Applied!

Data & Network Security

Behnam Amiri

ans.dailysec.ir

aNetSec.github.io

Spring 2025

SSL/TLS

Website Security

- HTTP protocol has no built in encryption!
- If you browse a website some body can sniff it!
- MiTM Attack is possible.

A Comparison of Threats on the Web

	Threats	Consequences	Countermeasures
Integrity	• Modification of user data • Trojan horse browser • Modification of memory • Modification of message traffic in transit	• Loss of information • Compromise of machine • Vulnerability to all other threats	Cryptographic checksums
Confidentiality	• Eavesdropping on the net • Theft of info from server • Theft of data from client • Info about network configuration • Info about which client talks to server	• Loss of information • Loss of privacy	Encryption, Web proxies
Denial of Service	• Killing of user threads • Flooding machine with bogus requests • Filling up disk or memory • Isolating machine by DNS attacks	• Disruptive • Annoying • Prevent user from getting work done	Difficult to prevent
Authentication	• Impersonation of legitimate users • Data forgery	• Misrepresentation of user • Belief that false information is valid	Cryptographic techniques

TLS Record Protocol Operation

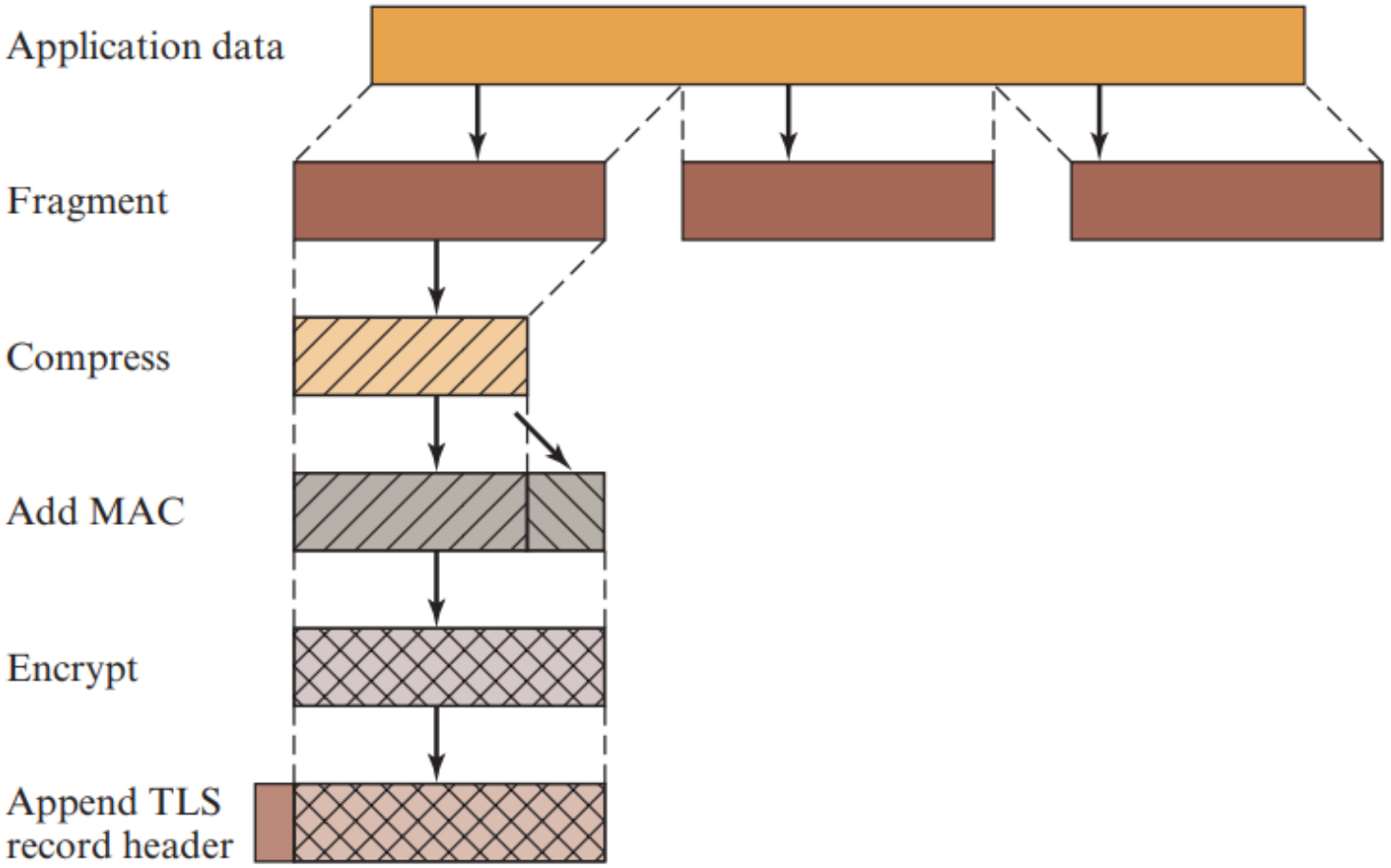
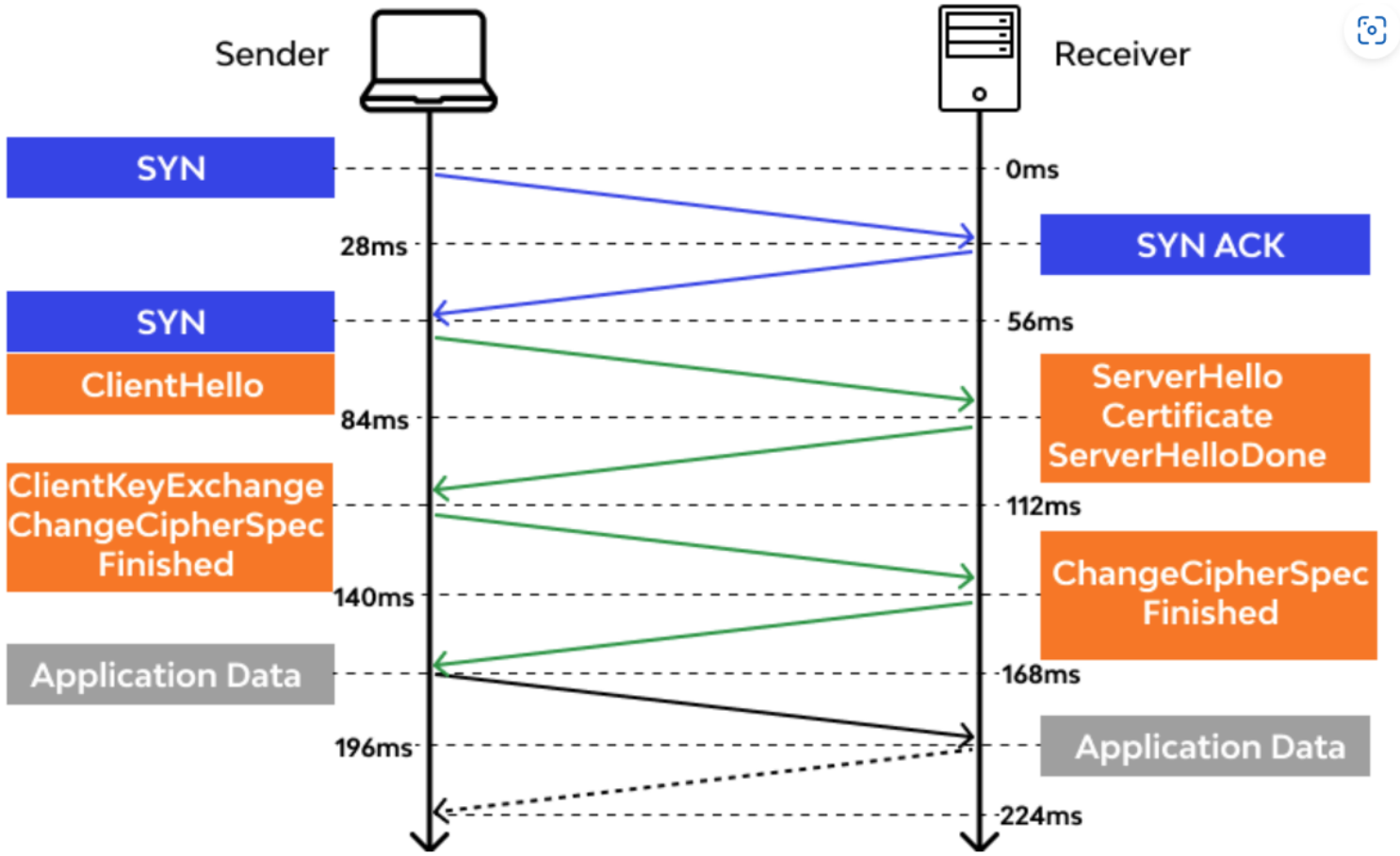
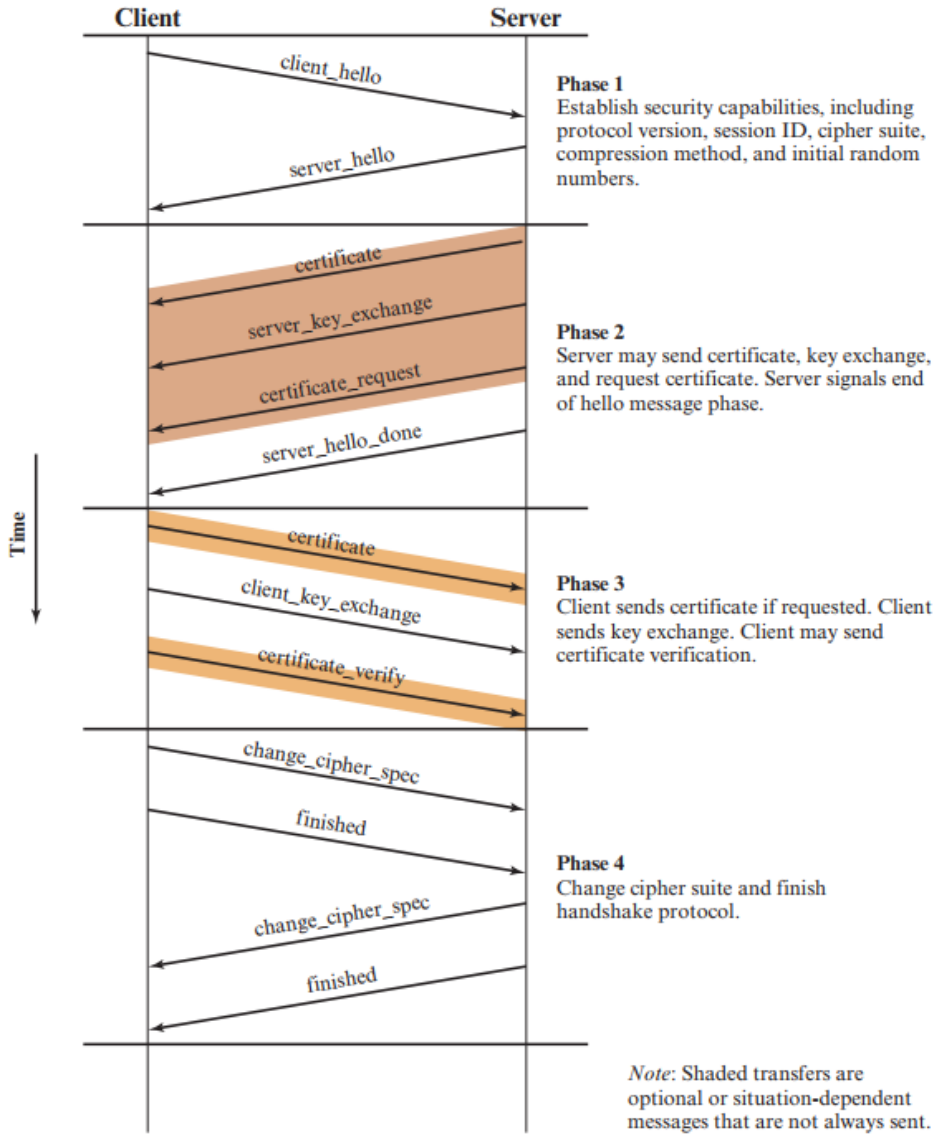


