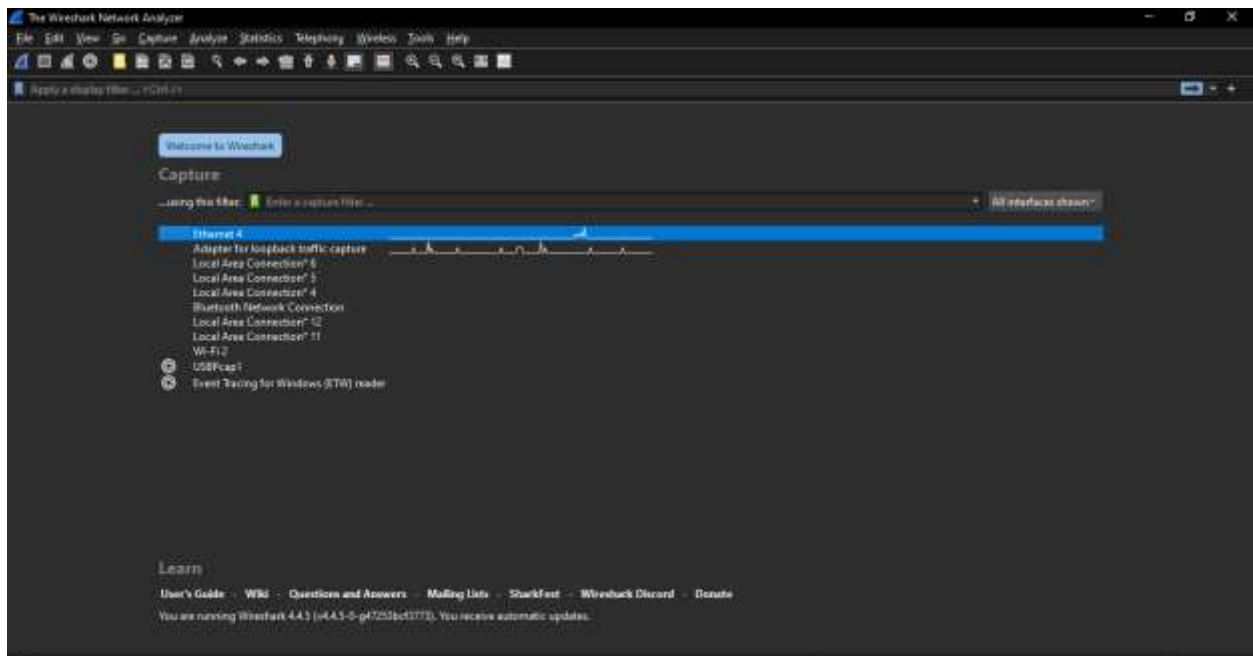
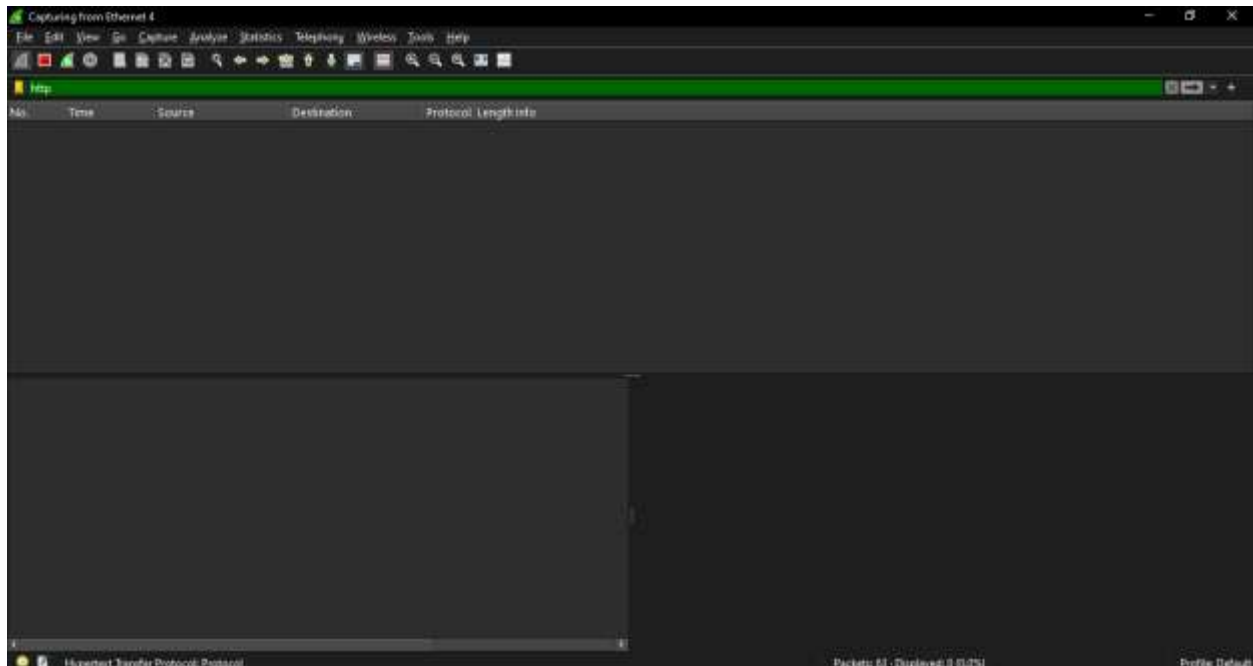


