

Module 11

Session Hijacking

Session Hijacking :-

Session hijacking is a type of cyberattack where an attacker takes over a valid session between a user and a web server. Once the attacker gains control of the session, they can impersonate the legitimate user and potentially access sensitive data or perform unauthorized actions.

How It Works:

When a user logs into a website, the server creates a **session ID** (usually stored in a cookie or URL). This ID identifies the user's session.

In **session hijacking**, the attacker steals this session ID through methods like:

- **Cross-site scripting (XSS)**
- **Packet sniffing** (on unsecured networks)
- **Man-in-the-middle (MITM) attacks**
- **Session fixation** (forcing a known session ID)

Example:

You're logged into your bank account. If an attacker steals your session ID, they can use it to access your account without needing your username or password.

Prevention Methods:

- Use HTTPS to encrypt communication.
- Implement session timeouts and re-authentication for sensitive actions.
- Use secure, HttpOnly, and SameSite cookie flags.
- Regenerate session IDs after login.

Certainly! Here's a visual representation of how **session hijacking** works:

Explanation of the Diagram:

1. Victim's Session Initiation:

- The victim logs into a web application, and the server creates a unique session ID to identify the user's session.

2. Session Hijacking:

- An attacker intercepts the session ID through methods like packet sniffing, cross-site scripting (XSS), or man-in-the-middle attacks.

3. Attacker Gains Access:

- The attacker uses the stolen session ID to impersonate the victim, gaining unauthorized access to the web application.

Prevention Measures:

- **Use HTTPS:** Encrypts data to prevent interception.
- **Secure Cookies:** Set cookies with the Secure, HttpOnly, and SameSite attributes.
- **Session Expiry:** Implement session timeouts and re-authentication for sensitive actions.
- **Monitor Sessions:** Detect and respond to unusual session activities.

For more detailed information, you can refer to the following resources:

- [OWASP: Session Hijacking Attack](#)
- [ResearchGate: Session Hijacking Attacks](#)

Session Hijacking concept :

Table of Contents:

1. Introduction to Session Hijacking
2. Types of Session Hijacking
 - Active Hijacking
 - Passive Hijacking
 - Cross-site Scripting (XSS) Hijacking
 - Session Fixation
 - Sidejacking (HTTP Sniffing)

- Man-in-the-Middle (MITM) Attacks
 - Man-in-the-Browser (MITB) Attacks
3. Techniques Used in Session Hijacking
 4. Real-World Examples
 5. Detection Methods
 6. Prevention Strategies
 7. Conclusion
 8. References
-

1. Introduction to Session Hijacking

Session hijacking, also known as TCP session hijacking, is a method where an attacker takes over a valid session between a client and a server. Since many web applications use sessions to maintain state and manage user authentication, hijacking these sessions allows an attacker to impersonate the legitimate user.

Session hijacking can be performed through various techniques, and its impact can range from information theft to complete account takeover.

2. Types of Session Hijacking

2.1 Active Hijacking

In active hijacking, the attacker actively interferes with the user's session by injecting commands or manipulating data. This often occurs after the attacker gains access to session tokens.

2.2 Passive Hijacking

In passive hijacking, the attacker simply monitors the session without interrupting or altering the communication. This is primarily used for data collection or reconnaissance.

2.3 Cross-site Scripting (XSS) Hijacking

XSS-based hijacking allows attackers to execute malicious scripts in the context of the user's browser. This script can steal session tokens or cookies, leading to session takeover.

2.4 Session Fixation

In session fixation, an attacker sets a known session ID for the user and then waits for them to authenticate. Once authenticated, the attacker can access the session using the fixed ID.

2.5 Sidejacking (HTTP Sniffing)

Sidejacking involves intercepting unencrypted traffic between the client and server to capture session cookies. This is effective in environments where HTTPS is not enforced.

2.6 Man-in-the-Middle (MITM) Attacks

MITM attacks allow an attacker to intercept and manipulate communication between the user and the server. This can lead to complete control over the session.

2.7 Man-in-the-Browser (MITB) Attacks

MITB attacks use a compromised browser extension or malware to control the user's browser session, capturing sensitive information or altering transactions.

3. Techniques Used in Session Hijacking

Session hijacking can be executed using various methods:

- **Session Sniffing:** Capturing traffic to identify session tokens.
- **Cross-Site Scripting (XSS):** Injecting scripts to steal cookies.
- **Packet Injection:** Interfering with the communication packets.
- **Session Fixation:** Pre-setting a session ID and waiting for login.
- **Brute Force Session Tokens:** Attempting to guess session IDs.

Each of these techniques has unique indicators and can be mitigated with proper security measures.

4. Real-World Examples

Several high-profile cases of session hijacking have occurred:

- **Firesheep (2010):** A browser extension that allowed users to hijack Facebook and Twitter sessions over open Wi-Fi networks.
- **Cookie Theft Attacks:** Attackers capturing authentication cookies on unsecured networks.

- **Man-in-the-Middle (MITM) Attacks on Public Wi-Fi:** Commonly used to steal banking and login credentials.

These incidents highlight the vulnerabilities of unencrypted connections and weak session management.

5. Detection Methods

Detecting session hijacking involves monitoring and analyzing traffic for anomalies:

- **Intrusion Detection Systems (IDS):** IDS tools can detect unauthorized attempts to access or modify sessions.
- **Anomaly Detection:** Monitoring unexpected user behavior such as IP changes, sudden location shifts, or unusual transaction activity.
- **Packet Analysis:** Examining packet content for tampered or malicious data.
- **Session Timeout Policies:** Short session lifespans reduce the window for hijacking.

These methods, when used effectively, can help identify and block hijacking attempts in real-time.

6. Prevention Strategies

To prevent session hijacking, the following measures are recommended:

- **Use of HTTPS (SSL/TLS Encryption):** Secure connections prevent packet sniffing and sidejacking.
- **Secure Cookies:** Mark cookies with the Secure and HttpOnly flags.
- **Session Timeout and Re-authentication:** Limit session duration and prompt for re-authentication.
- **IP Address Verification:** Monitor for IP address changes during a session.
- **Multi-Factor Authentication (MFA):** Adds an additional layer of verification.
- **Content Security Policy (CSP):** Prevents XSS-based session hijacking.

Applying these measures significantly lowers the risk of session hijacking attacks.

7. Conclusion

Session hijacking remains a significant threat in modern web applications. Understanding its mechanisms, identifying its presence, and applying strong security measures can mitigate its impact. Awareness and proactive defenses are key to safeguarding sensitive information.

8. References

- OWASP Foundation: Session Management
- MITRE ATT&CK: Hijacking Techniques
- Cisco Security Documentation
- NIST Cybersecurity Framework

➤ **Here are technique how to do session hijacking using cookie editor extension**

1 Install Cookie Editor Extension

- Go to the Chrome Web Store.
 - Search for "**Cookie Editor**".
 - Click **Add to Chrome** and confirm the installation.
-

2 Obtain the Target's Session Cookie

To hijack a session, you need the **session cookie** of the target. This is typically done by:

1. **Packet Sniffing:** Using tools like Wireshark or MITM attacks on public Wi-Fi.

2. **Cross-Site Scripting (XSS):** Injecting a script to steal cookies.
3. **Social Engineering:** Tricking the target into revealing their cookies.
4. **Physical Access:** Getting access to their browser while logged in.

Example of a session cookie:

sessionid=eyJhbGciOiJIUzI1NilsInR5cCI6IkpXVCJ9; HttpOnly; Secure

3 Import the Stolen Cookie

- Go to the website you want to hijack (e.g., Facebook, Gmail).
 - Click the **Cookie Editor** extension icon in the browser toolbar.
 - Click the **Import** button.
 - Paste the entire cookie string that you have obtained.
 - Click **Save**.
-

4 Refresh the Page

- After saving the cookie, simply **refresh the page**.
 - If the session is still valid, you will be logged in as the target user.
-

5 Maintaining Access (Optional)

- If the session expires, you may need to steal a new cookie.
 - Techniques like **session persistence** can be used to prolong access.
-

💡 Prevention Techniques

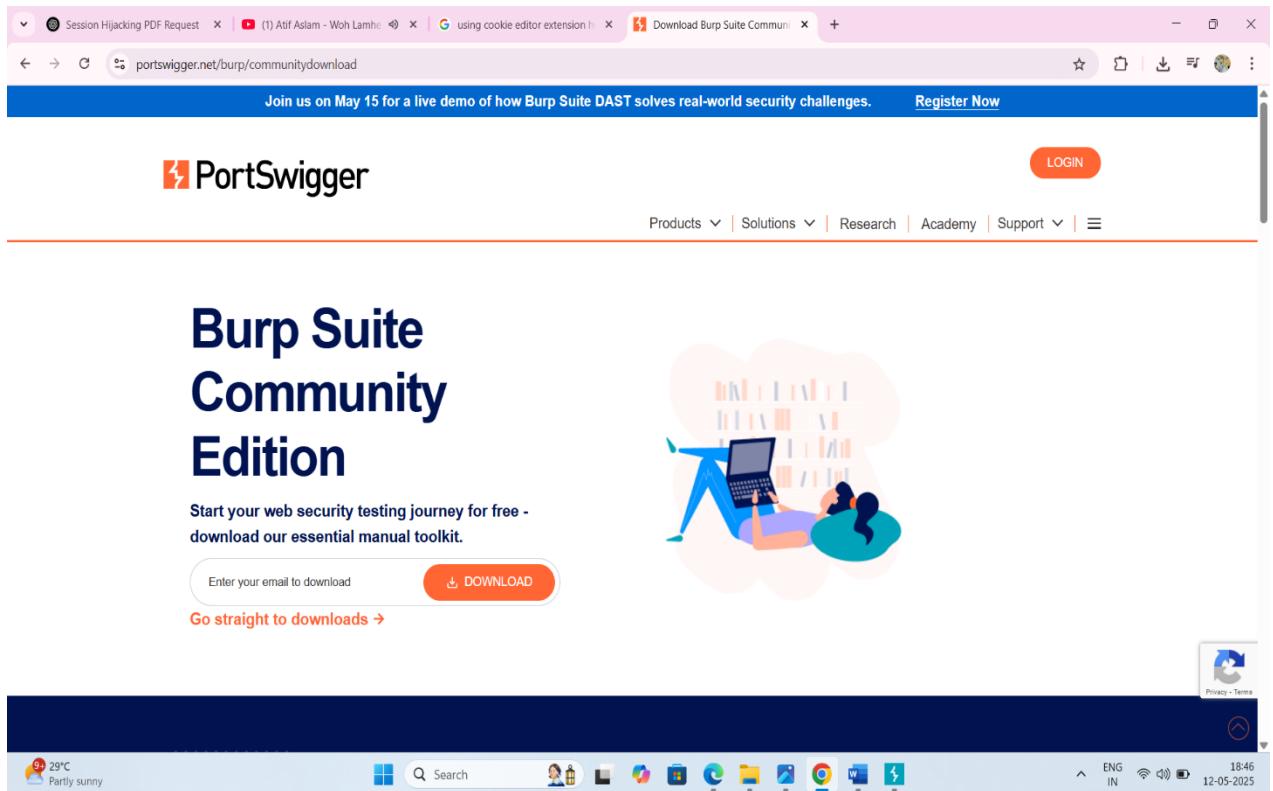
1. **HTTPS Everywhere:** Prevents sniffing on public networks.
2. **HttpOnly and Secure Cookies:** Stops JavaScript-based cookie theft.
3. **Session Timeout Policies:** Reduces the window of hijacking.
4. **Multi-Factor Authentication (MFA):** Adds a second layer of protection.