Figure 17.3 TLS Record Protocol Operation

TLS


Sender
Receiver
SYN
0ms
28ms
SYN ACK
SYN
56ms
ClientHello
ServerHello
Certificate
ServerHelloDone
84ms
ClientKeyExchange
ChangeCipherSpec
Finished
112ms
ChangeCipherSpec
Finished
140ms
Application Data
168ms
196ms
Application Data
224ms

Handshake Protocol Action



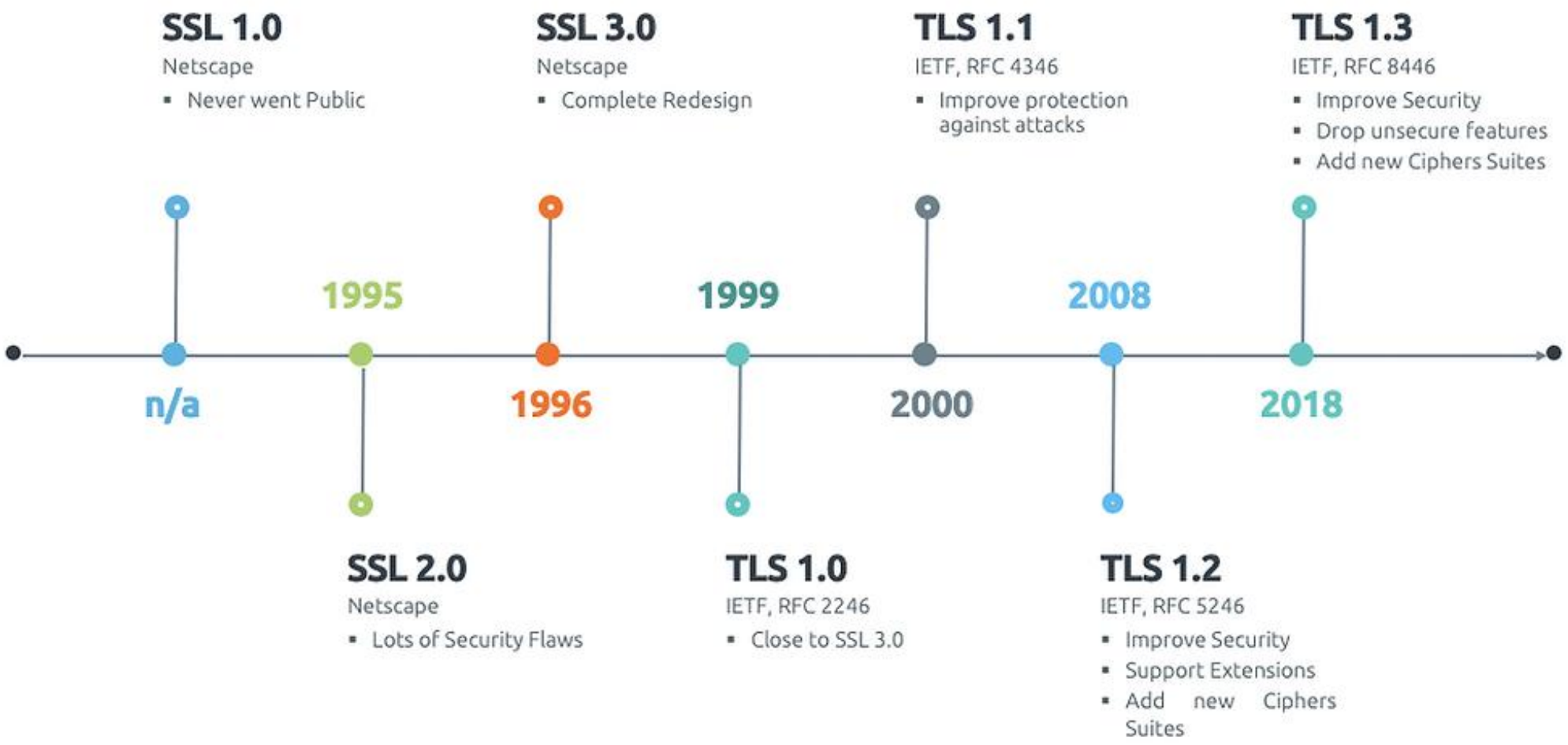
TLS Handshake Protocol Message Types

Message Type	Parameters
hello_request	null
client_hello	version, random, session id, cipher suite, compression method
server_hello	version, random, session id, cipher suite, compression method
certificate	chain of X.509v3 certificates
server_key_exchange	parameters, signature
certificate_request	type, authorities
server_done	null
certificate_verify	signature
client_key_exchange	parameters, signature
finished	hash value

Why Hello?

- Why client & server send hello and don’t use fixed protocols?