## ANALYZING HTTP HEADERS.



### 1) Install and Set Up Wireshark

- Download and install Wireshark.
- Open Wireshark and select the network interface you're using (Ethernet). Start capturing packets.

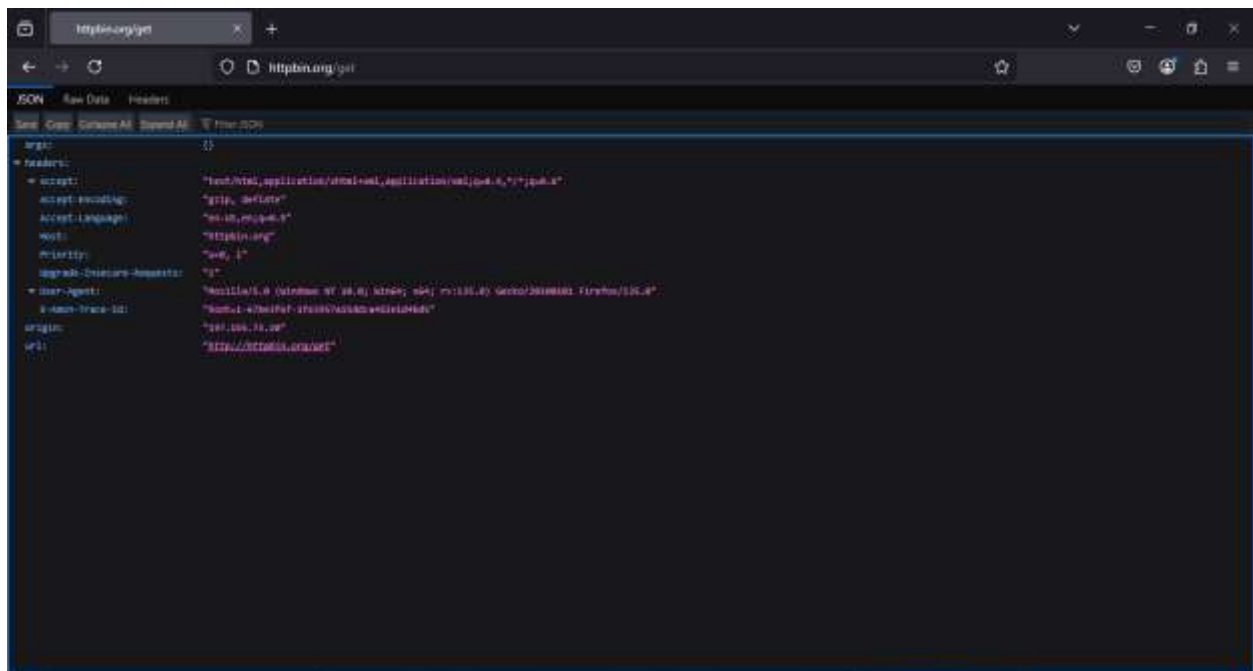


In the filter bar at the top, type http which will filter out the search and narrow it down to http headers only.

