Cleartext Protocol Analysis: HTTP

Hypertext Transfer Protocol (HTTP)

- It is the standard type of network activity to request/serve web pages, and by default, it is not blocked by any network perimeters.

HTTP/1.1 and HTTP/2 usually run over TCP
HTTP/3 runs over QUIC (UDP-based, faster handshake)

Browser sends an HTTP request.
- Method (what we want to do)
- Path
- Headers (metadata)
- Body (POST/PUT)

GET → fetch data
POST → send data
PUT/PATCH → update
DELETE → remove
HEAD → like GET but headers only
OPTIONS → "what is allowed"

Wireshark

Global search → http
↳ specific → http2
http3

HTTP request methods

GET → http.request.method == "GET"
POST → http.request.method == "POST"
-all request → http.request

HTTP Response Status Code

200 OK → Successful
301 Moved Permanently → moved to new URL
302 Moved Temporarily → temporarily moved to new URL
400 Bad Request → didn't understand the request
401 Unauthorized → URL need authorization
403 Forbidden → No access to the requested
404 Not Found → can't find the URL
405 Method Not Allowed → method is not suitable or blocked
408 Request Timeout → timeout
500 Internal Server Error → request not completed, unexpected
503 Service Unavailable → server or service is down

http.response.code == xxx

HTTP parametes

- User agent : browser or OS identification to a web
- Request URI : Points the requested resource from the server
- Full * URI : Complete URI information

URI = identifier (string that identifies a resource)

```
http.user_agent contains "nmap"
http.request_uri contains "admin"
http.request.full_uri contains "admin"
```

- Server = server name
- Host = hostname
- Connection = Status
- Line-based text data : Clearted data by the server
- HTML Form URL Encoded = web form information

```
http.server contains "apache"
http.host contains "...."
http.host == "...."
http.connection == "Keep-Alive"
data-text-lines contains "keyword"
```

User Agent field is one of the great resources for spotting anomalies in HTTP traffic.

user agent Wireshark

Global search http.user_agent

~~Resource~~

Research outcomes

- ~~Non~~-Different user agent info from the same host in a short time
- Non-standart and custom user agent info
- Subtly spelling & differences
- Audit tools
- Payload data in the user agent field

Questions

1) Investigate the user-agent. what is the number of anomalous "user-agent" type?

6

Statistics → HTTP → Request

2) what is the packet numbes with a subtle spelling difference in the user_agent?

Mozlila/6.0 → 52

3) Locate the "Log4j" attack starting phase. What in the packet numbes?

Log4j → attack before lauching Wireshark
→ attack starts with POST
→ known pattern "jndi:ldap" and "Exploit.class"

http.request.~~code~~ method == "POST" && (frame contains "jndi" || frame contains "Exploit) && (http.user_agent contains "$" || http.users_agent contains "==")

4) Locate the "Log4j" attack starting phase and decode the base 64 command. What is the IP address contacted by the adversary?

62[.]210[.]130[.]250