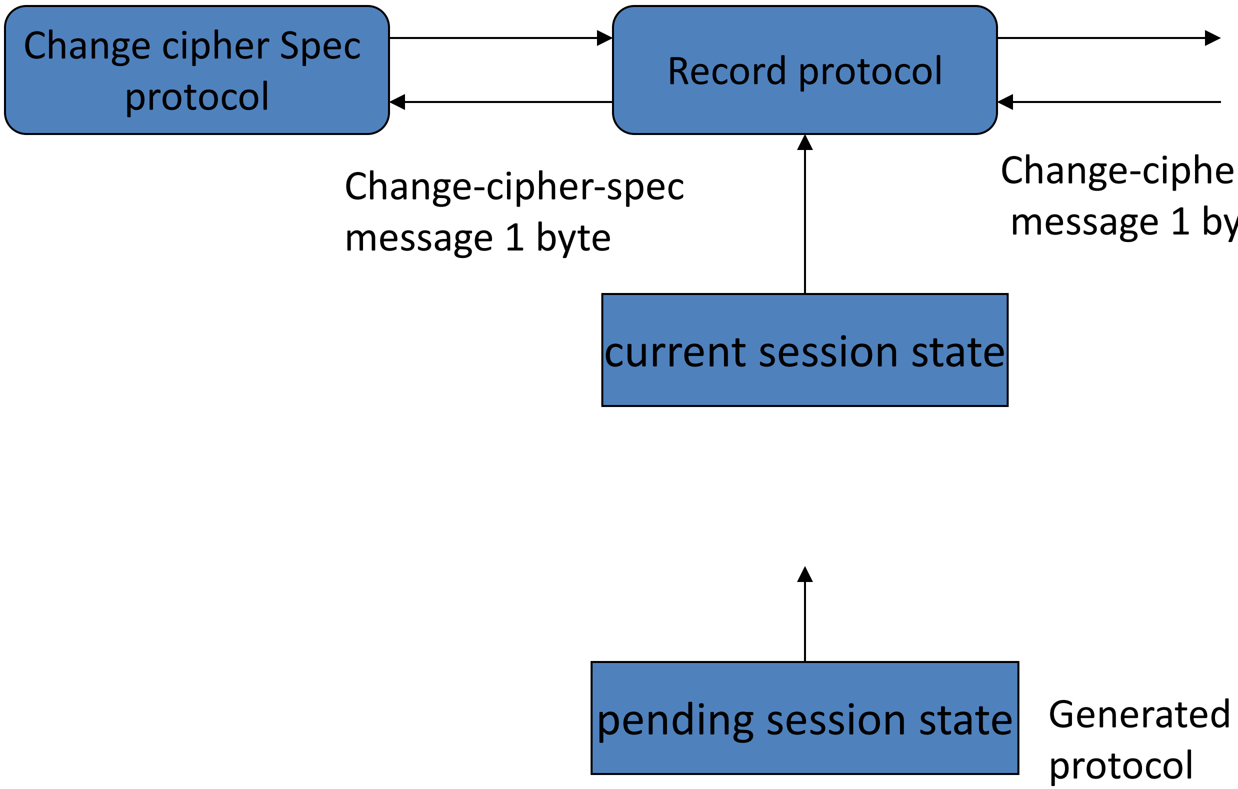
Pictorial description of SSL protocols

SSL Procedure: Protocol 1



After handshaking, pending state is produced

SSL Procedure: Protocol 2



Change cipher Spec protocol

Record protocol

Change-cipher-spec message 1 byte