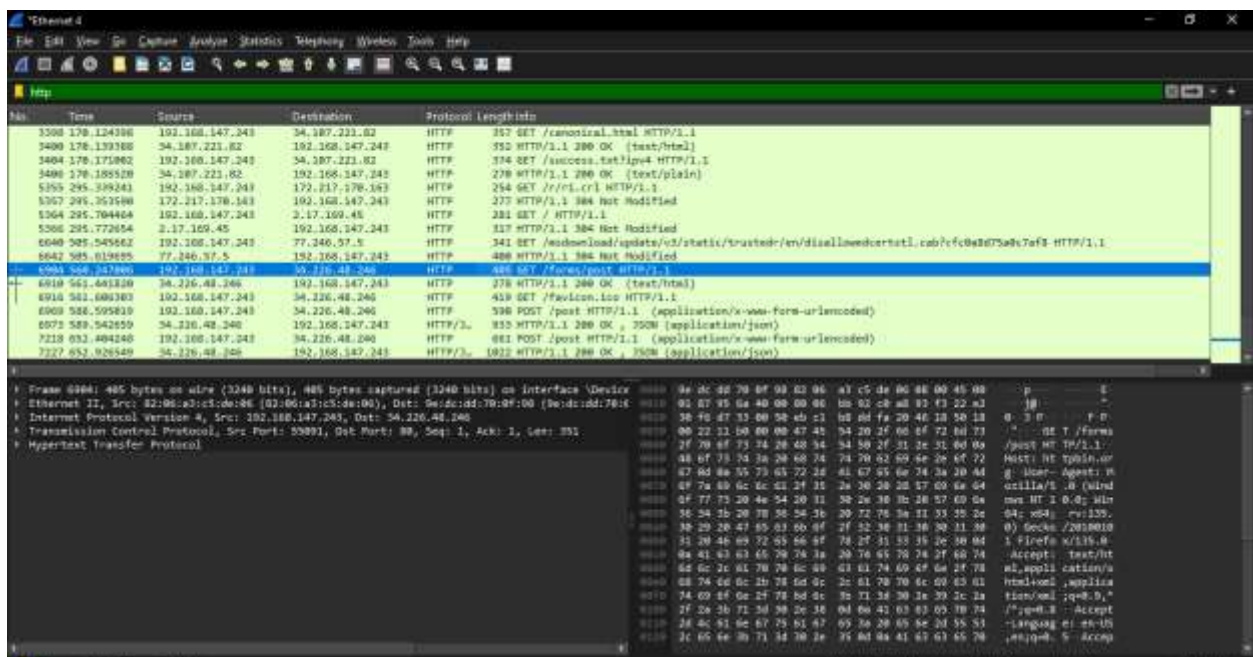
## Capture an HTTP GET Request

- Open your browser and navigate to a simple site like [httpbin.org/get](http://httpbin.org/get).

This generates an HTTP GET request to retrieve the webpage.

