Mobile Phone Security

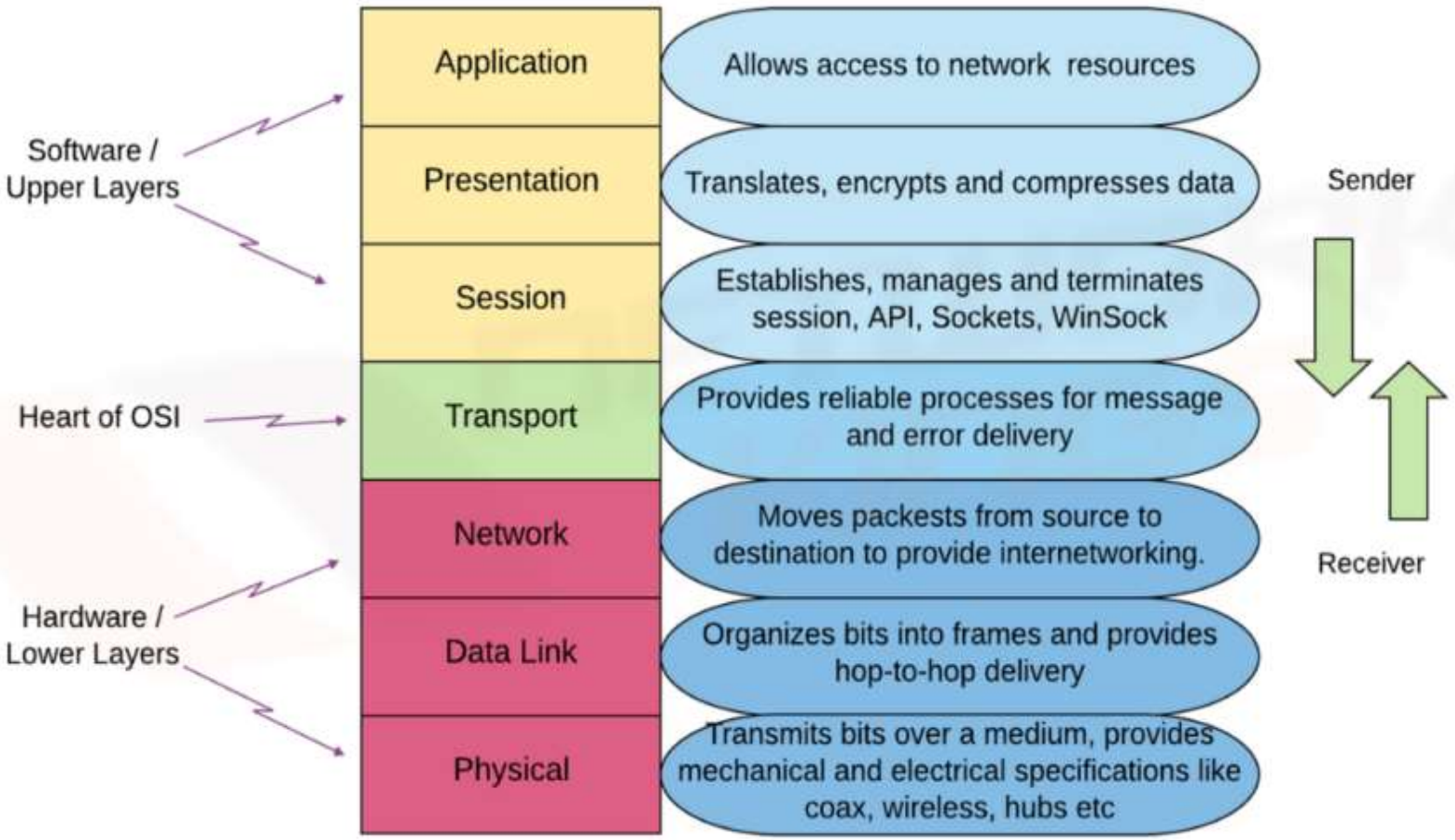
Dr. Digvijaysinh Rathod
Associate Professor
School of Cyber Security and Digital Forensics
National Forensic Sciences University

digvijay.rathod@nfsu.ac.in

HTTP

(Hyper Text Transfer Protocol)

OSI Model


Application
Presentation
Session
Transport
Network
Data Link
Physical
Allows access to network resources
Translates, encrypts and compresses data
Establishes, manages and terminates session, API, Sockets, WinSock
Provides reliable processes for message and error delivery
Moves packests from source to destination to provide internetworking.
Organizes bits into frames and provides hop-to-hop delivery
Transmits bits over a medium, provides mechanical and electrical specifications like coax, wireless, hubs etc
Software / Upper Layers
Heart of OSI
Hardware / Lower Layers
Sender
Receiver

TCP IP Model

#	Layer Name	Protocol	Protocol Data Unit	Addressing
5	Application	HTTP, SMTP, etc..	Messages	n/a
4	Transport	TCP/UDP	Segment	Port #'s
3	Network	IP	Datagram	IP address
2	Data Link	Ethernet, Wi-Fi	Frames	MAC Address
1	Physical	10 Base T, 802.11	Bits	n/a

HTTP Web Basics:

- ✓ The Hyper Text Transfer Protocol is a Stateless Protocol, TCP / IP based protocol used for communicating on the WWW.
- ✓ HTTP defines the precise manner in which Web clients communicate with Web Servers.
- ✓ HTTP Version 1.0 is the most commonly used version but now many Browsers and Web Servers now support HTTP 1.1 Version because it is standardized.