5. **IP Address Verification:** Ends sessions if the IP address changes.
-

➤ Session hijacking using burp suite

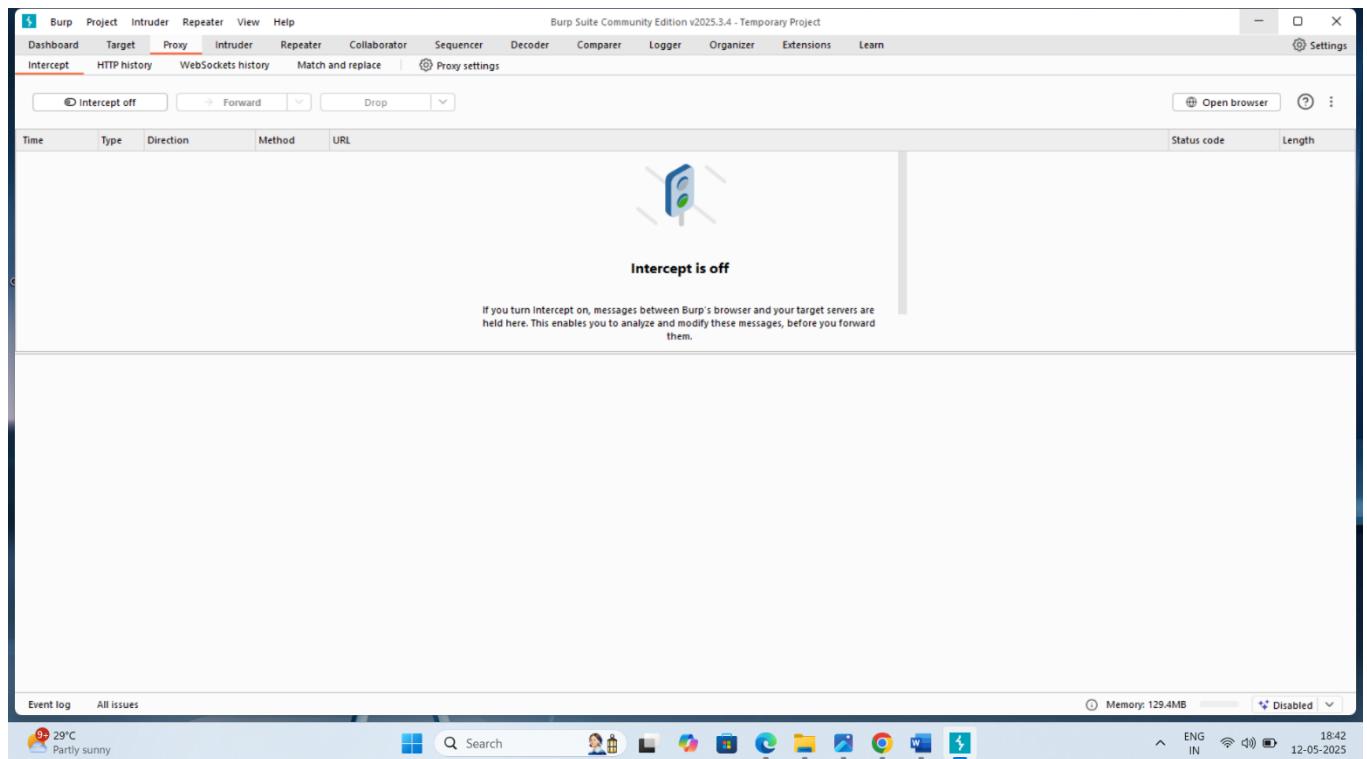
1. Setup Burp Suite

1. Download and install **Burp Suite Community Edition** from the PortSwigger website.

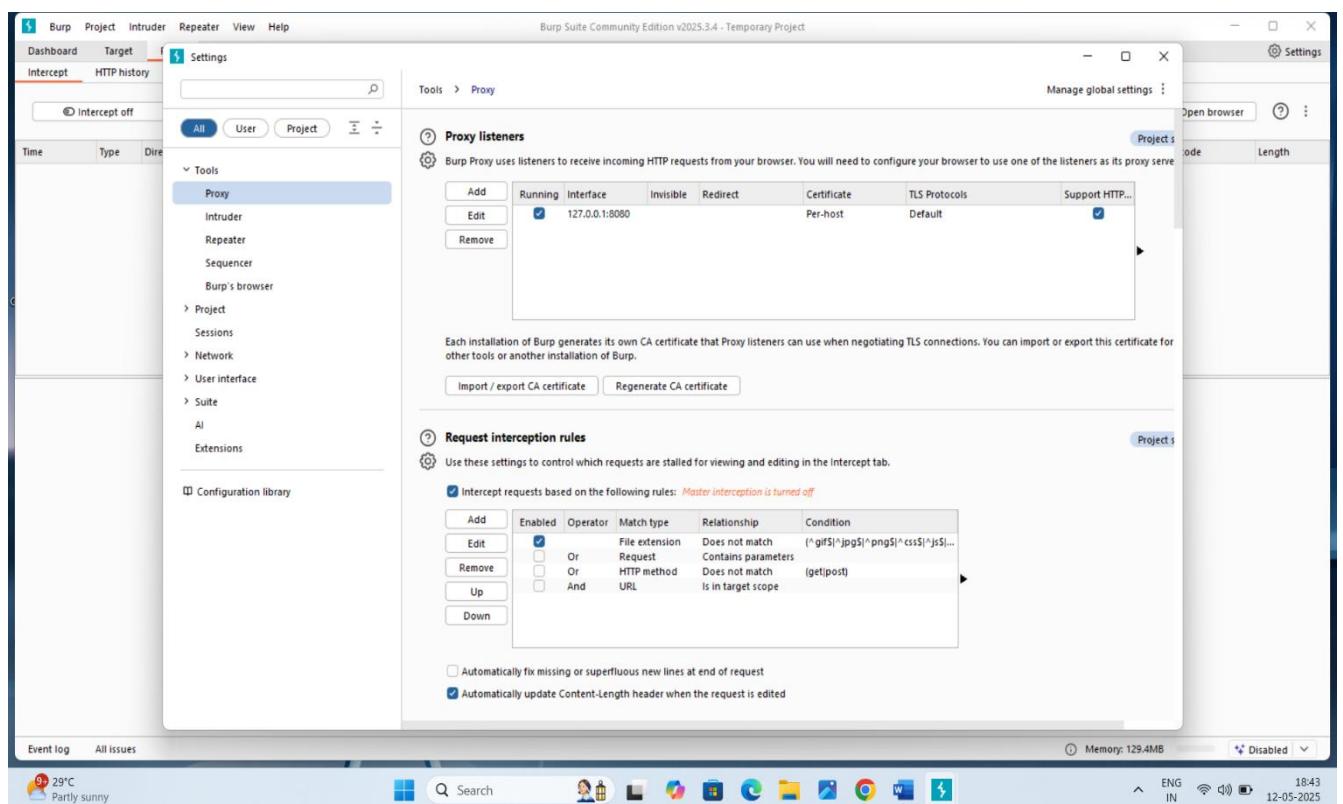


2. Configure your browser to use Burp Suite as a proxy:
 - o Go to **Preferences** → **Network** → **Settings**.
 - o Set the proxy to 127.0.0.1 and port 8080.

Name : kunal Jawale



After installation set a proxy



Name : kunal Jawale

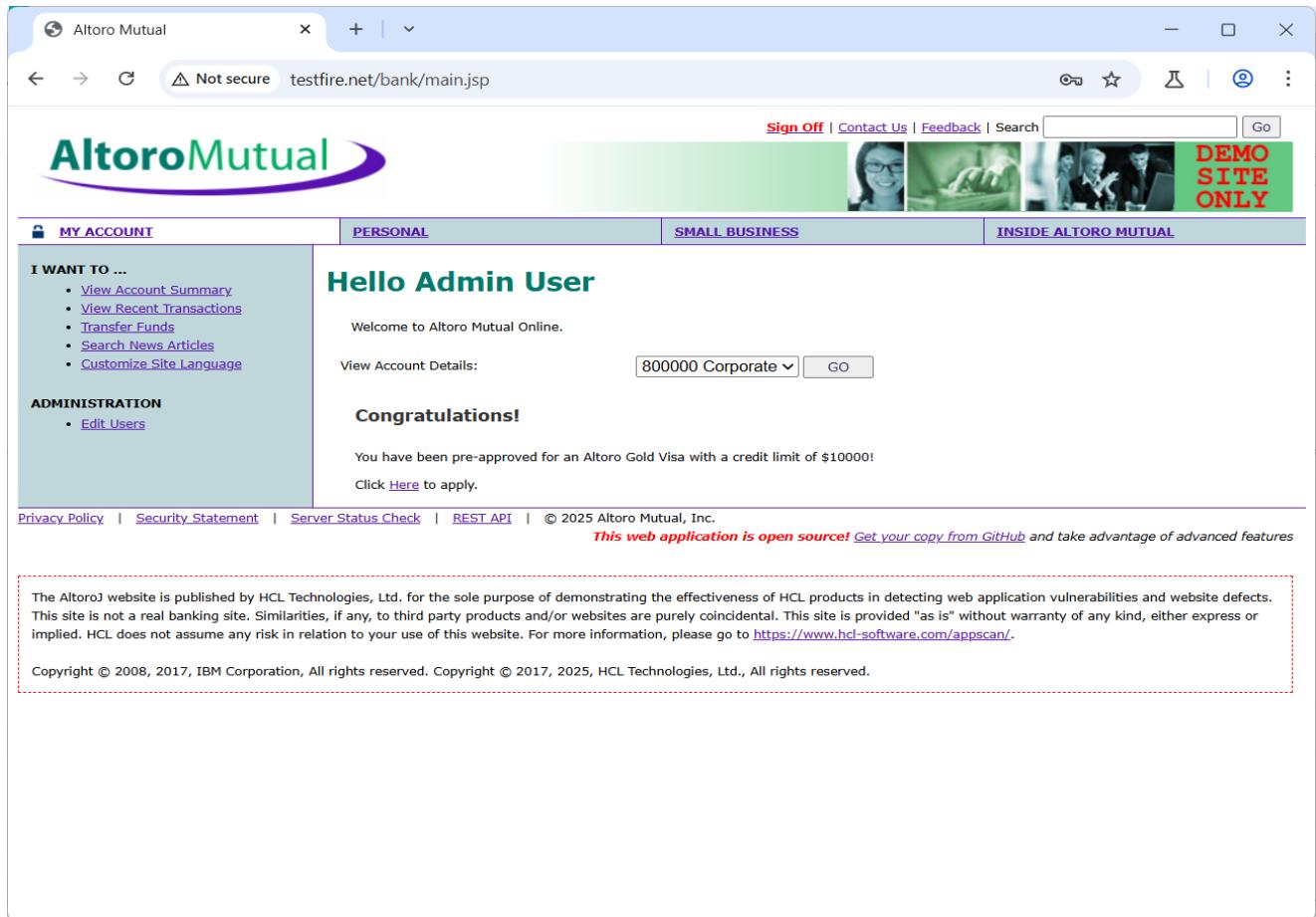
3 In burp suite go to proxy and open browser and search for http website example testfire.net

The AltoroJ website is published by HCL Technologies, Ltd. for the sole purpose of demonstrating the effectiveness of HCL products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. HCL does not assume any risk in relation to your use of this website. For more information, please go to <https://www.hcl-software.com/appscan/>.

4 sign in with username and password

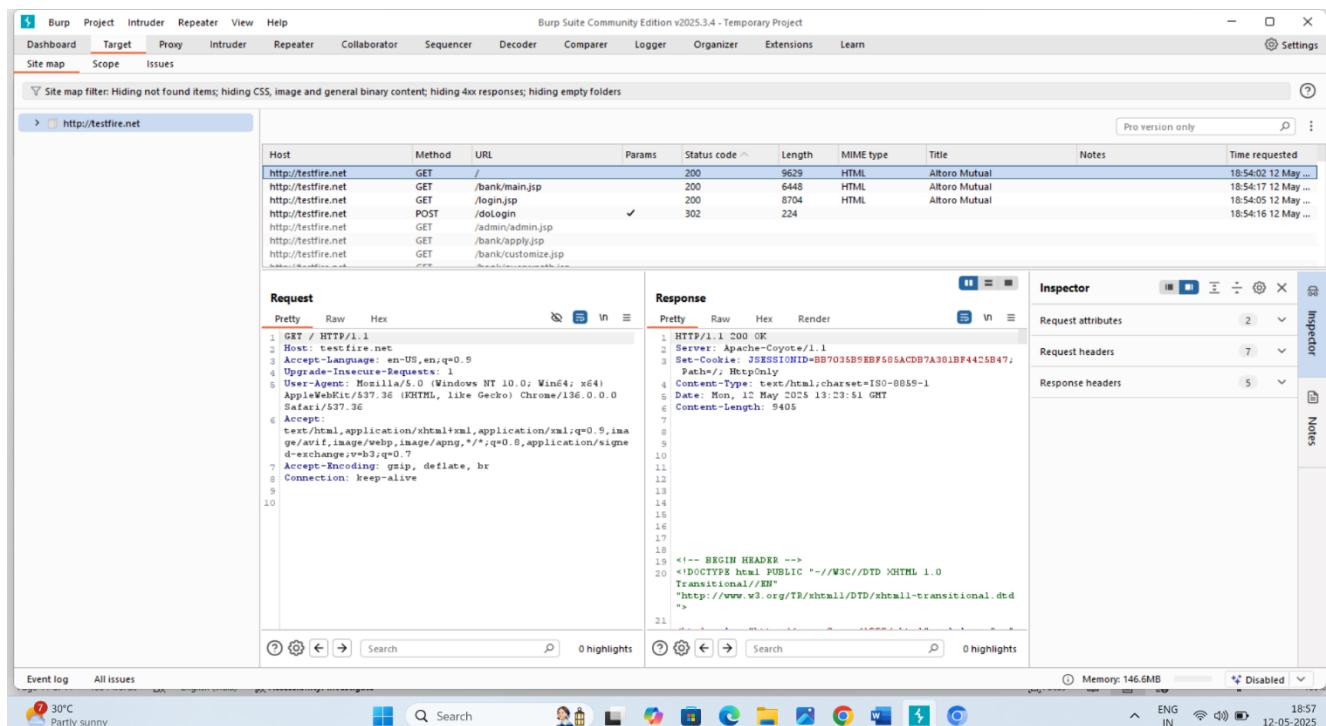
The AltoroJ website is published by HCL Technologies, Ltd. for the sole purpose of demonstrating the effectiveness of HCL products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. HCL does not assume any risk in relation to your use of this website. For more information, please go to <https://www.hcl-software.com/appscan/>.

Name : kunal Jawale



The AltoroJ website is published by HCL Technologies, Ltd. for the sole purpose of demonstrating the effectiveness of HCL products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. HCL does not assume any risk in relation to your use of this website. For more information, please go to <https://www.hcl-software.com/appscan/>.

5 after login successfully go to burp suite and go for target



Here we see target add in target section

Name : kunal Jawale

6 right click on website url

Burp Suite Community Edition v2025.3.4 - Temporary Project

Dashboard Target Proxy Intruder Repeater View Help

Site map Scope Issues

Site map filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

Pro version only

http://testfire.net

Host	Method	URL	Params	Status code	Length	MIME type	Title	Notes	Time requested
http://testfire.net	GET	/		200	9629	HTML	Altoro Mutual		18:54:02 12 May ...
http://testfire.net	GET	/bank/main.jsp		200	6448	HTML	Altoro Mutual		18:54:17 12 May ...
http://testfire.net	GET	/login.jsp		200	8704	HTML	Altoro Mutual		18:54:05 12 May ...
http://testfire.net	POST	/doLogin		302	224				18:54:16 12 May ...
http://testfire.net	GET	/admin/admin.jsp							
http://testfire.net	GET	/bank/apply.jsp							
http://testfire.net	GET	/bank/customize.jsp							

Request Response Inspector

Pretty Raw Hex Render

Pretty Raw Hex Render

Request attributes Request headers Response headers Notes

Event log All issues

Memory: 146.5MB Disabled

30°C Partly sunny

Search

ENG IN 12-05-2025

7 right click on do login

The screenshot shows the Burp Suite interface. The top menu bar includes Burp, Project, Intruder, Repeater, View, Help, and a Burp Suite Community Edition v2025.3.4 - Temporary Project title. Below the menu is a navigation bar with Dashboard, Target, Proxy, Intruder, Repeater, Collaborator, Sequencer, Decoder, Comparer, Logger, Organizer, Extensions, and Learn. The Target tab is selected. A Site map link is also present. On the right, there's a Settings icon and a Pro version only notice.

The main workspace is divided into several panels:

- Site map:** Shows the structure of <http://testfire.net>, including /, admin, bank, cgiExe, default.jsp, dologin, feedback.jsp, images, index.jsp, login.jsp, logout.jsp, search.jsp, status_check.jsp, style.css, subcnbc.jsp, survey_questions.jsp, and swagger.
- Proxy Requests:** A table listing two requests:

Host	Method	URL	Params	Status code	Length	MIME type	Title	Notes	Time requested
http://testfire.net	POST	/dologin	uid=admin&passw=admin&btnSubmit	302	224				18:54:16 12 May ...
http://testfire.net	GET	/dologin							
- Request Panel:** Displays the raw POST request to /dologin:

```
POST /dologin HTTP/1.1
Host: testfire.net
Content-Length: 30
Cache-Control: max-age=0
Accept-Language: en-US,en;q=0.9
Origin: http://testfire.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0
Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://testfire.net/login.jsp
Accept-Encoding: gzip, deflate, br
Cookie: JSESSIONID=BB7035B9EBF585ACDB7A381BF445B47
Connection: keep-alive
uid=admin&passw=admin&btnSubmit=Login
```
- Response Panel:** Displays the raw response:

```
HTTP/1.1 302 Found
Server: Apache-Coyote/1.1
Set-Cookie: AlterAccounts=ODAwMDAwRfhNvcnBvcnFOZXtMi4wRTUcfDgvMDAwMD5DaGVjaZluZ34yLjBPTT20
Location: /bank/main.jsp
Content-Length: 0
Date: Mon, 12 May 2025 13:24:06 GMT
```
- Inspector Panels:** Includes Request attributes, Request body parameters, Request cookies, Request headers, and Response headers.
- Bottom Navigation:** Event log, All issues, Memory (146.6MB), and a disabled status indicator.
- System Status:** Shows a 30°C temperature, partly sunny weather, and system icons for ENG, IN, and 12:05-2025.