Change-cipher-spec message 1 byte

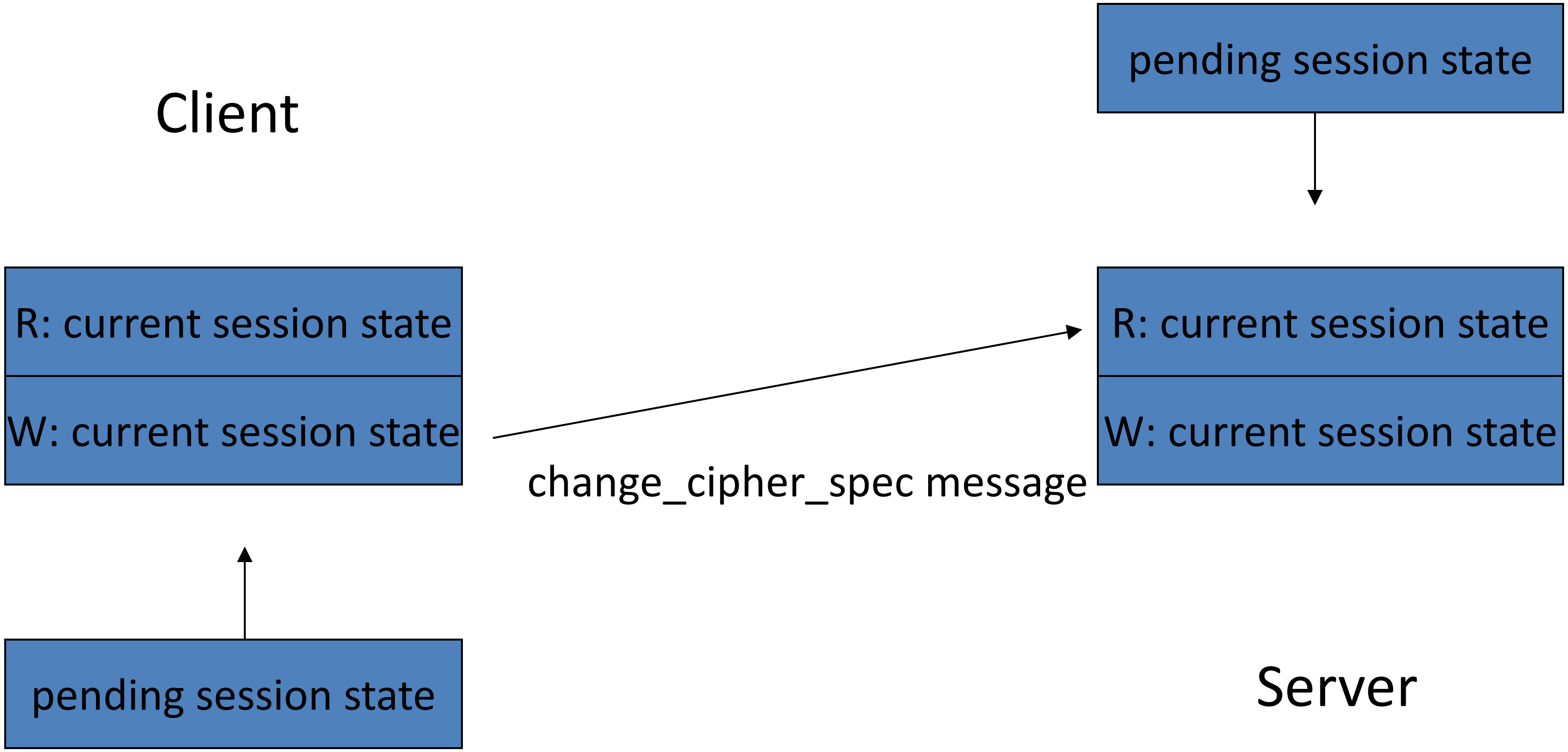
current session state

pending session state

Generated earlier by handshaking protocol

Copy the pending state into current state, after finishing the change cipher protocol