SSL/TLS History

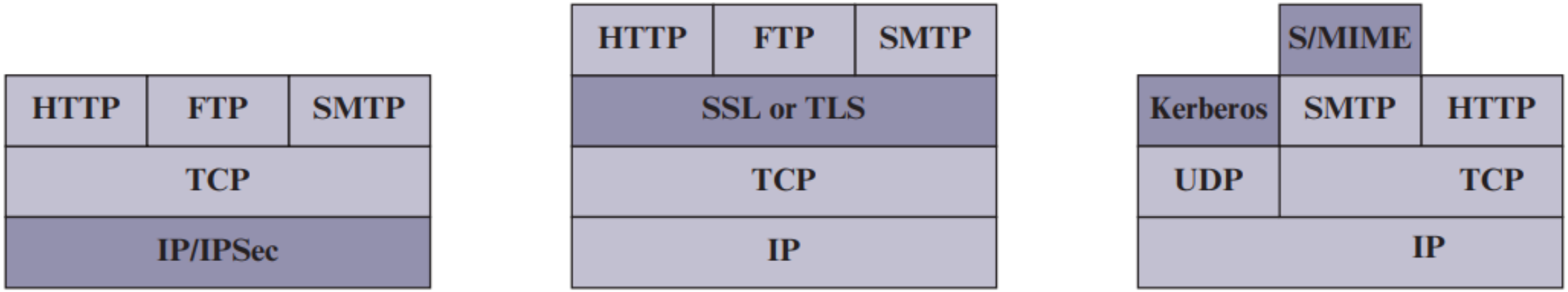


SSL/TLS Usage

SSL	
SSL Version	**Status**
SSL 1.0	Never Released
SSL 2.0	Dead/Deprecated
SSL 3.0	Dead/Deprecated

TLS	
TLS Version	**Status**
TLS 1.0	Dead/Deprecated
TLS 1.1	Dead/Deprecated
TLS 1.2	Currently Used
TLS 1.3	Currently Used