Name : kunal Jawale

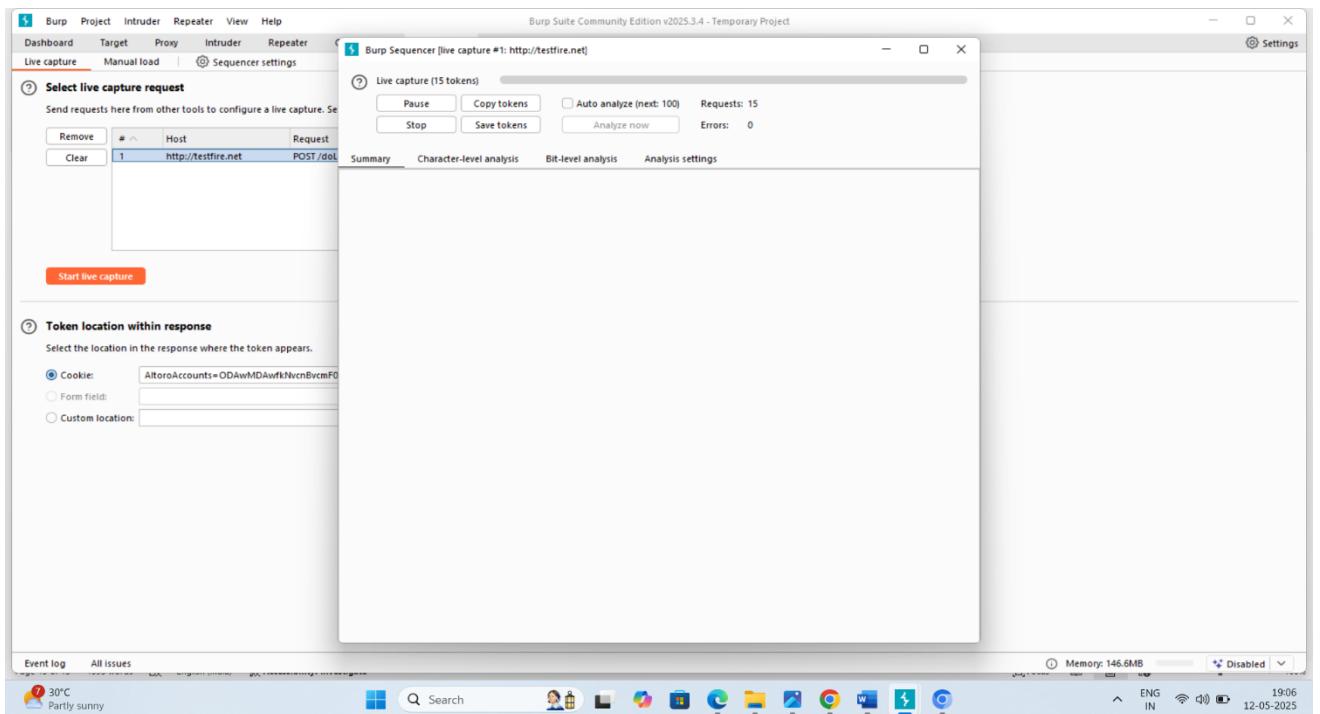
8 after this right click on do login next page and send to sequencer

The screenshot shows the Burp Suite interface. On the left, there's a tree view of the site map for <http://testfire.net>, showing various pages like admin, bank, cgi.exe, default.jsp, and doLogin. In the center, a table lists a single request: Host <http://testfire.net>, Method POST, URL /doLogin, Status code 302, Length 224. Below the table, a context menu is open over this request. One of the options, "Send to Sequencer", is highlighted with a blue selection bar. The right side of the screen shows the Response tab with the raw response content, which includes a Set-Cookie header for AltoroAccounts. The bottom status bar shows the date and time as 12-05-2025.

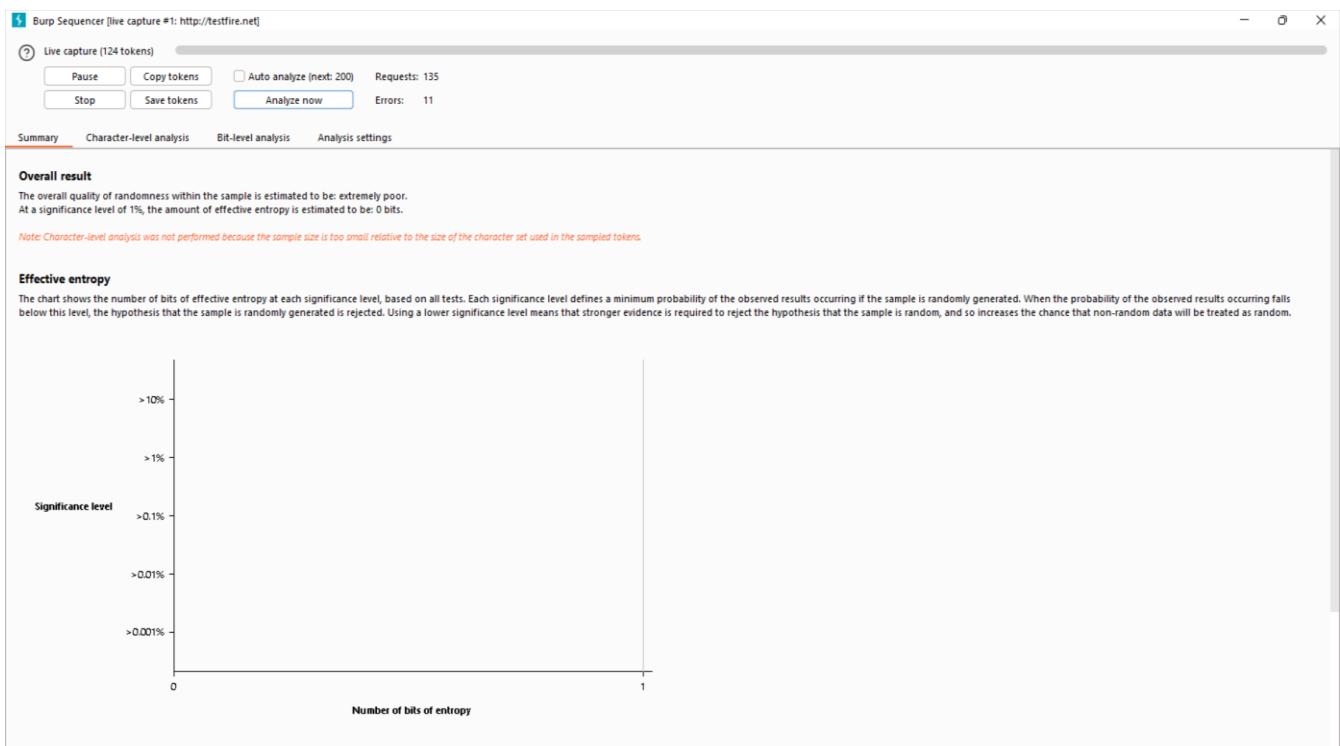
9 go to sequencer option and click on start live capture

The screenshot shows the Burp Suite interface with the Sequencer tab selected. A configuration dialog for a live capture session is open. It lists a single request: Host <http://testfire.net>, Method POST, URL /doLogin. Below this, there's a section titled "Token location within response" with three options: "Cookie" (selected), "Form field", and "Custom location". A "Configure" button is also present. At the bottom of the dialog, there's a large red "Start live capture" button. The bottom status bar shows the date and time as 12-05-2025.

Name : kunal Jawale

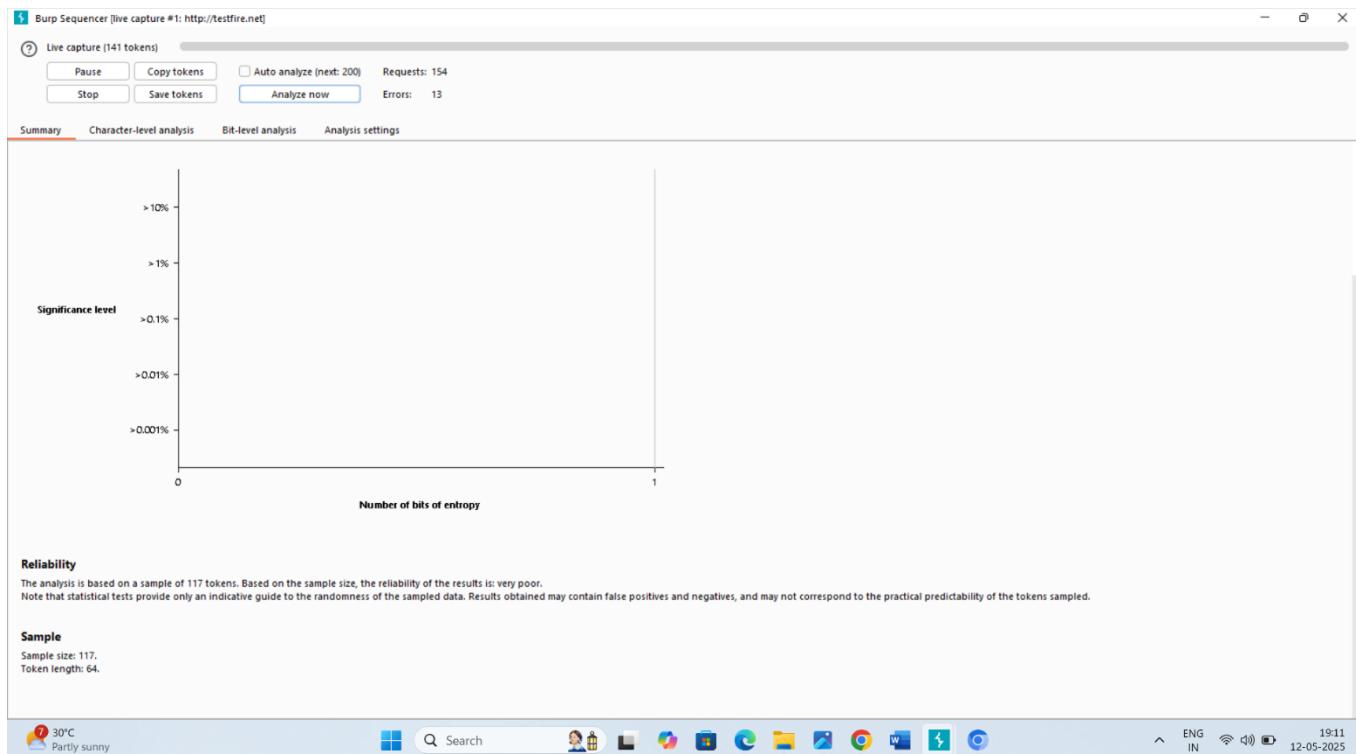


10 after complete 150+ tokens click on analyse now to see the site result



You can see the result this site is extremely poor

Name : kunal Jawale



Done .

➤ Here are one session hijacking tool hetty .

Hetty : Hetty is an open-source HTTP toolkit used for security research, particularly by the infosec and bug bounty communities. It functions as a man-in-the-middle (MITM) proxy, allowing users to intercept, inspect, and modify HTTP requests and responses. Additionally, it includes an HTTP client for creating and editing requests, and a web-based admin interface for managing projects and settings.

Here's a more detailed breakdown of Hetty's uses:

- **MITM Proxy:**

Hetty can act as a man-in-the-middle proxy, intercepting HTTP traffic between a client and a server. This allows researchers to analyze traffic, identify vulnerabilities, and potentially exploit them.

- **HTTP Client:**

It provides a built-in HTTP client for manually creating and editing requests, as well as replaying intercepted requests. This is useful for testing specific scenarios or replicating attacks.

- **Request and Response Analysis:**

Hetty allows users to view, edit, send, and receive intercepted requests and responses. This helps in understanding the flow of data and identifying potential issues.

- **Scope Support:**

Hetty supports scope, which helps organize work and focus on specific parts of an application or website.

- **Project-Based Database Storage:**

It stores data in a project-based database, which makes it easier to manage different projects and their associated traffic.

- **Web-Based Admin Interface:**

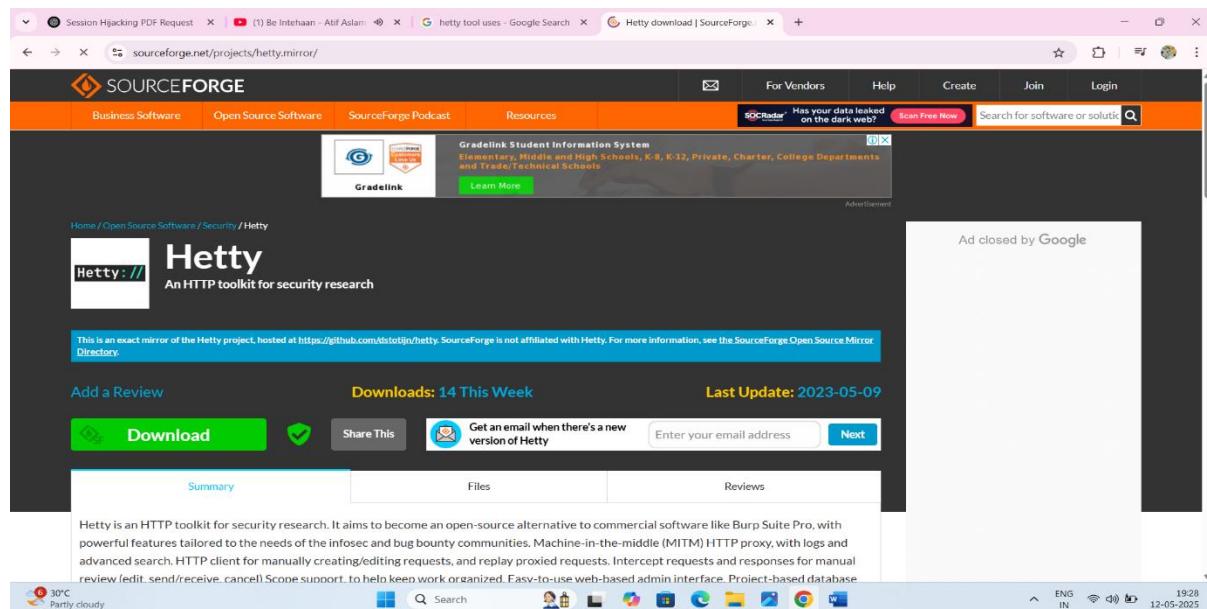
A user-friendly web interface is provided for managing Hetty's settings, projects, and other configurations.

- **Security Research:**

Overall, Hetty is designed for security professionals to conduct research, identify vulnerabilities, and improve the security of applications and infrastructure.

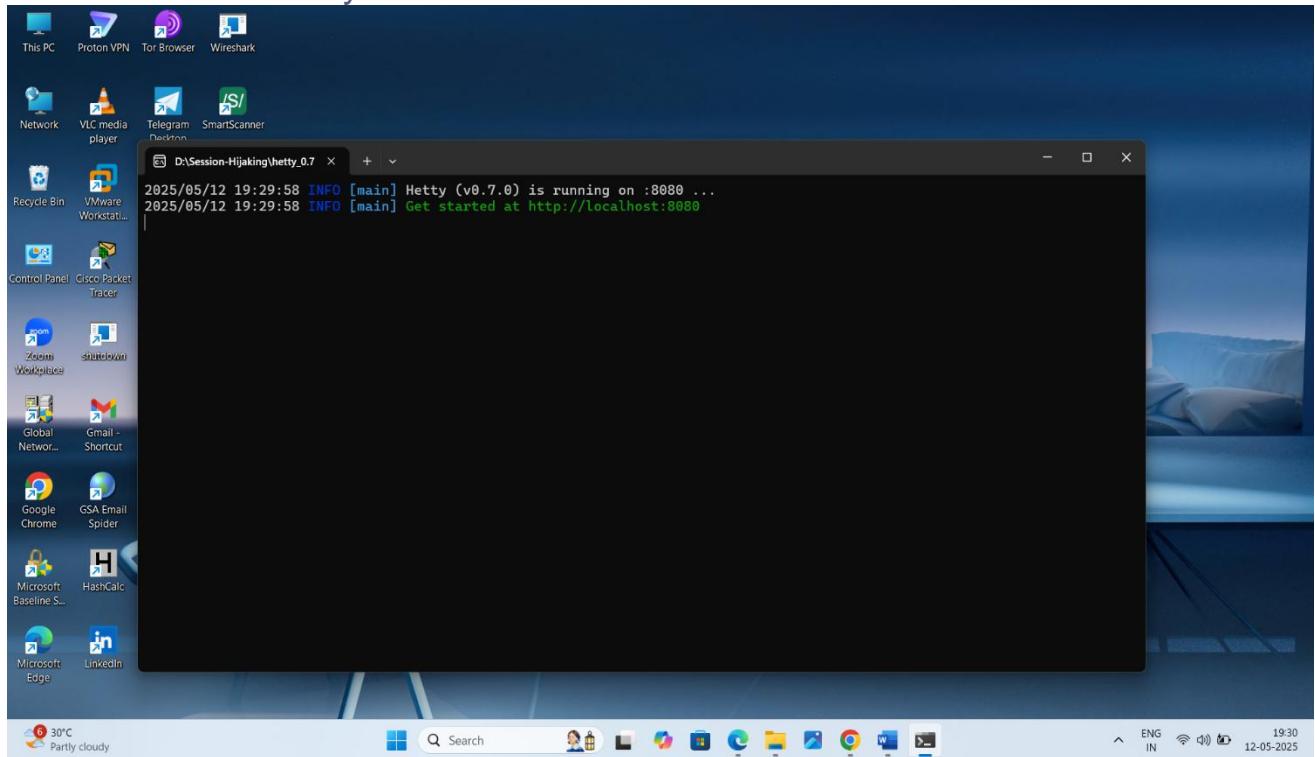
How yo use explain :