SSL Procedure: Protocol 2 more detailed information



Client

pending session state

R: current session state

W: current session state

change_cipher_spec message

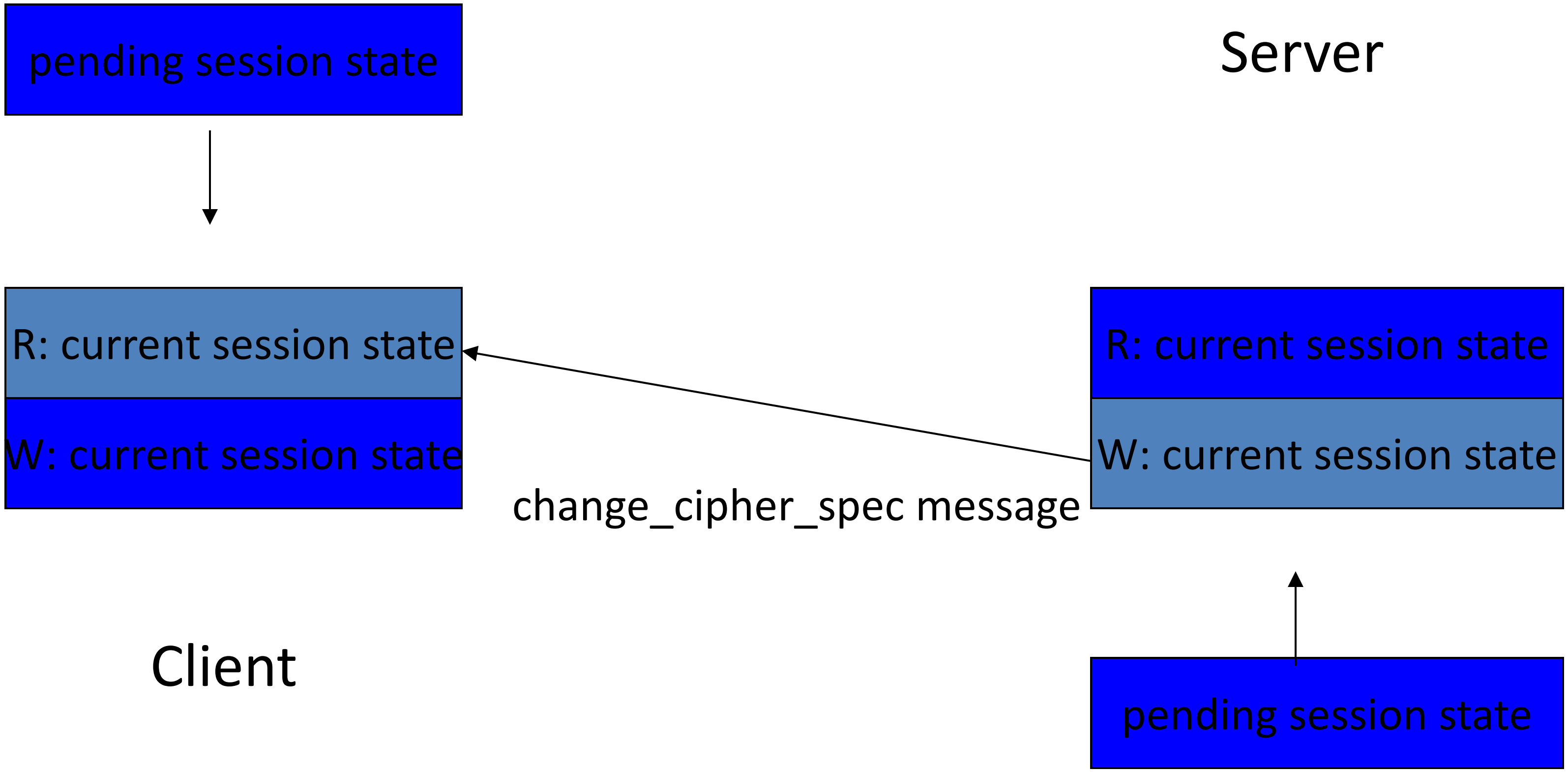
R: current session state

W: current session state

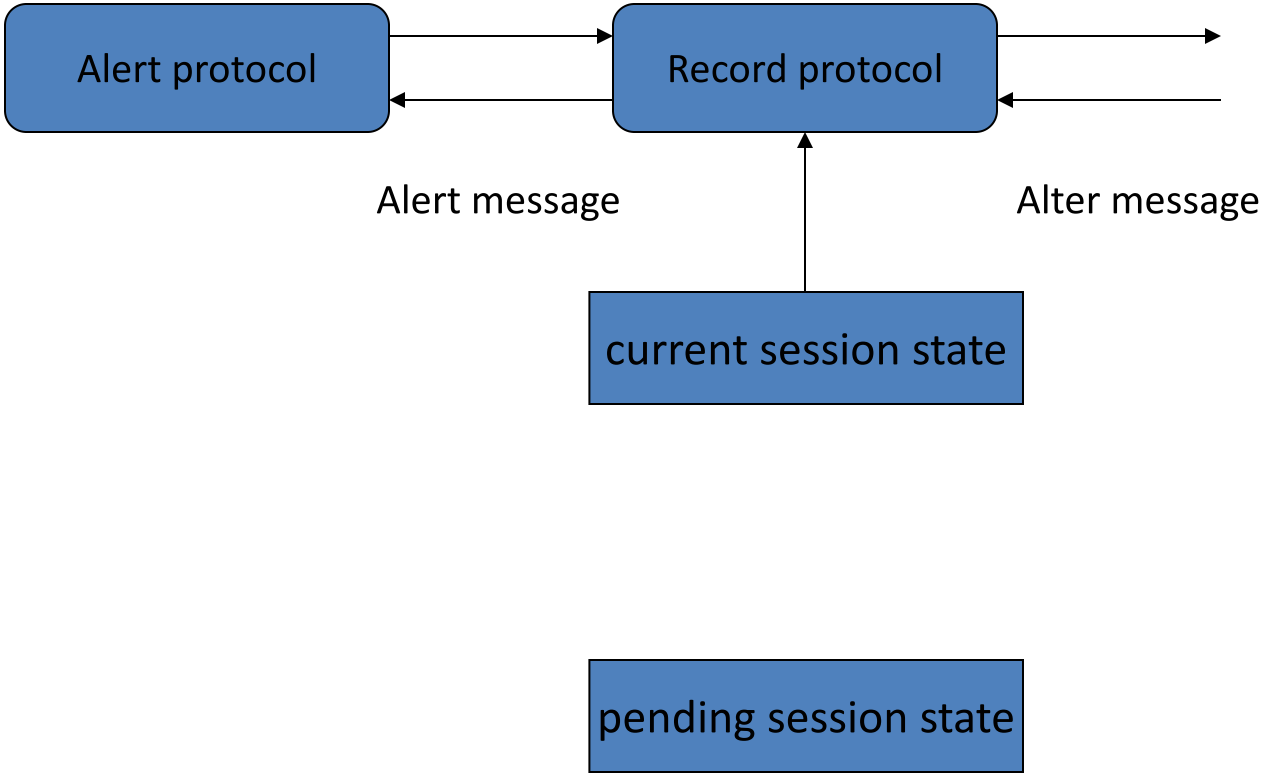
pending session state

Server

SSL Procedure: Protocol 2 more detailed information