Relative Location of Security Facilities in the TCP/IP Protocol Stack



(a) Network level

(b) Transport level

(c) Application level

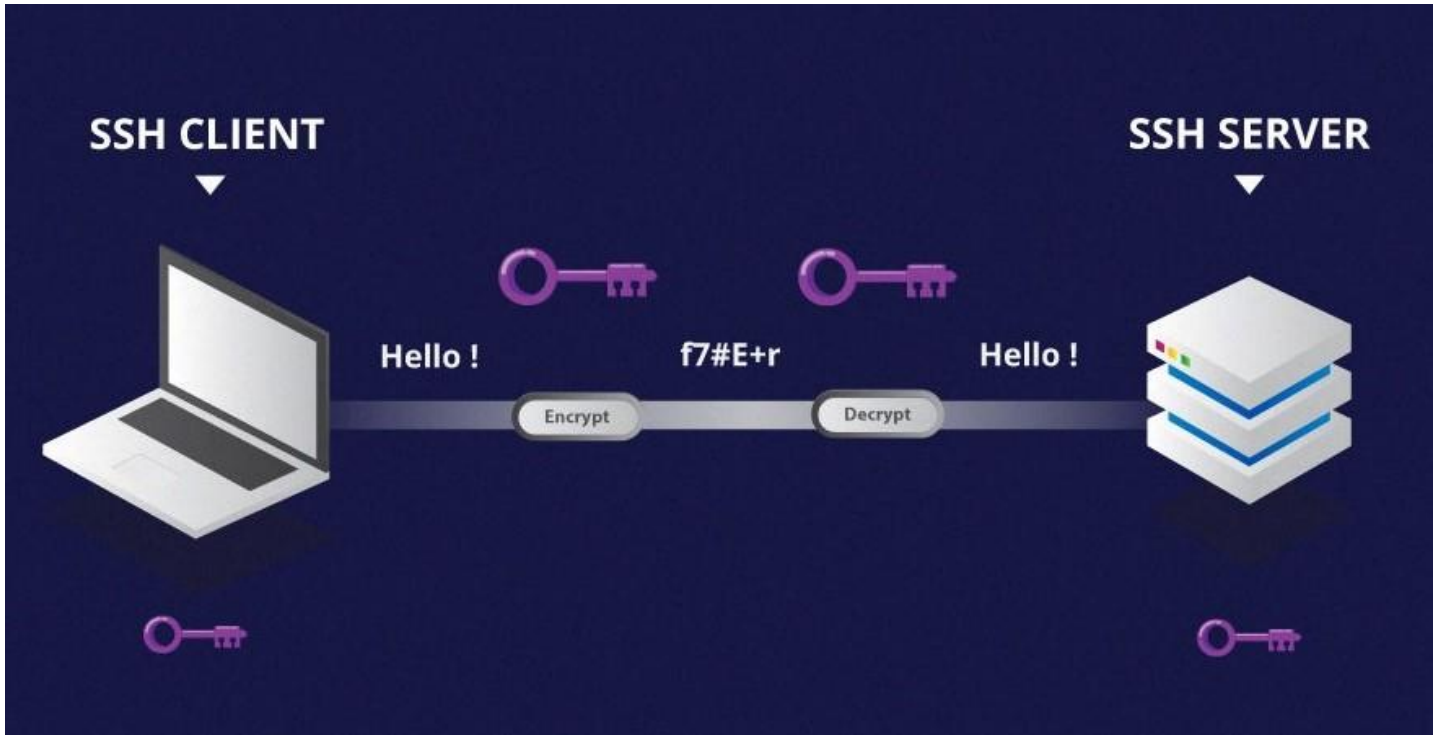
SSH

Remote Access

- We need remote access protocols to access remote servers.
- Telnet, SSH, RDP, VNC, … are remote protocols.
- Telnet, SSH are command line protocols.
- RDP, VNC hash GUI.

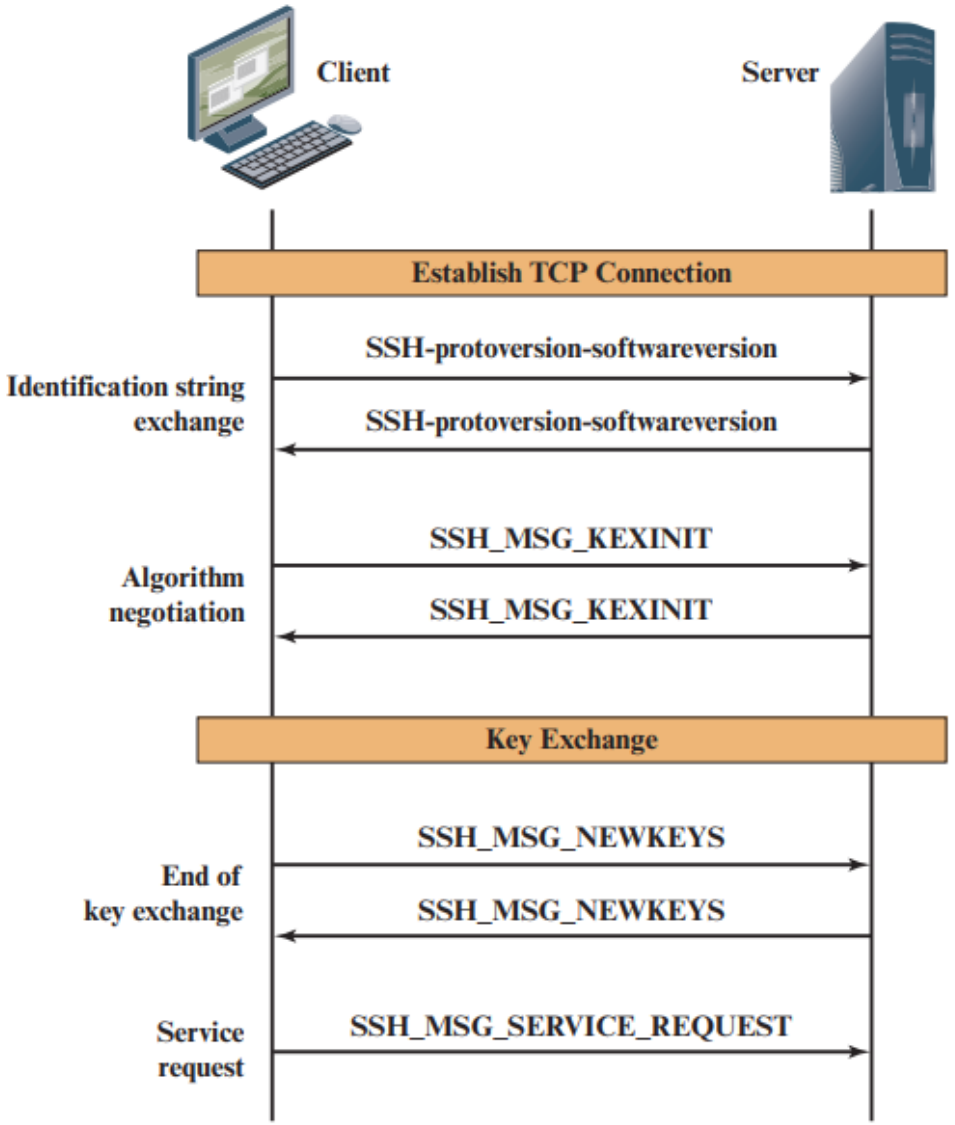
Secure Shell - SSH

- SSH is a cryptographic network protocol that allows secure access to a computer over an unsecured network.