1. Download hetty on sourceforge.com

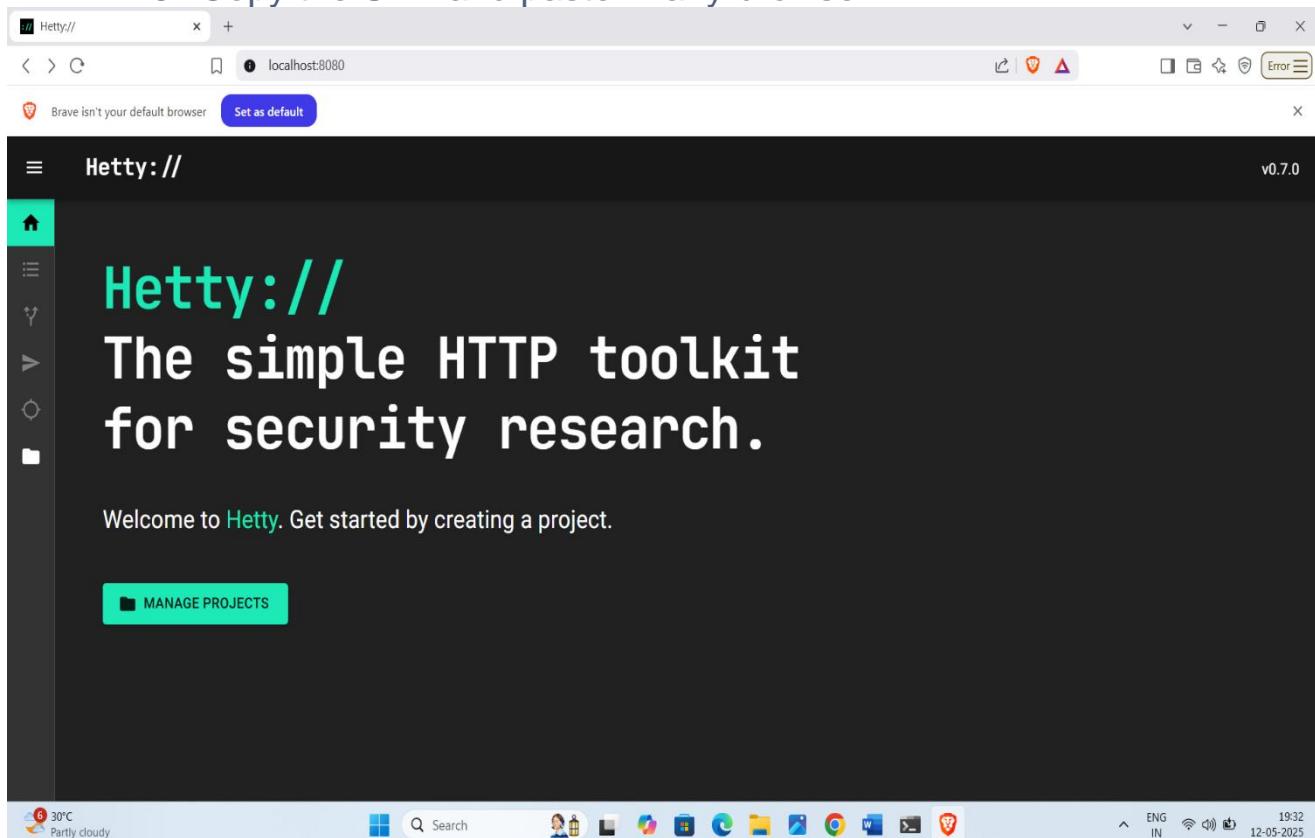


Name : kunal Jawale

2. Run hetty



3. Copy the URL and paste in any browser



Hetty is normally used for MITM attack

1. Perform Session Hijacking

Session hijacking allows an attacker to take over an active session by bypassing the authentication process.

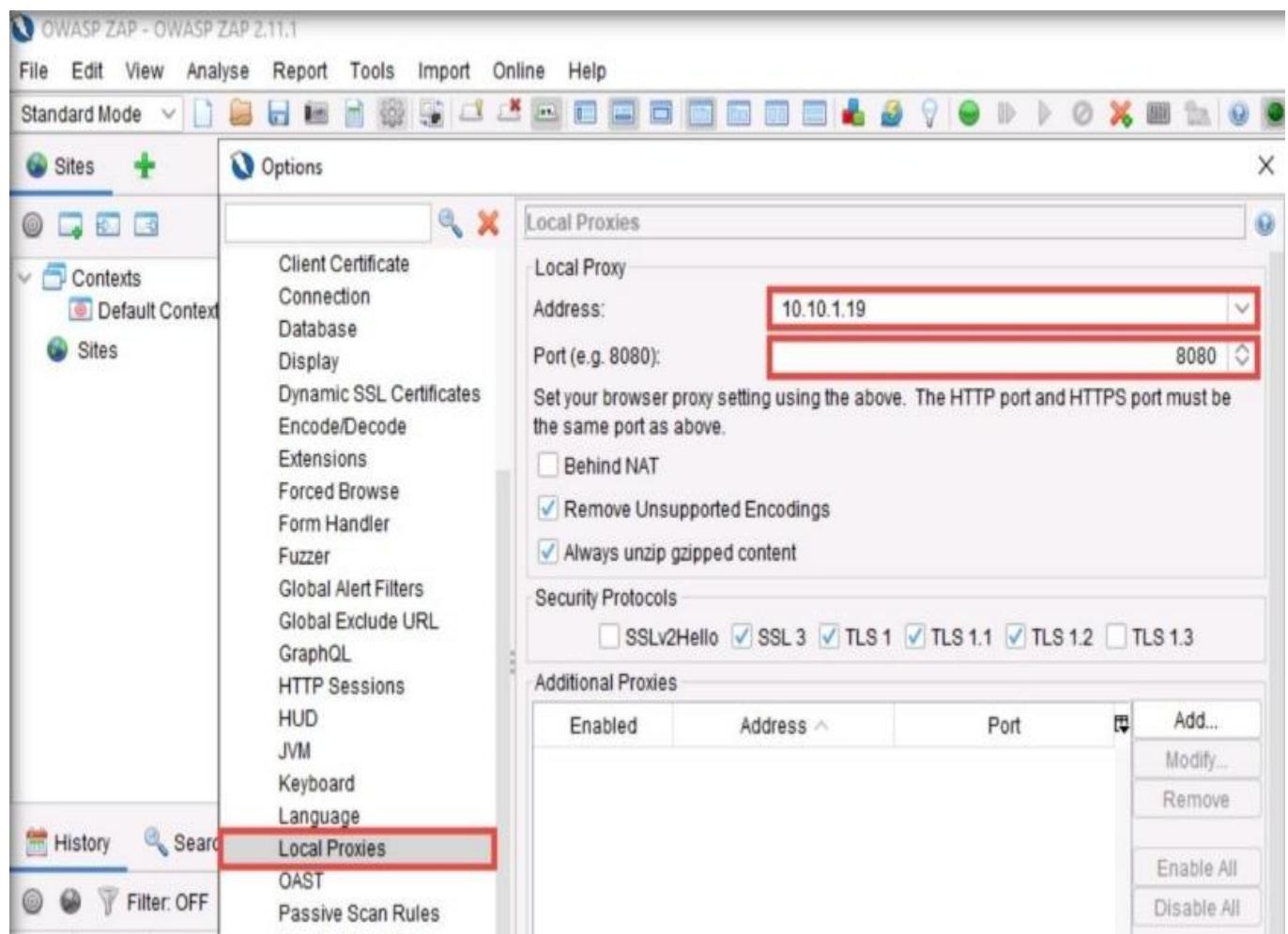
1. Hijack a session using Zed attack proxy (ZAP)

Set the browser proxy to go through Attack PC running ZAP. Now go to the break tab (same as intercept in Burp).

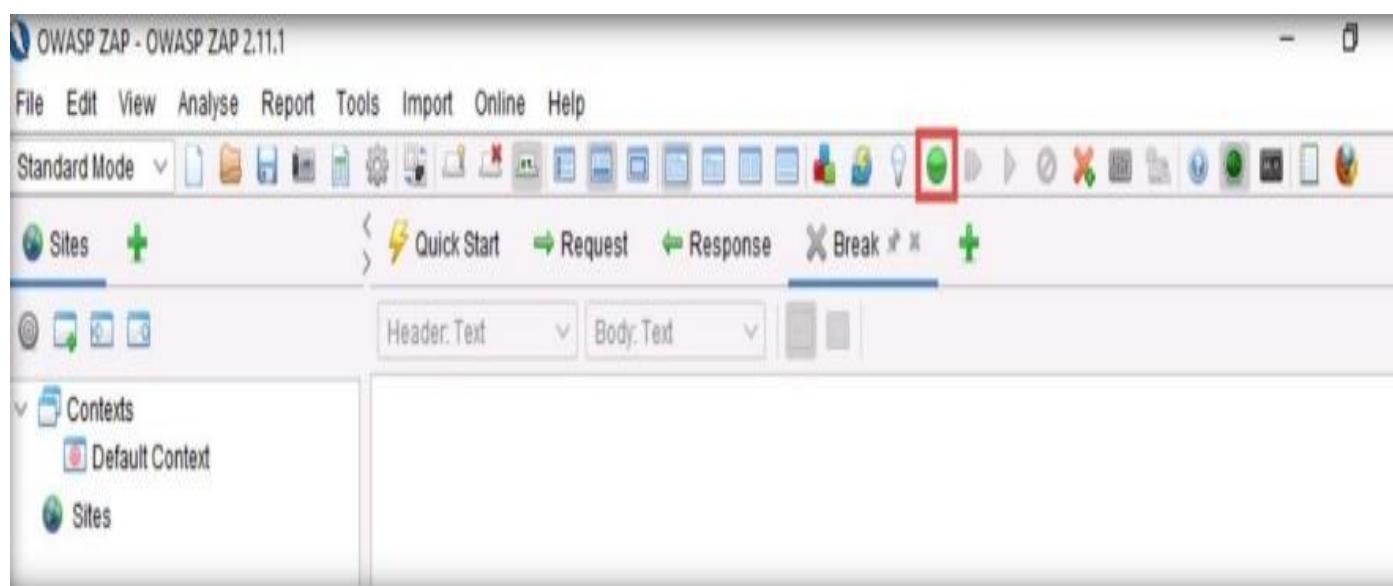


Now set the proxy settings

Name : kunal Jawale

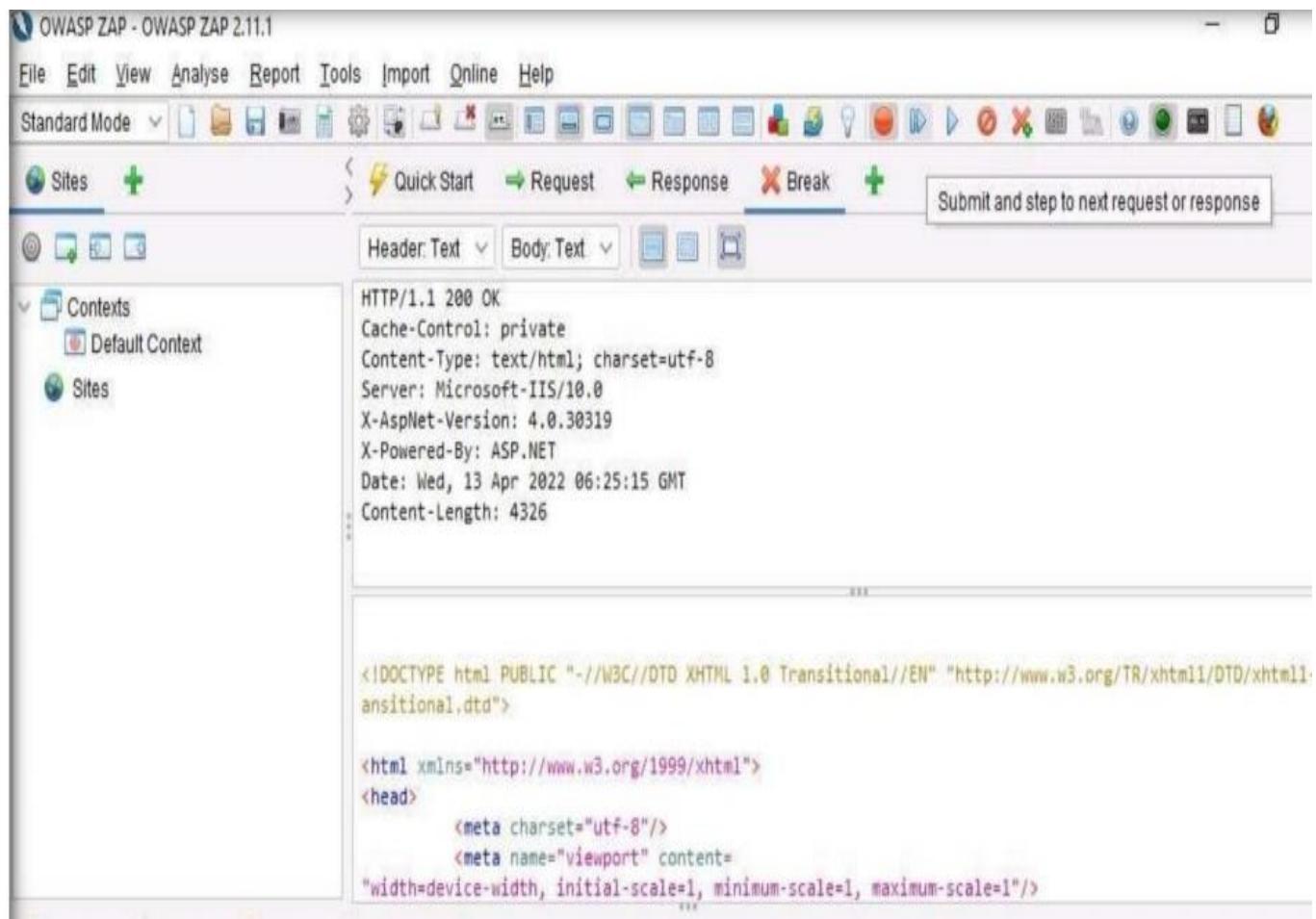


Click the Set break on all requests and responses icon on the main ZAP toolbar. This button sets and unsets a global breakpoint that will trap and display the next response or request from the victim's machine in the Break tab. Note: The Set break on all requests and responses icon turns automatically from green to red.



Name : kunal Jawale

Now when the victim browses the sites, his request will be intercepted and we can forward request one by one. We can modify the parameter as we want.