SSL Procedure: Protocol 3



Alert protocol

Record protocol

Alert message

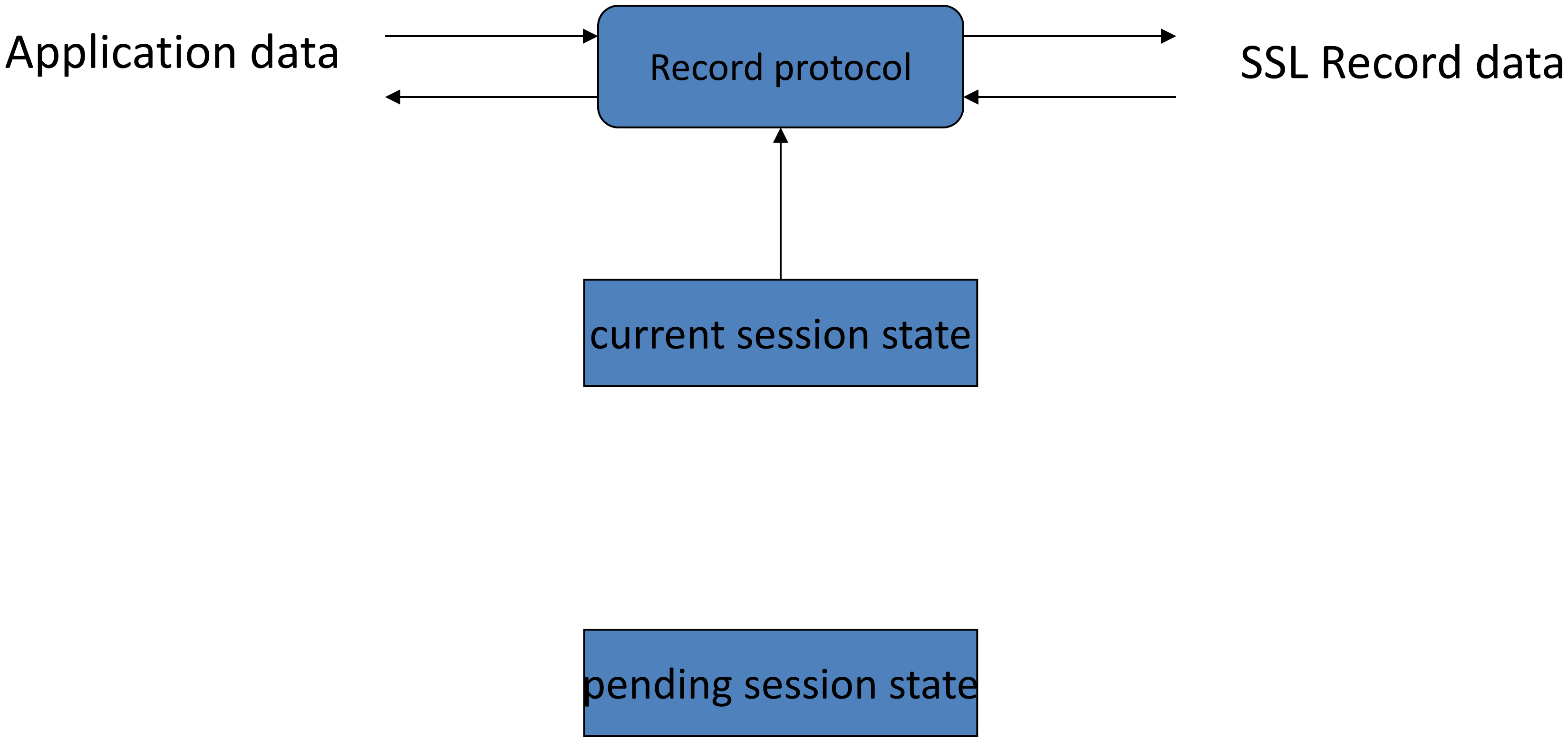
Alter message

current session state

pending session state

After this protocol, whether this connection should be terminated

SSL Procedure: Protocol 4


Application data
Record protocol
SSL Record data
current session state
pending session state