SSH Demo

SSH Transport Layer Protocol Packet Exchanges



Connection via SSH tunnel

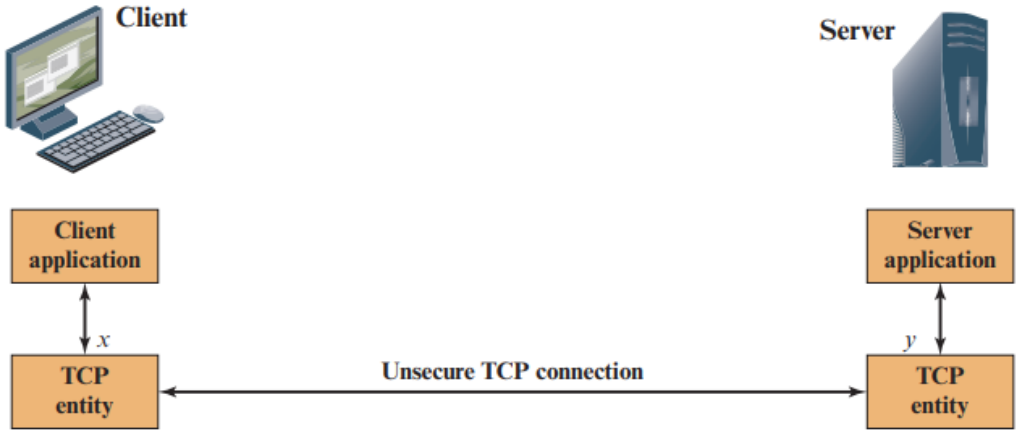
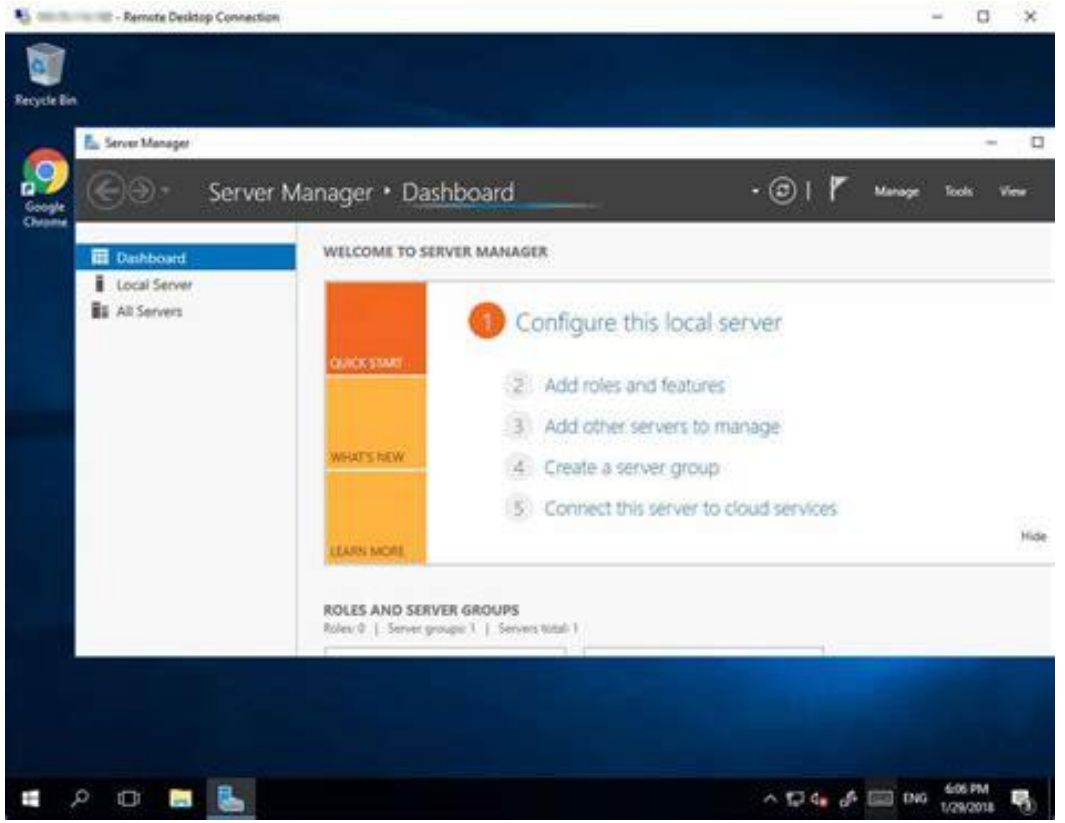