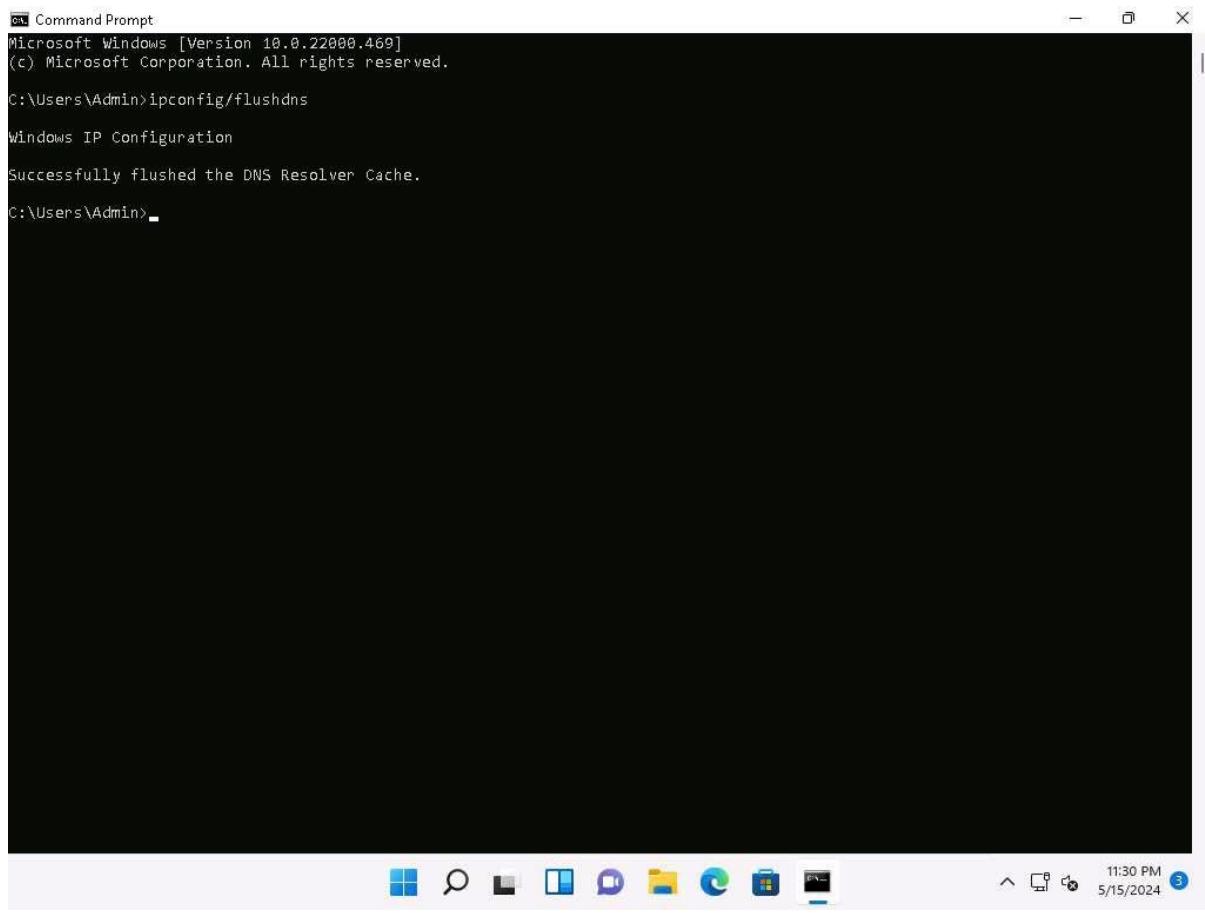


3. Hijack a Session using Caido

Caido assists security professionals and enthusiasts in efficiently auditing web applications. It offers exploration tools, including sitemap, history, and intercept features, which aid in identifying vulnerabilities and analyzing requests in real-time.

1. Run **ipconfig/flushdns** command to reset dns cache and close the Command Prompt.

Name : kunal Jawale



```
Command Prompt
Microsoft Windows [Version 10.0.22000.469]
(c) Microsoft Corporation. All rights reserved.

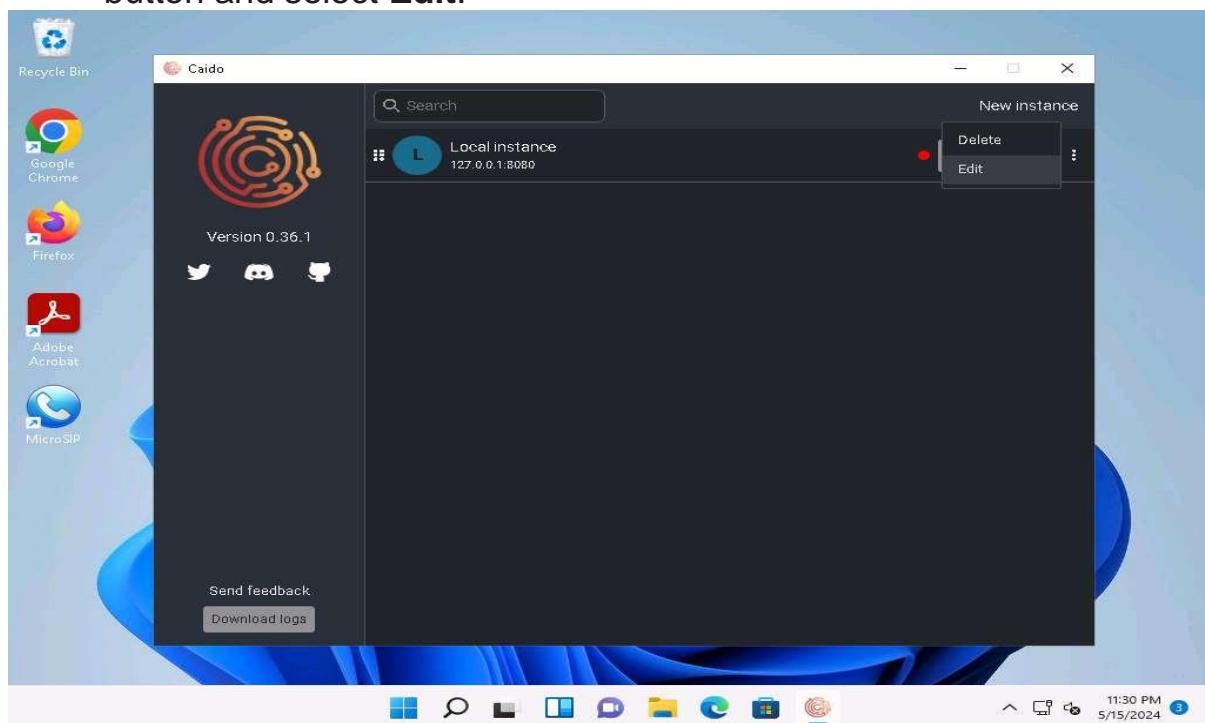
C:\Users\Admin>ipconfig/flushdns

Windows IP Configuration

Successfully flushed the DNS Resolver Cache.

C:\Users\Admin>
```

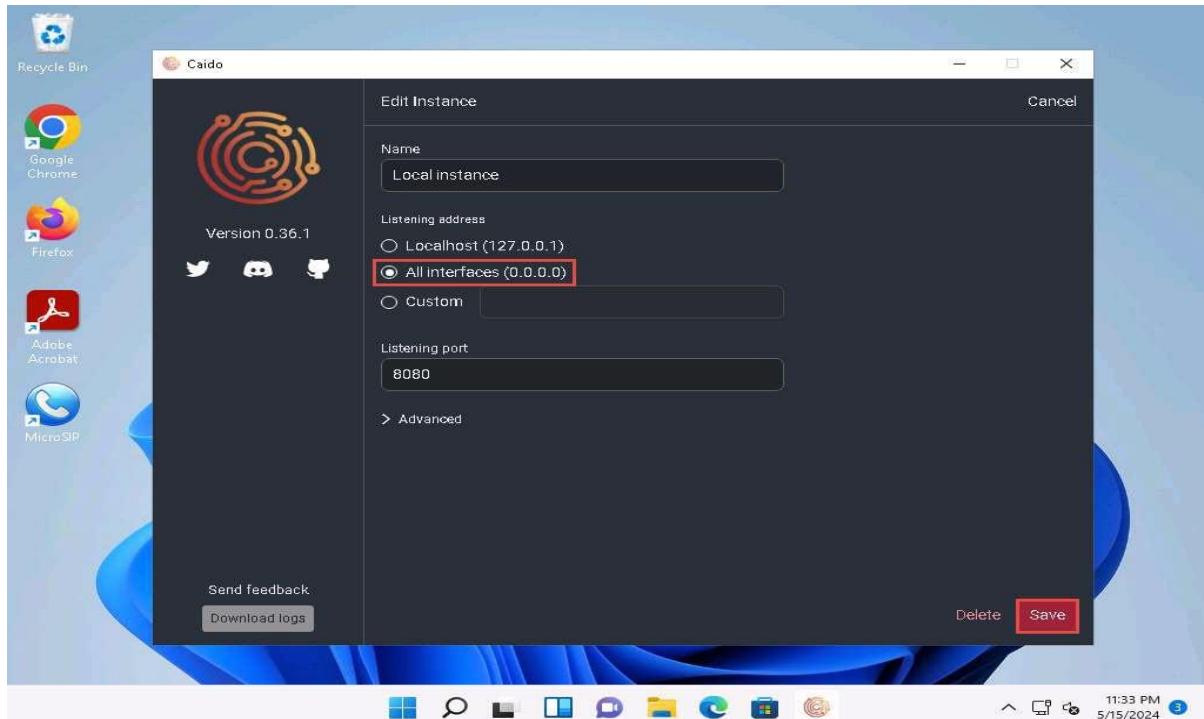
2. Click windows **Search** icon on the **Desktop**, search for **Caido** and launch **Caido** from search bar.
3. **Caido** application window appears, click on **menu** besides Start button and select **Edit**.



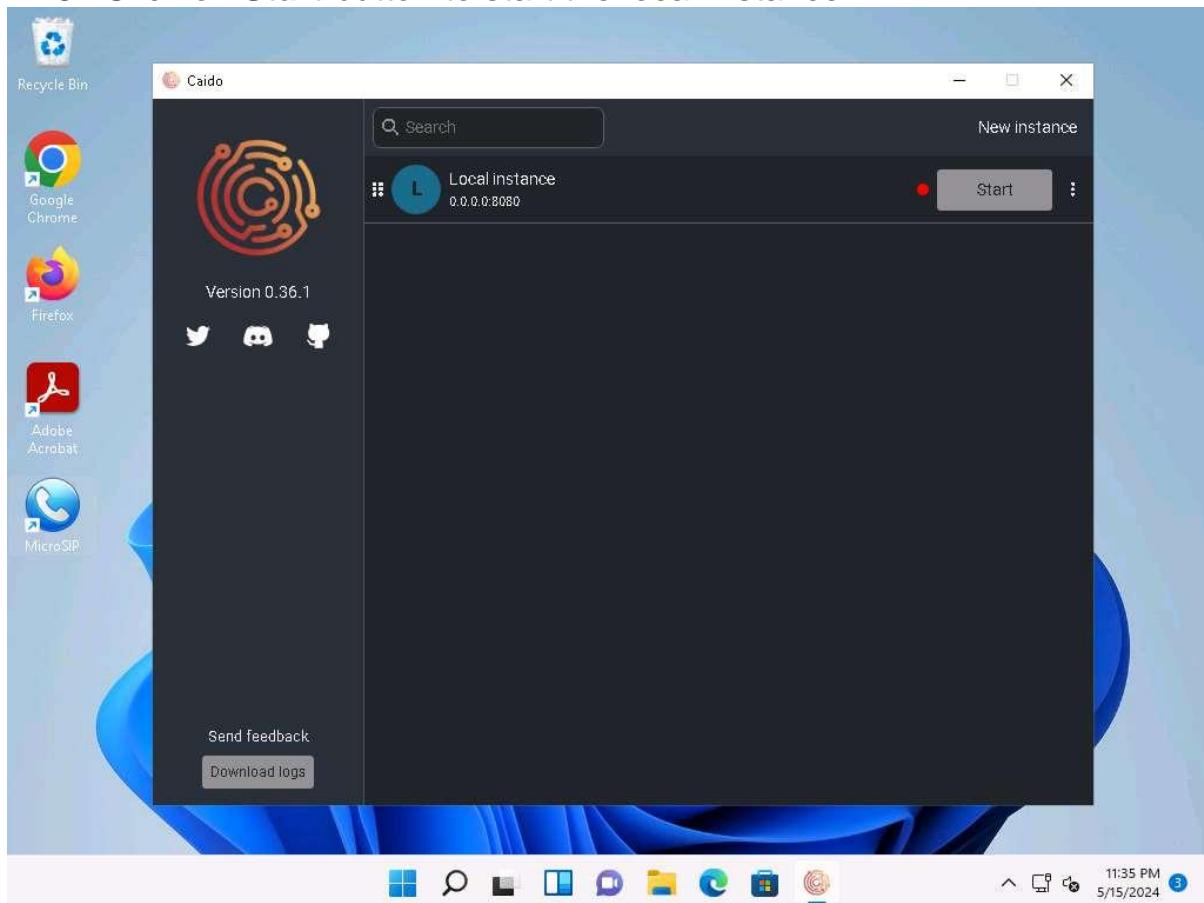
Name : kunal Jawale

Screeshot

4. In **Edit Instance** window, click on the radio button besides **All interfaces (0.0.0.0)** to listen on all the available network interfaces and click on **Save**.