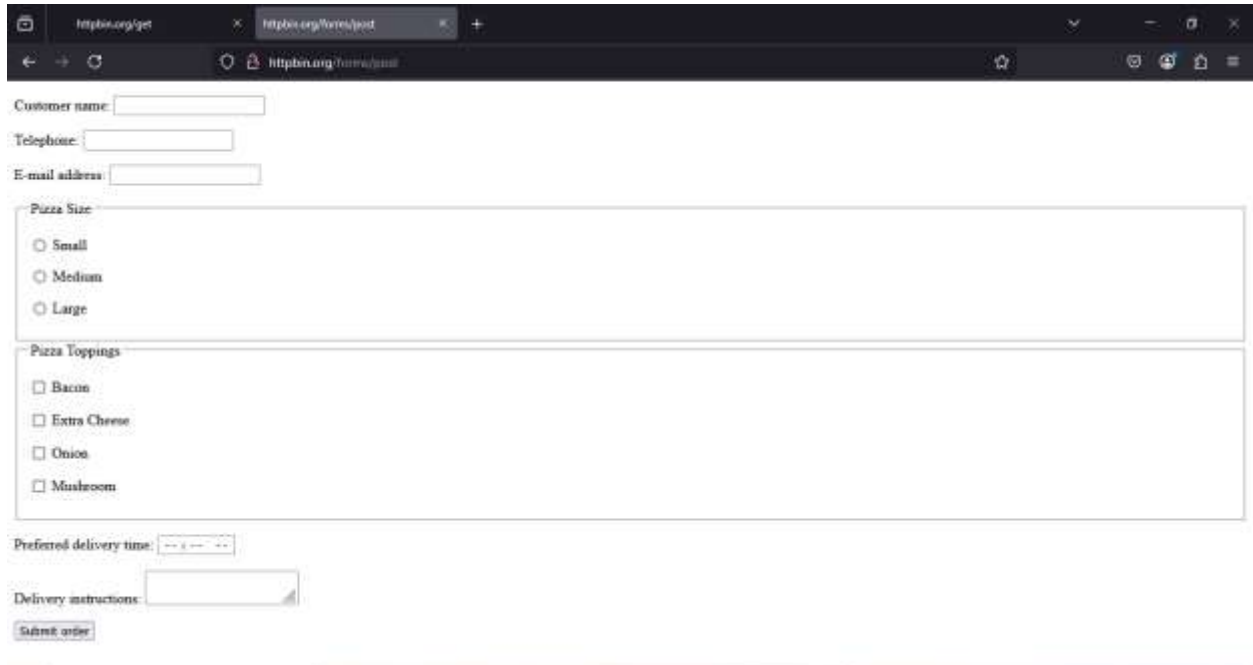


- In Wireshark, the HTTP GET request will be visible under the info column on GET / .

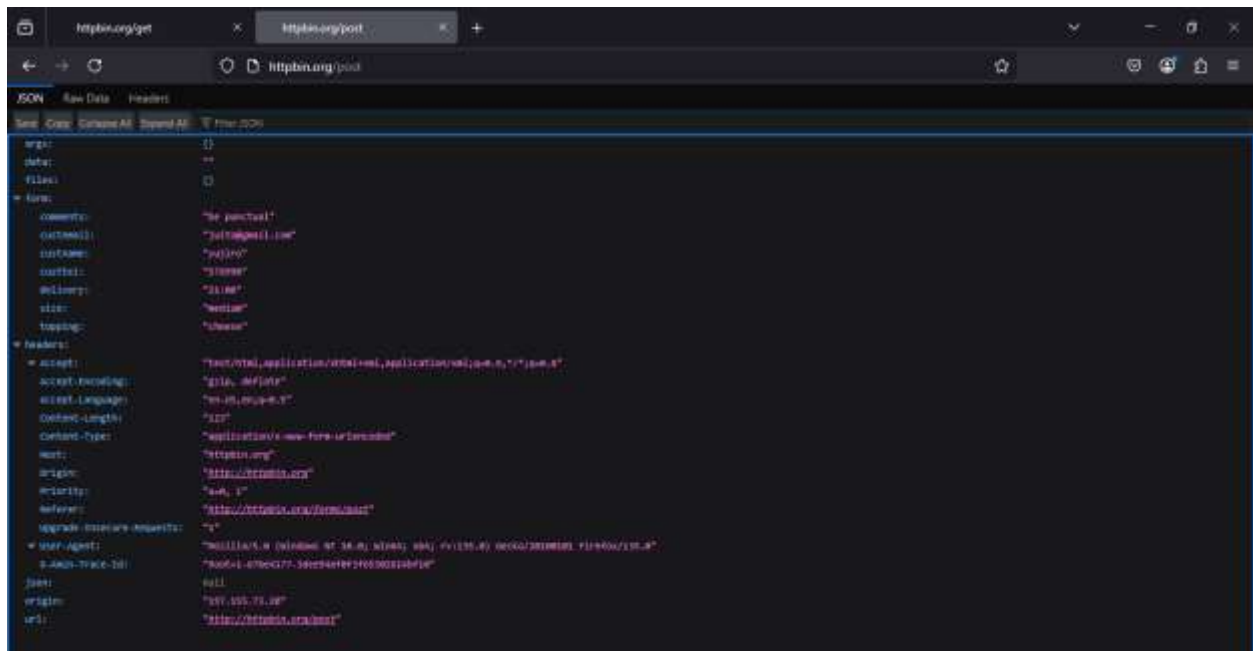


## Capture an HTTP POST Request

- To generate a POST request, visit a site with a form <httpbin.org/forms/post>.
- Submit a form after entering data.



A screenshot of a web browser showing the `httpbin.org/forms/post` page. The form contains several input fields: "Customer name:", "Telephone:", "E-mail address:", "Pizza Size:" (with radio buttons for Small, Medium, and Large), "Pizza Toppings:" (with checkboxes for Bacon, Extra Cheese, Onion, and Mushroom), "Preferred delivery time:" (a dropdown menu), and "Delivery instructions:". At the bottom is a "Submit order" button.



