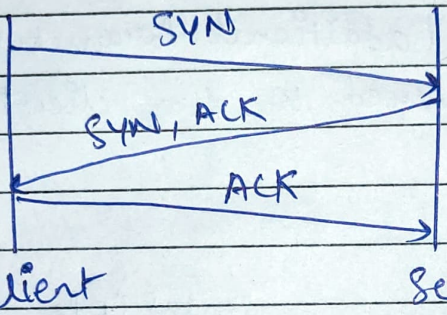
Date ..../..../.....

Web Application Security

* TCP

→ Transmission Control Protocol.
→ lies b/w the application and transport Layer.
→ Connection - Oriented.
→ TCP Conn. Establishment (3way handshake).



* HTTP

→ HyperText Transfer Protocol.
→ Access the world wide web.
→ Message Based Model
  ↳ Client → Request Message
  ↳ Server → Response Message.
→ Uses TCP as its transport mechanism.

Date ..../..../.....

* HTTP Request

Line 1: verb (HTTP Method) url (Request URL) version (either 1 or 1.1)

Line 2: Referer (URL from which Request originated)

Line 3: user Agent (Browser being used)

Line 4: Host (Specifies the hostname that is accessed)

Line 5: Cookies (additional parameters that server issued to the client).

* HTTP Response

Line 1: version (1/1.1) Status Code (Result of Request) Reason Phrase (further description)

Line 2: Server-Name (web server software)

Line 3: set-cookies (Resubmitted when request is generated again)

Line 4: Pragma (instruct browser to not store cache)

Line 5: content type

Line 6: content length

Date ..../..../.....

* HTTP Methods

1. GET :- Most Basic. Retrieve Resource.
2. POST :- Perform Actions for the screen to display.
3. HEAD :- like GET But NO Response
4. TRACE :- Diagnostic Purpose. Same content is send in response.
5. OPTIONS :- Report Methods available for a resource.
6. PUT :- Upload resource to the server.

* URI :- Uniform Resource Identifier.

* URL :- Uniform Resource Locator.

URL format

protocol :// hostname [:port] / [path/] file [?param=value]

* REST

→ Representational State Transfer.
→ Style of Architecture for the URL
Eg:- http://wahh-app.com/search?make=ford&model=pinto
REST-style
http://wahh-app.com/search/ford/pinto.

Date ..../..../.....

* HTTP Header

→ General Header

1. Connection
2. Content-Encoding
3. Content-length
4. Content-type
5. Tranfer-Encoding

→ Request Header

1. Accept
2. Accept-Encoding
3. Authentication / Authorization.
4. Cookies
5. Host

→ Response Header

1. Access Control Allow Origin
2. Cache control
3. Expiry
4. Location

* HTTPS

→ Tunneled over SSL
→ Session key is generated

Date ..../..../.....

* Encoding

→ URL Encoding

→ Encode any character within extended ASCII to transport them safely over HTTP.

Eg:- "%3d" → "="
"%25" → "%"
"%20" → Space.

→ Unicode Encoding

→ 16-Bit unicode-encoded form of a character transmitted over HTTP.

Eg:- "%u2215" → "/"
"%u00e9" → "é"

→ 8-Bit unicode encoding also available.

Eg:- "%c2%a9" → "©"

→ HTML Encoding

→ Characters encoded to incorporate with the HTML document.

Eg:- """ → """
"&" → "&"
"<" → "<"
">" → ">"

Date ...16...4...

* Cookies = Enable server to send item of data to client, which client store and resubmit to the server

① Expire :- Set a date until which the cookie is valid.

② Domain :- Specifies the domain for which the cookie is valid.

③ Path :- Specifies the URL for which cookie is valid.

④ Secure :- Submitted only in HTTPS requests.

⑤ HTTPOnly :- Cannot be directly accessed via clientside JS.

* Status Code

① 1XX :- Informational
Eg :- 100 Continue

② 2XX :- The Request is successful
Eg :- 200 OK
201 Created.

③ 3XX :- The client is redirected to a different source.
Eg :- 301 Moved Permanently.
302 Found.

④ 4XX :- Request contain some error (Client Side Errors)
Eg :- 400 Bad Request
404 Not Found.

⑤ 5XX :- Server side errors.
Eg :- 500 Internal Server Error.
503 service unavailable.

Spiral

Date ..../..../.....

* Origin : used in cross-domain Ajax requests to indicate the domain from which the request is generated.

* Web server fingerprint

→ Banner Grabbing
Collecting disclosed server infos. like the server software or about the components that are installed.

* Subdomain Enumeration (SDE).

→ Process of discovering the subdomains associated with a given domain Name.
→ Methods of SDE :-
1. Google Dorking
2. Brute Force Enumeration
3. Reverse DNS looking.
4. Public Databases : VirusTotal.

* Virtual Hosting

→ Hosting Multiple websites on a single physical server.
→ Name-Based Virtual Hosting
→ IP-Based Virtual Hosting
→ Port-Based Virtual Hosting
Eg:- 10.20.30.40/xyz

Date ..../..../.....

* DNS Poisoning

→ System DNS cache is alter.
→ DNS server is poisoned.
  ↳ Spoofing the server
  ↳ compromising the server.

* Reconaissance (Gathering Info about target)

→ Active
  ↳ Visiting website
  ↳ Port Scanning
    [can tell about the OS]
  ↳ Vulnerability Scanning
  ↳ Packet Sniffing

→ Passive
  ↳ Job Ads and tenders
  ↳ Publically available info
  ↳ Social Engineering.

* Lifecycle of PT / Hacking

① Reconaissance / Enumeration
↓
② Identification of Device or Network
↓
③ Identification of Vuln.
↓
④ exploit the vuln. → ⑤ Erase the traces
⑥ manipulate the logs.