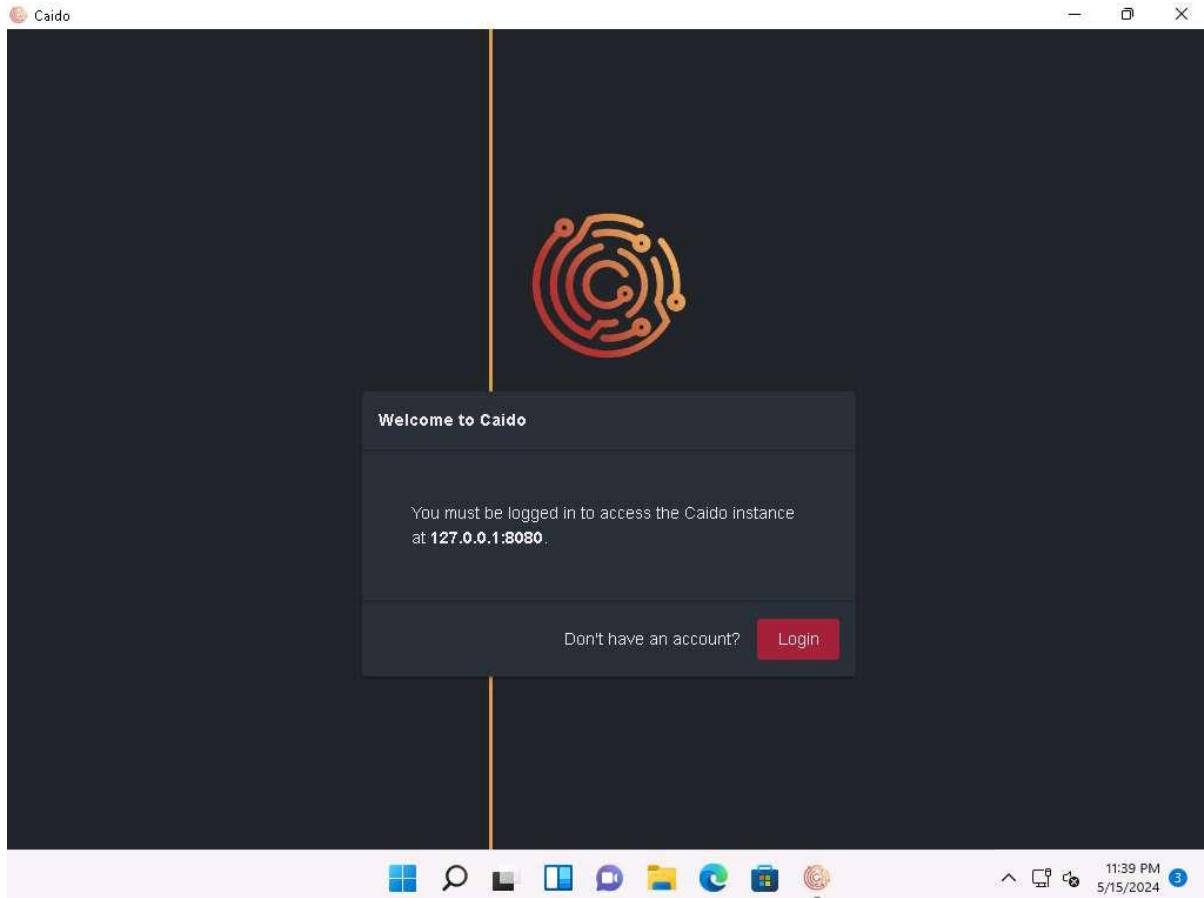


5. Click on **Start** button to start the local instance.



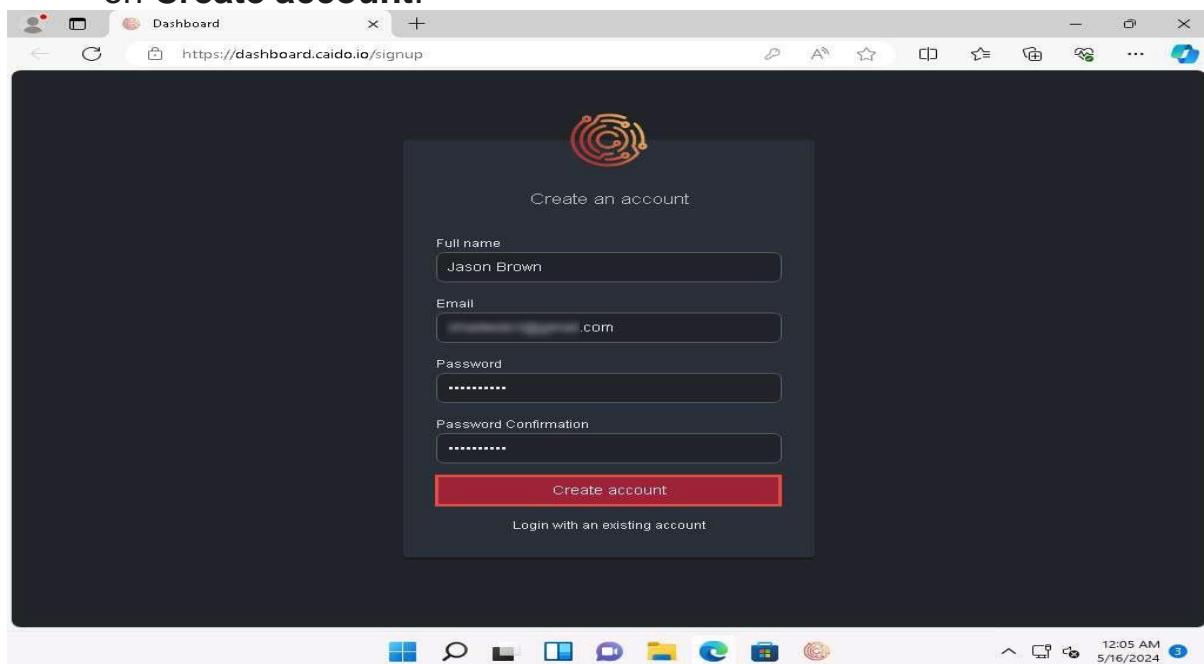
Name : kunal Jawale

6. **Welcome to Caido** pop-up appears, click on **Login** if you have an account already. If not, select **Don't have an account?**, you will be redirected to Dashboard.



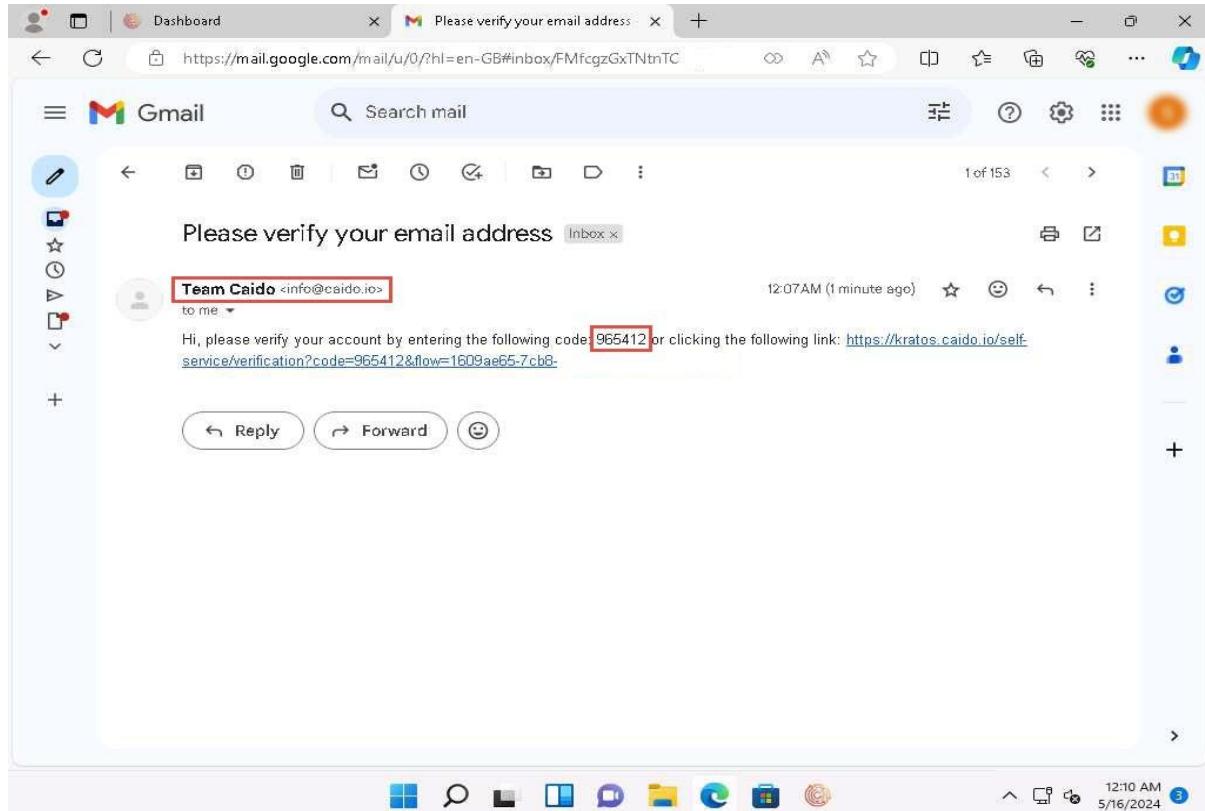
Screenshot

7. **Create an account** window appears, here fill in the details and click on **Create account**.

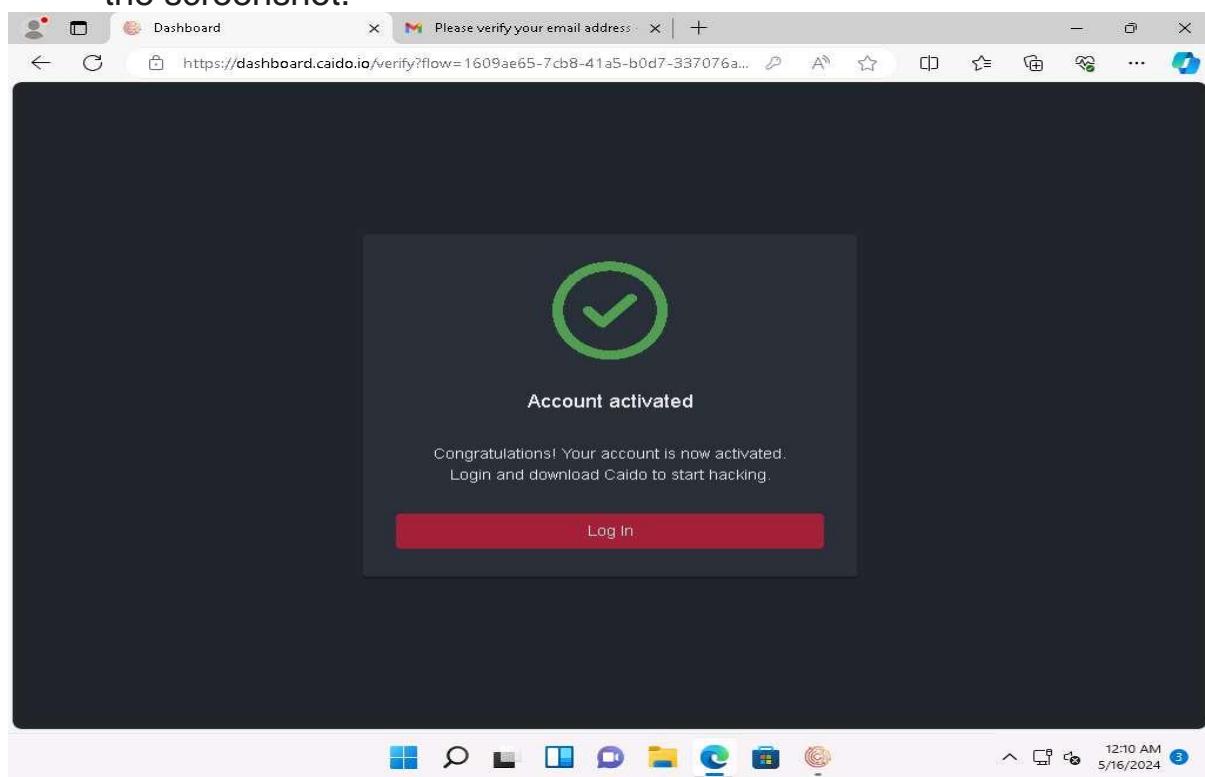


Name : kunal Jawale

8. Login to your mail account, you will receive a verification mail from **Team Caido** copy the code and paste it in the Caido verification window.