Figure 17.12 SSH Transport Layer Packet Exchanges

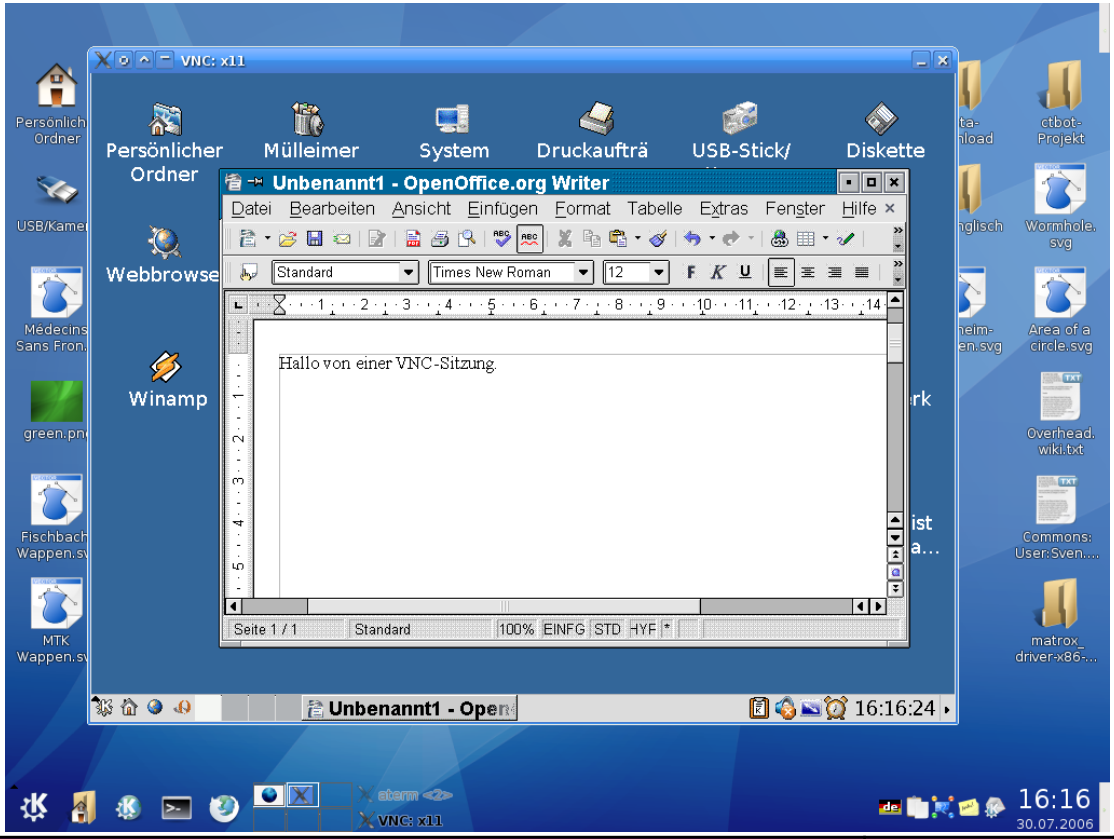
RDP: Remote Desktop Connection

- RDP is for windows OS.
- Provides a user with GUI to another computer over a network.
- We need RDP client & RDP server.
- RDP default port is 3389



VNC: Virtual Network Computing

- Is a graphical desktop-sharing system to remotely control another computer.
- VNC works on Linux and Windows.



Security

- RDP: Built-in encryption and security features
- VNC: No built-in encryption; requires secure tunneling (e.g., SSH)
- MiTM Attack is possible for both.
- There are other remote protocols.