Understanding HTTP Working:

- ✓ The HTTP follows a very simple Request–Response paradigm (i.e. architecture, standard).
- ✓ A conversation between a Client (Web Browser) and a Server (Web Server) goes something like this:

1. the Client opens a connection (TCP/IP) to the Server,
2. the Client makes a request to the Server,
3. the Server responds to the request, and
4. the Connection is closed.

Understanding HTTP Working:

1. At this stage there is not a much of HTTP specific activity takes place because TCP/IP connection is established at the Transport Layer of the Protocol Stack.

2. The Client's request of **http://www.awl.com/index.html** goes in the form

GET / index.html HTTP/1.0

This request can be broken into 3 parts:

a) the request method i.e. GET,
b) the source name i.e. /index.html, and
c) the protocol i.e. HTTP/1.0.

Understanding HTTP Working:

GET is an HTTP method requesting the Server to send a file. /index.html is a relative path to the file being requested. HTTP/1.0 is the name and version of the Protocol implemented by the Client.

In addition to the GET method or command, the Browser may send other information about itself to the Server.

Understanding HTTP Working:

3. The Server responds to the request with the Status Code, Various Header fields, and if possible, the contents of the requested file.

Protocol	:	HTTP/1.0 200 OK
Server	:	Netscape-Enterprise/2.01
Content-Type	:	text/html
Content-Length:	87	

HTML Code of the File…

Understanding HTTP Working:

4. The TCP/IP connection is closed either by Server or Client or both.

✓ **Connection Oriented and Connectionless Protocol:**

HTTP is a Connectionless Protocol. Due to HTTP's Connectionless nature very few Server resources are required to service large number of Clients.

At the same time the Connectionless nature also have a drawback. For those sites that receive requests in millions have strong performance overhead.

✓ **Stateless Protocol:**

HTTP is a stateless protocol. By definition a protocol is said to be stateless if it has no memory of prior connections and cannot distinguish one client's request from that of another.

Features : HTTP

- ✓ In contrast, in a stateful protocols like FTP, the connection is not opened and closed with every request. For example, after the login, the stateful protocols maintains the user's credentials throughout the session. Due to its stateless nature, there is no inherent method in HTTP for tracking a client's traversal of a site.

Features : HTTP

- ✓ The stateless nature of HTTP is both a strength and a weakness. Strength is that it keeps the protocol very simple and straightforward. It can handle multiple clients because it doesn't have to remembered them all and thus require less resource. Its weakness is that some special mechanism is required to track the user's traversal.

Status Codes: HTTP

Code Range	Category	Description
1xx	Informational	Only from HTTP 1.1
2xx	Successful	Request is successfully received, understood, and accepted.
3xx	Redirection	Server requesting the Web Client to redirect to another URL.
4xx	Client Error	Request is improperly formatted or cannot be fulfilled.
5xx	Server Error	A valid request was received but the server cannot fulfill it.

GET Method:

- ✓ It is the most common method and used to request he resource from the server. It contains no body content, a GET request is composed of only a method statement and various request header fields.

GET / login.html?username=autin&password=servlets HTTP/1.0

User–Agent: Mozilla/4.02 [en] (Win95; I)

Accept: image/gif, image/jpeg; image/pjpeg, */*

GET Method:

- ✓ A drawback in using a GET method is that the (like in login transactions) information entered by the user is appended to the URL and displayed in plain text by the browser.
- ✓ Another disadvantage is that a limited amount of data can be passed as part of the URL in a GET request.
- ✓ HTTP Servers store GET parameters in system environment variable that can be accessed by CGI programs and other out-of-process applications. The number of system variables differ from OS to OS.

POST Method:HTTP

- ✓ POST is an HTTP method commonly used in passing user input to the server. The POST method differs from GET method in that all parameter information is stored in the body of the request rather than in the URL portion of the method statement.

POST Method:HTTP

- ✓ This has 2 merits first, the information submitted by the user is not visible in the URL. Second, there is no limit to the amount of information that can be passed when it is stored in the body of the request. This is because the name/value pairs passed in a POST request are accessed via the client's input stream rather than the server's environment variables (like GET parameters).

POST Method:HTTP

POST / login.html HTTP/1.0

User–Agent: Mozilla/4.02 [en] (Win95; I)

Accept: image/gif, image/jpeg; image/pjpeg, */*

Content–Length: 34

Username=auting&password=servlets

Mobile Phone Security

Dr. Digvijaysinh Rathod
Associate Professor
School of Cyber Security and Digital Forensics
National Forensic Sciences University

digvijay.rathod@nfsu.ac.in