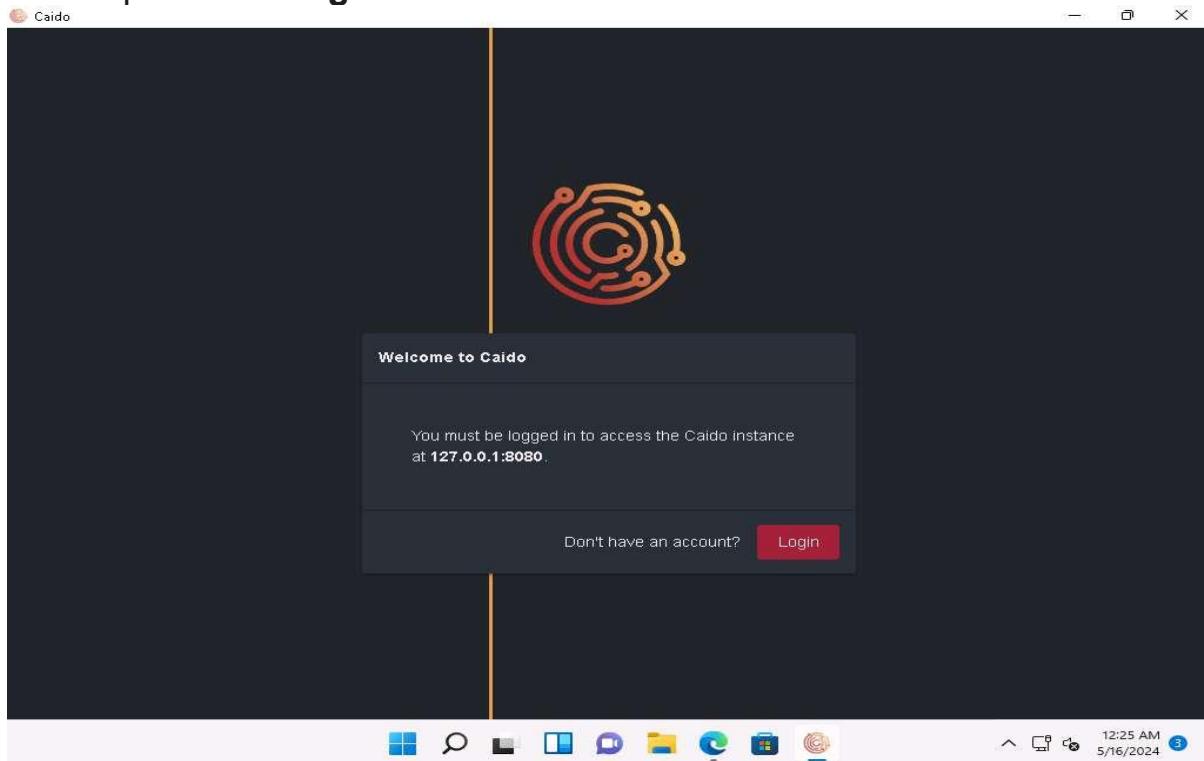


9. After entering the code, your account will be activated as shown in the screenshot.

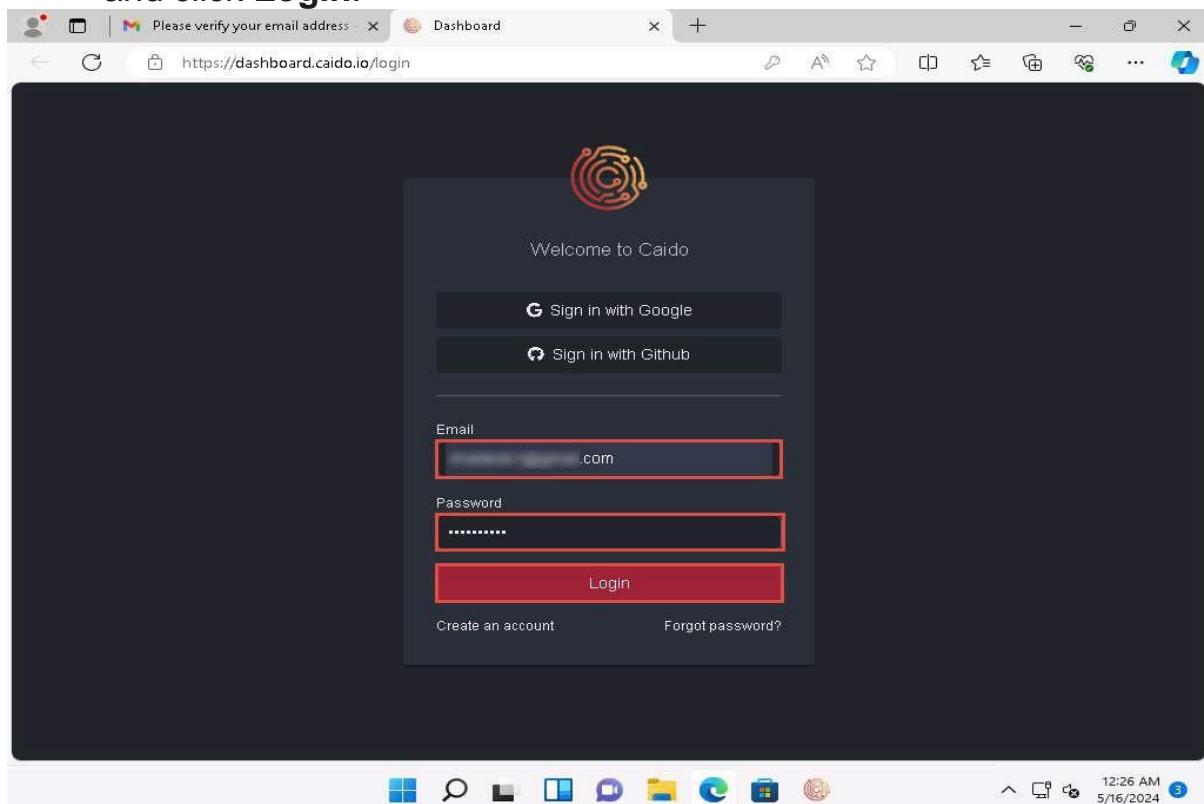


Name : kunal Jawale

10. Navigate back to Caido application, in **Welcome to Caido** pop-up click on **Login**.

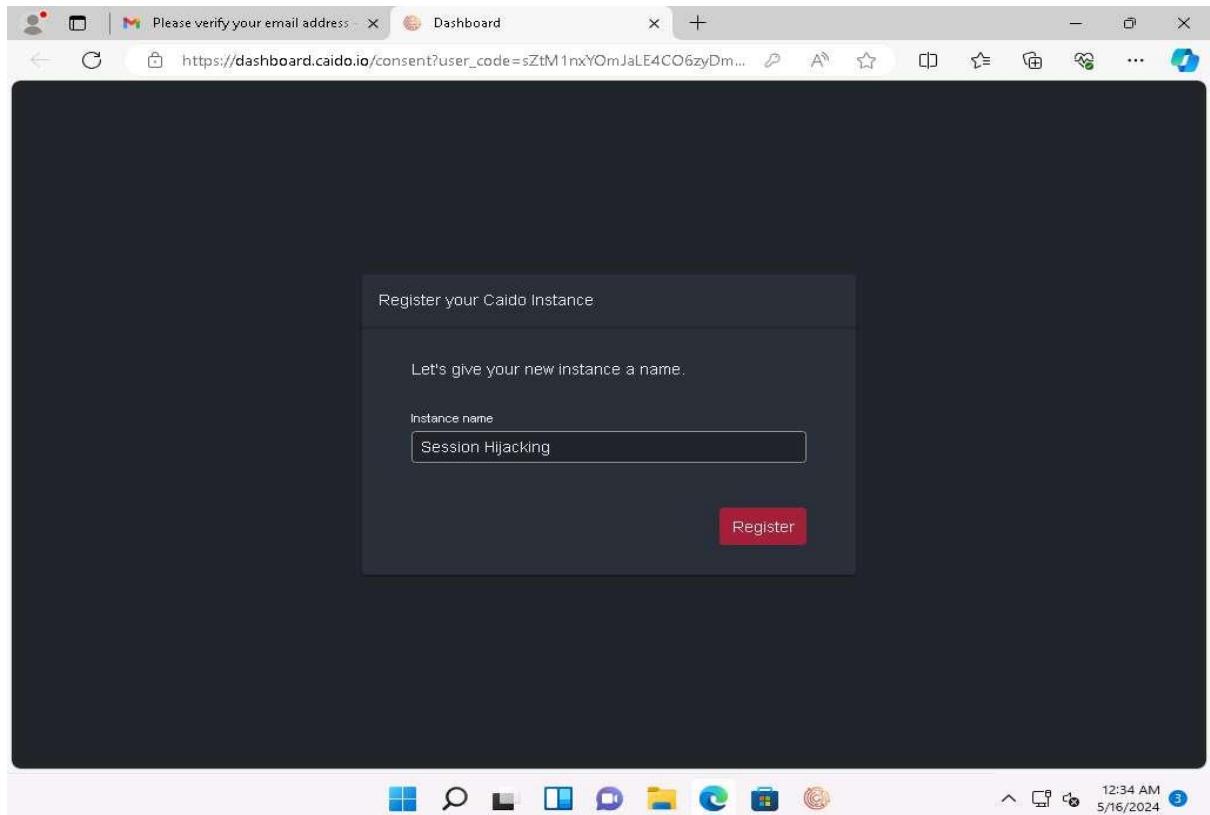


11. **Welcome to Caido** page will appear, enter your credentials and click **Login**.

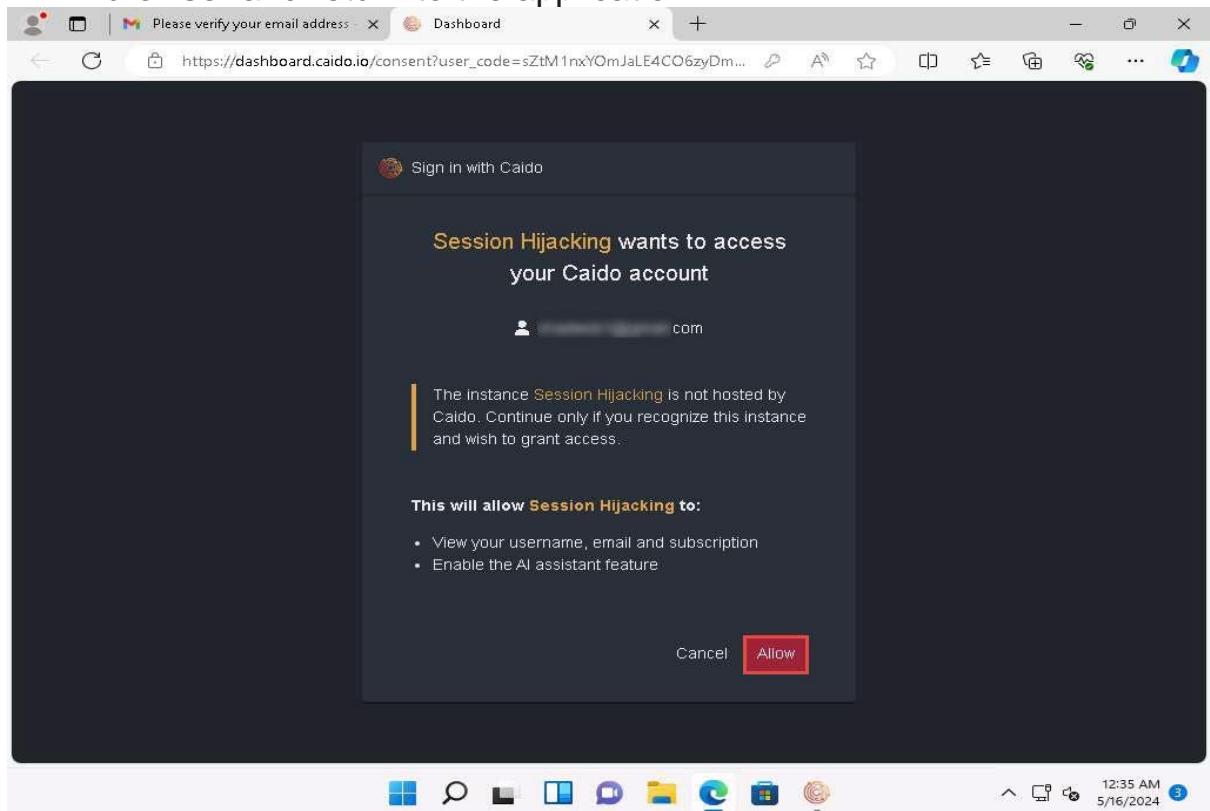


12. Once logged in, **Register your Caido Instance** pop-up will appear. Type **Session Hijacking** and click **Register**.

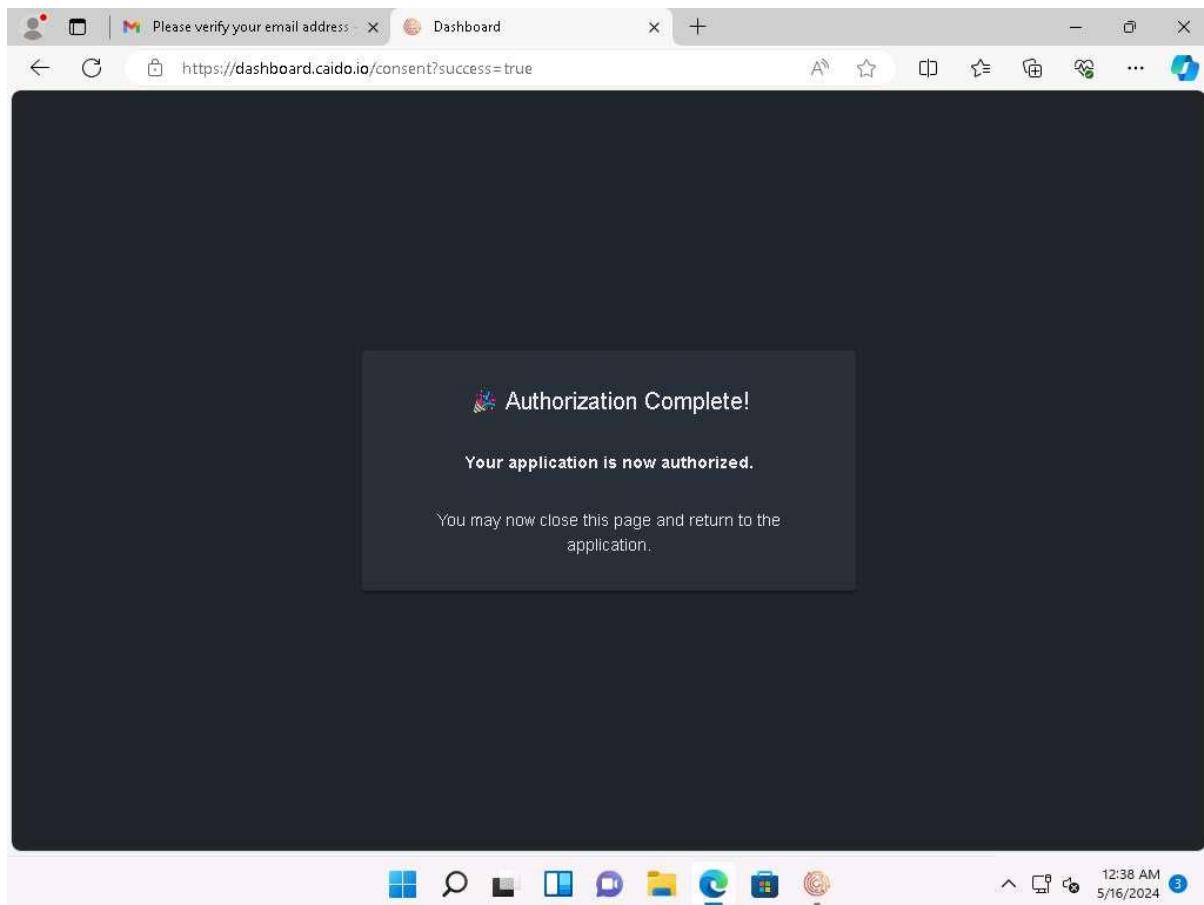
Name : kunal Jawale



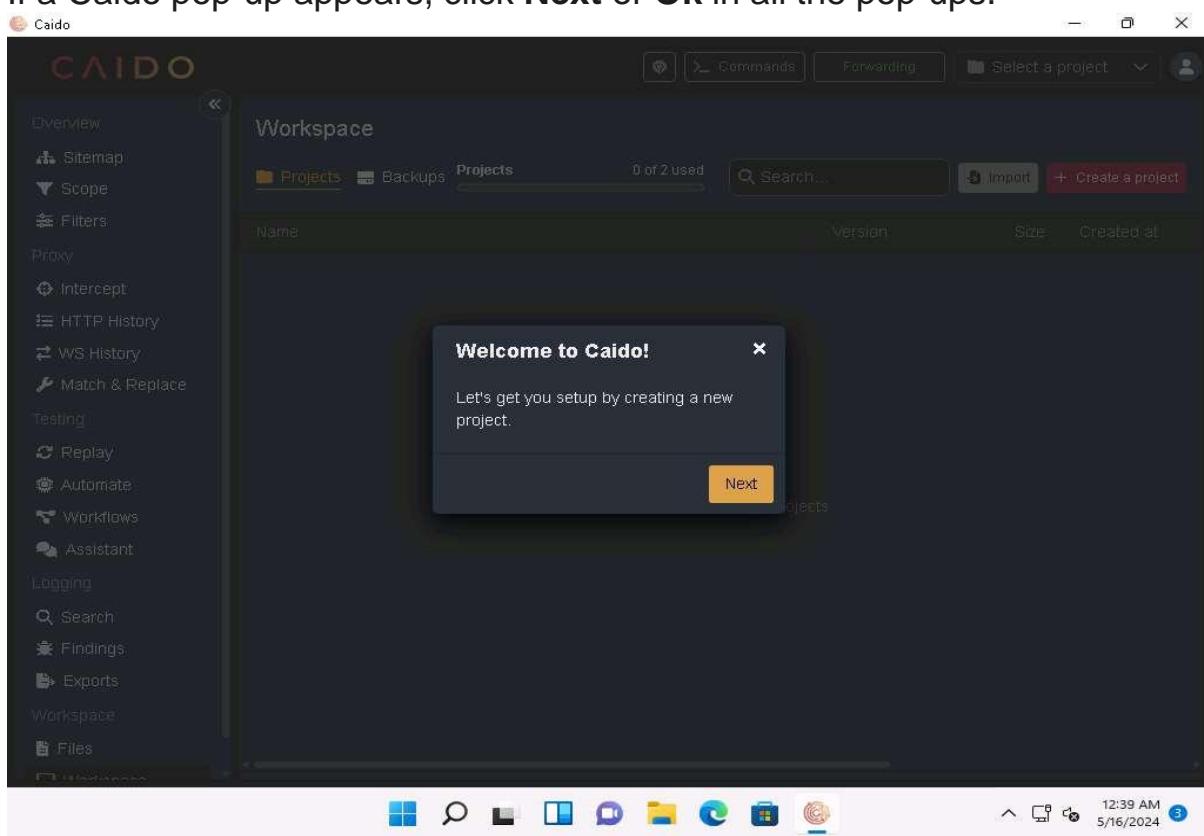
13. **Sign in with Caido** window appears, click **Allow** to allow the access. **Authorization Complete!** pop-up appears, close the web browser and return to the application.