A screenshot of a Wireshark packet capture. The top pane shows the packet list with a selected packet of type "HTTP". The bottom pane shows the packet details for "HTTP POST". The "Raw" pane shows the raw packet data in hexadecimal and ASCII. The "Info" column shows the packet details in a structured format.

```
Raw Data: [hex dump]
Info: HTTP POST
  Host: httpbin.org
  Origin: http://httpbin.org
  Referer: http://httpbin.org/forms/post
  Accept: text/html,application/xhtml+xml,application/javascript;q=0.9,*/*;q=0.8
  Accept-Encoding: gzip, deflate
  Accept-Language: en-US,en;q=0.5
  Content-Length: 120
  Content-Type: application/x-www-form-urlencoded
  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:105.0) Gecko/20100101 Firefox/105.0
  ...
```

In Wireshark, look for a packet with POST / in the "Info" column.

**Packet 7218: 461 bytes on wire (3288 bits), 461 bytes captured (3288 bits) on interface 'Device'**

**Ethernet II, Src: 82:00:00:00:00:00 (02:00:00:00:00:00), Dst: 00:00:00:00:00:00 (00:00:00:00:00:00)**

**Internet Protocol Version 4, Src: 102.168.147.243, Dst: 34.226.48.240**

**Transmission Control Protocol, Src Port: 55897, Dst Port: 80, Seq: 1, Ack: 1, Len: 0**

**Hypertext Transfer Protocol**

**Host: httpbin.org**

**User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:115.0) Gecko/20100101 Firefox/115.0**

**Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8**

**Accept-Language: en-US,en;q=0.5**

**Accept-Encoding: gzip, deflate**

**Content-Type: application/x-www-form-urlencoded**

**Content-Length: 223**

**Origin: http://httpbin.org**

**Connection: keep-alive**

**Referer: http://httpbin.org/forms/post**

**Upgrade-Insecure-Requests: 1**

**Priority: u=0, l=v**

## A) Request and response headers.

### 1) GET REQUEST HEADER.

**Packet 6994: 485 bytes on wire (3288 bits), 485 bytes captured (3288 bits) on interface 'Device'**

**Ethernet II, Src: 82:00:00:00:00:00 (02:00:00:00:00:00), Dst: 00:00:00:00:00:00 (00:00:00:00:00:00)**

**Internet Protocol Version 4, Src: 102.168.147.243, Dst: 34.226.48.240**

**Transmission Control Protocol, Src Port: 55897, Dst Port: 80, Seq: 1, Ack: 1, Len: 0**

**Hypertext Transfer Protocol**

**Host: httpbin.org**

**User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:115.0) Gecko/20100101 Firefox/115.0**

**Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8**

**Accept-Encoding: gzip, deflate**

**Content-Type: application/x-www-form-urlencoded**

**Content-Length: 223**

**Origin: http://httpbin.org**

**Connection: keep-alive**

**Referer: http://httpbin.org/forms/post**

**Upgrade-Insecure-Requests: 1**

**Priority: u=0, l=v**

GET /forms/post HTTP/1.1

Host: httpbin.org

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:135.0) Gecko/20100101 Firefox/135.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

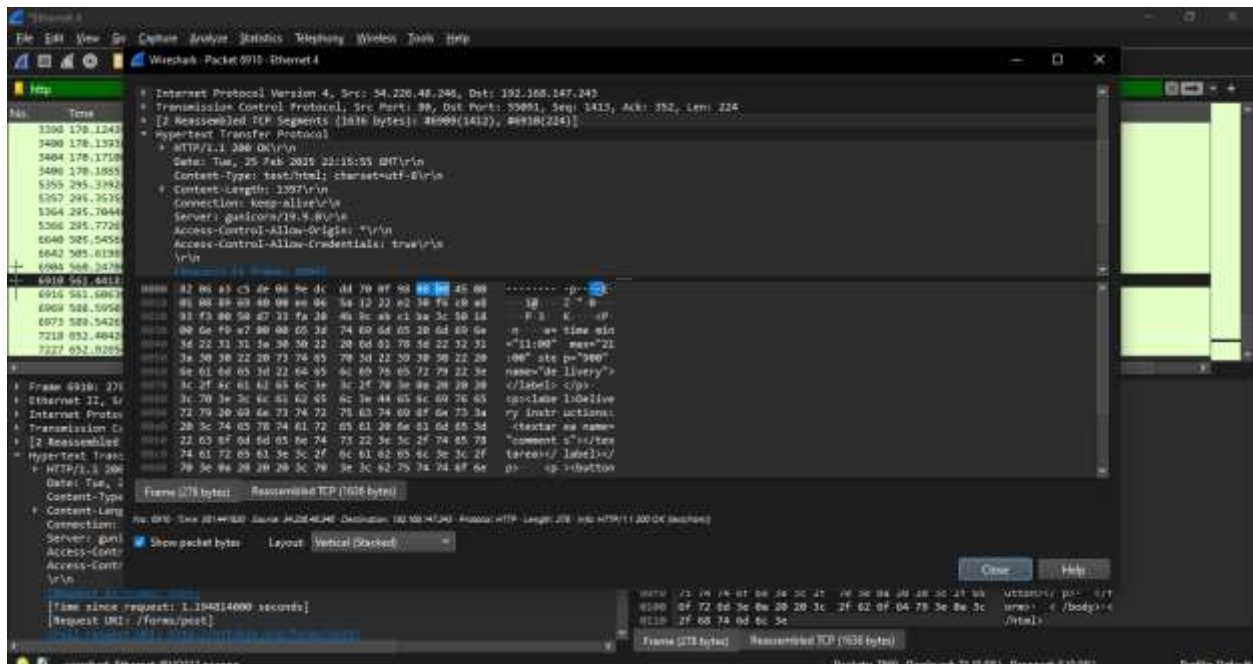
Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Connection: keep-alive

Upgrade-Insecure-Requests: 1

## 2)GET RESPONSE HEADER.



HTTP/1.1 200 OK

Date: Tue, 25 Feb 2025 22:15:55 GMT

Content-Type: text/html; charset=utf-8

Content-Length: 1397

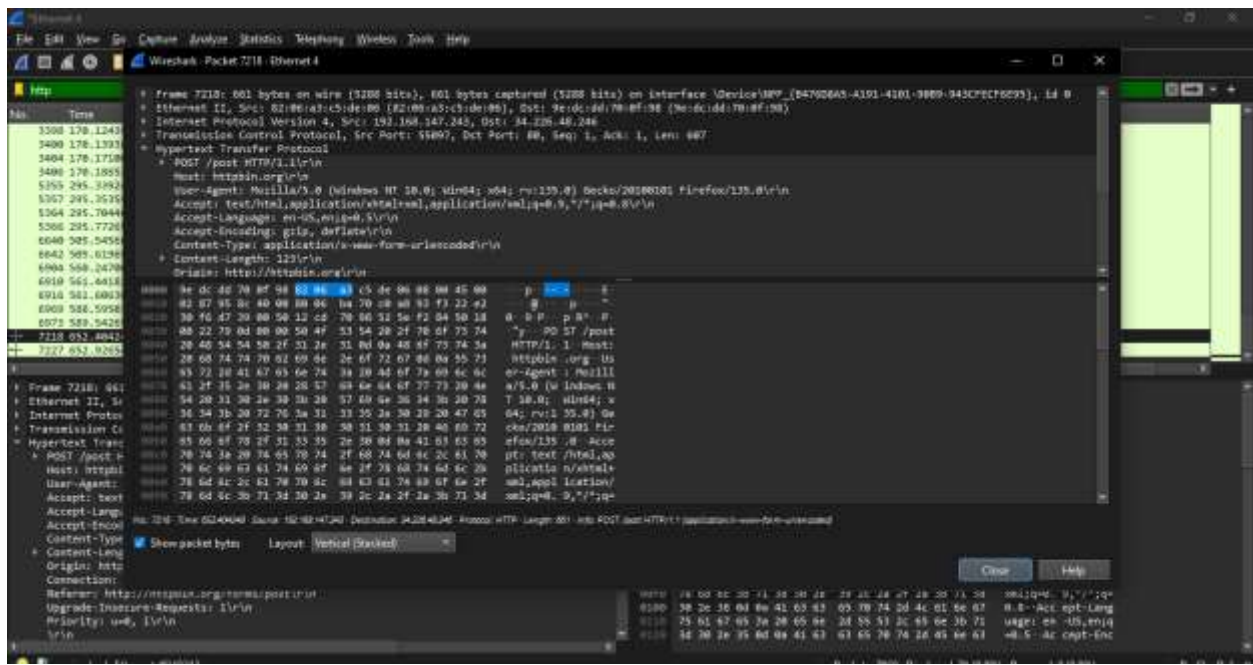
Connection: keep-alive

Server: gunicorn/19.9.0

Access-Control-Allow-Origin: \*

Access-Control-Allow-Credentials: true

### 3) POST REQUEST HEADER.



POST /post HTTP/1.1

Host: httpbin.org

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:135.0) Gecko/20100101

Firefox/135.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Content-Type: application/x-www-form-urlencoded

Content-Length: 123

Origin: http://httpbin.org

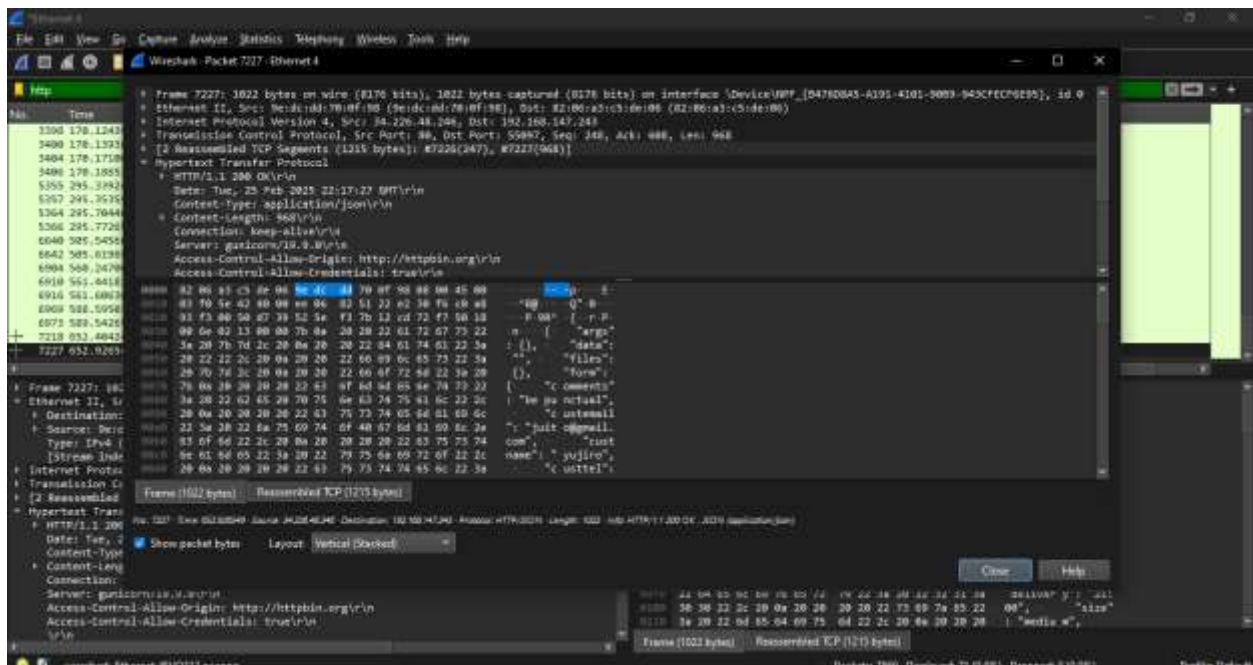


Connection: keep-alive

Referer: http://httpbin.org/forms/post

Upgrade-Insecure-Requests: 1

#### 4)POST RESPONSE HEADER.



HTTP/1.1 200 OK

Date: Tue, 25 Feb 2025 22:17:27 GMT

Content-Type: application/json

Content-Length: 968

Connection: keep-alive

Server: gunicorn/19.9.0

Access-Control-Allow-Origin: http://httpbin.org

Access-Control-Allow-Credentials: true.

❖ MIME TYPE OF THE GET RESPONSE: text/html; charset=utf-8

- ❖ HTTP STATUS CODE AND EXPLANATION OF THE GET RESPONSE: 200 OK –  
The request succeeded, and the server returned the requested resource (the HTML form page).
- ❖ MIME TYPE OF THE POST RESPONSE: application/json
- ❖ HTTP STATUS CODE AND EXPLANATION OF THE POST RESPONSE: 200 OK –  
The POST request was successful, and the server processed the submitted form data, returning a JSON response.