Name : kunal Jawale

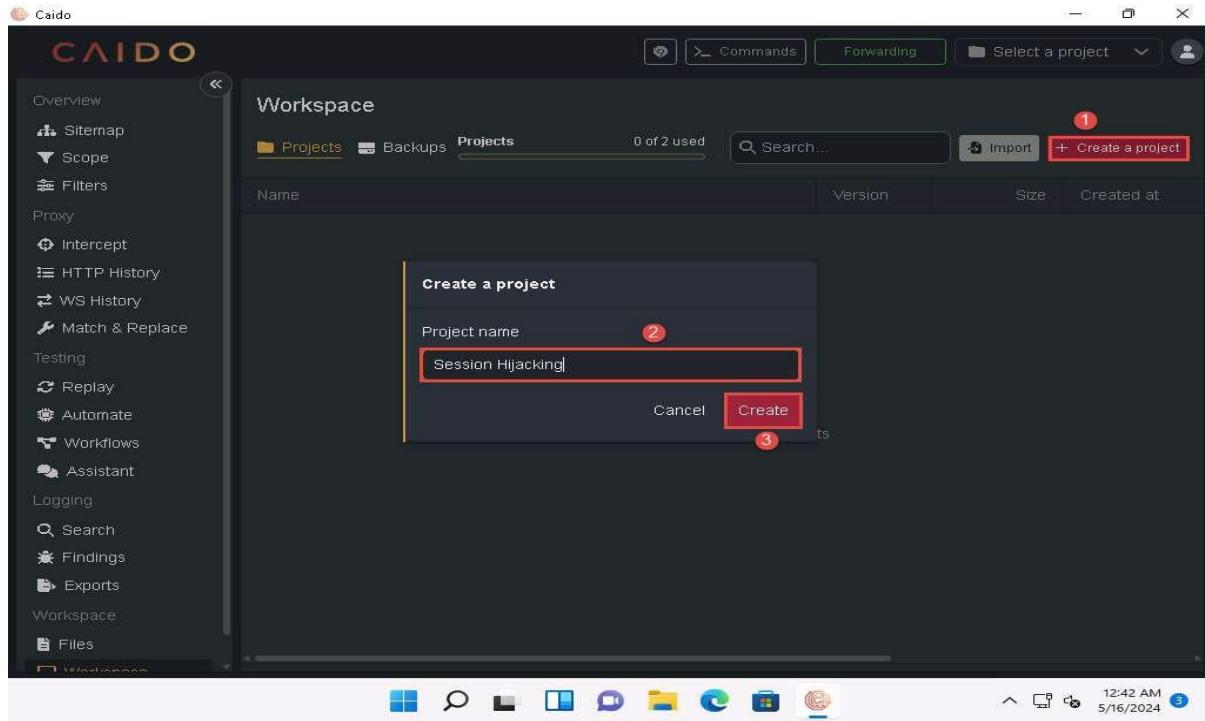


14. The **Caido** main window appears.
If a Caido pop-up appears, click **Next** or **Ok** in all the pop-ups.

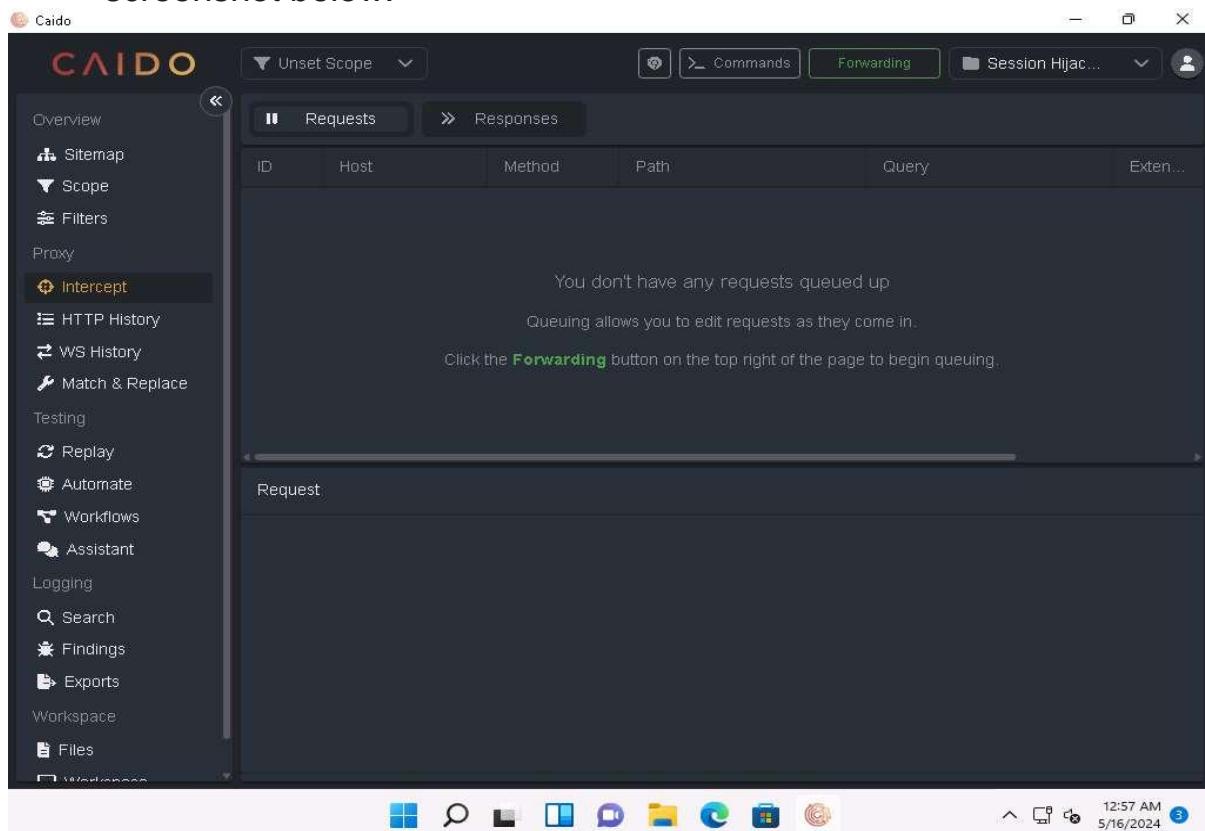


Name : kunal Jawale

15. Click on **+ Create a project** button to create a new project. **Create a project** pop-up appears, name it as **Session Hijacking** and click **Create**.



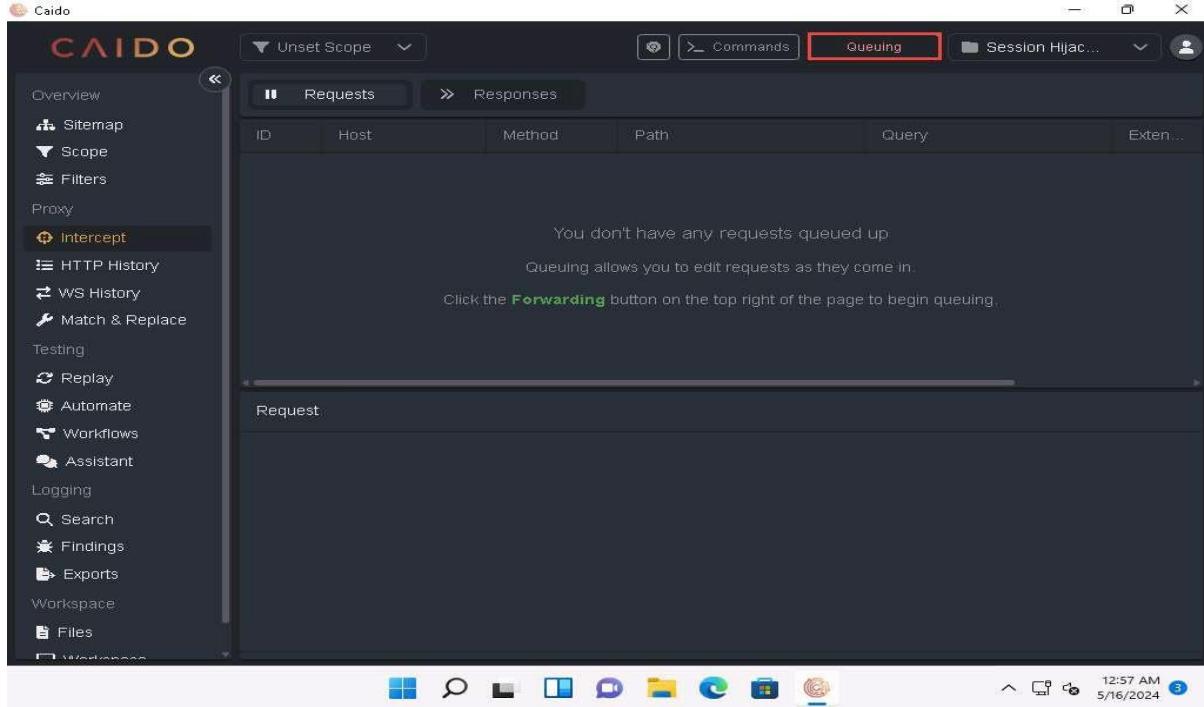
16. Click on **Intercept** option on the left pane, as shown in the screenshot below.



Name : kunal Jawale

17. Click the **Forwarding** icon and wait until it changes to **Queuing**. This button will trap and display the next response or request from the victim's machine in the **Intercept** tab.

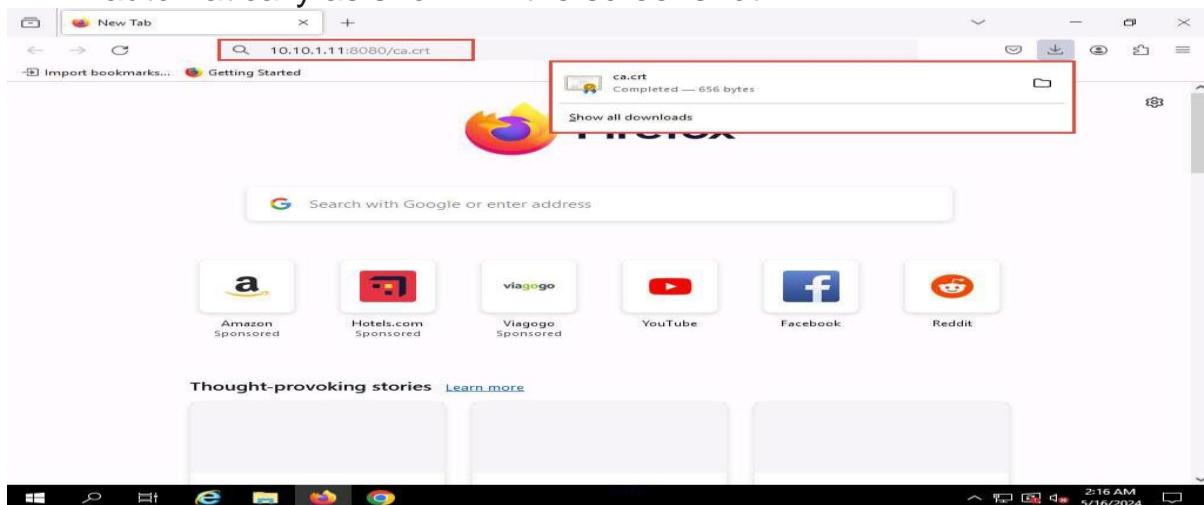
The **Forwarding** icon turns automatically from green to red.



18. Click Windows Server 2019 to switch to the **Windows Server 2019** machine. Click [Ctrl+Alt+Delete](#) to activate the machine and login using **Administrator/Pa\$\$w0rd**.

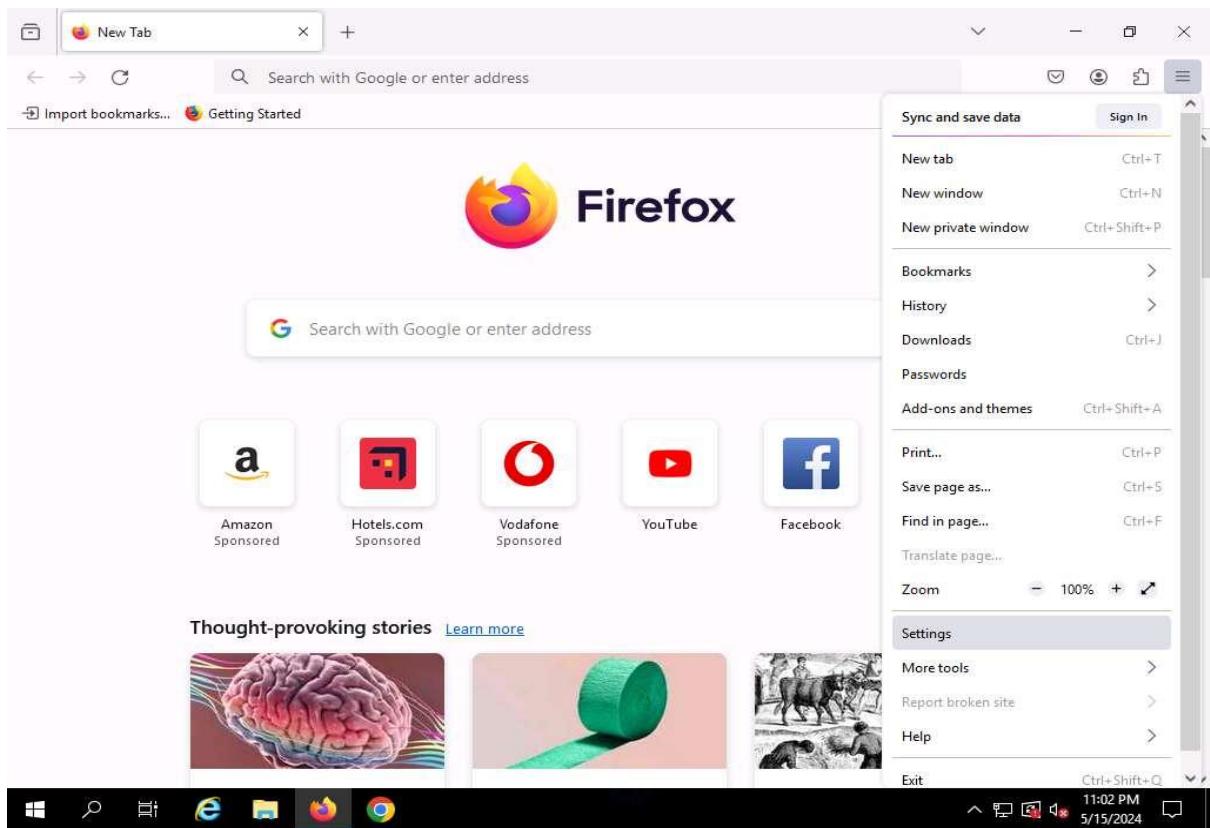
Networks screen appears, click **Yes** to allow your PC to be discoverable by other PCs and devices on the network.

19. Open **Firefox** web browser and navigate to <http://10.10.1.11:8080/ca.crt>. CA certificate will be downloaded automatically as shown in the screenshot.

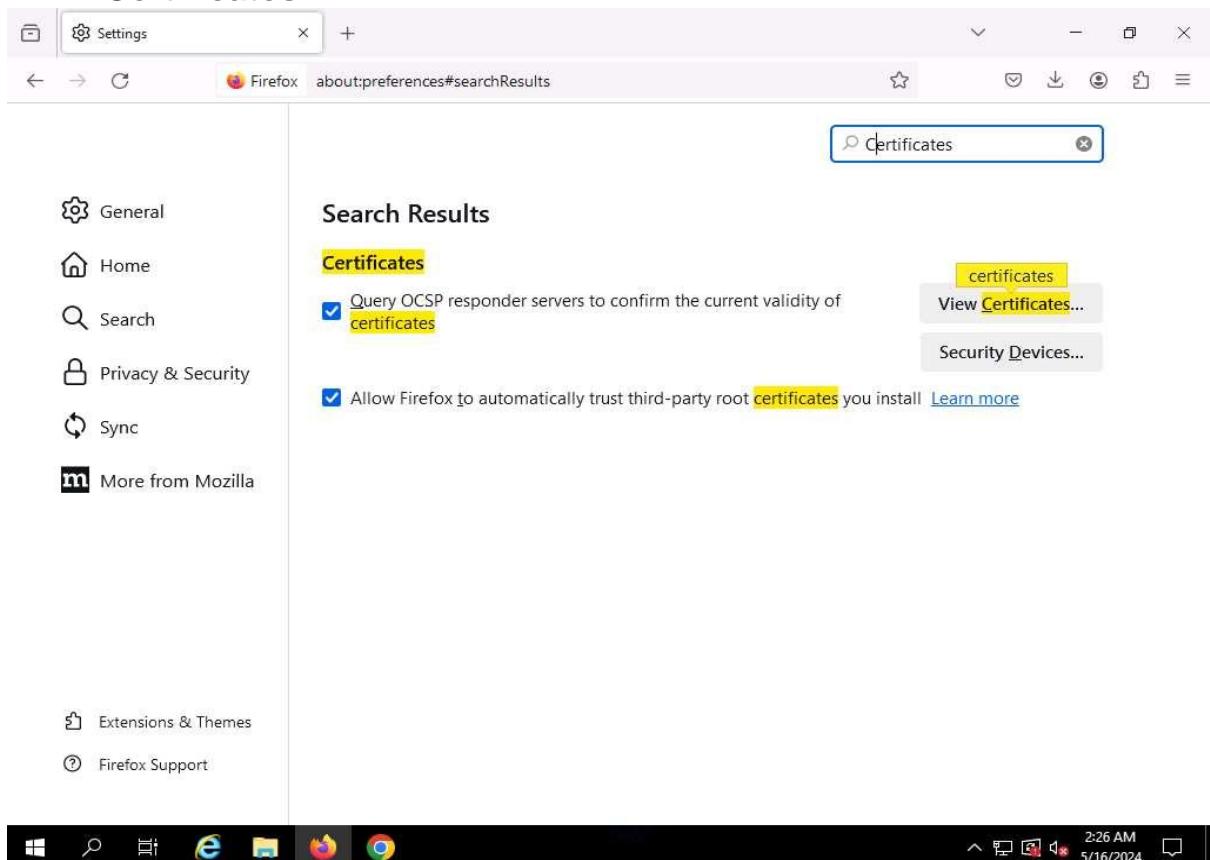


20. In **Firefox** web browser, select **Settings** from the context menu.

Name : kunal Jawale

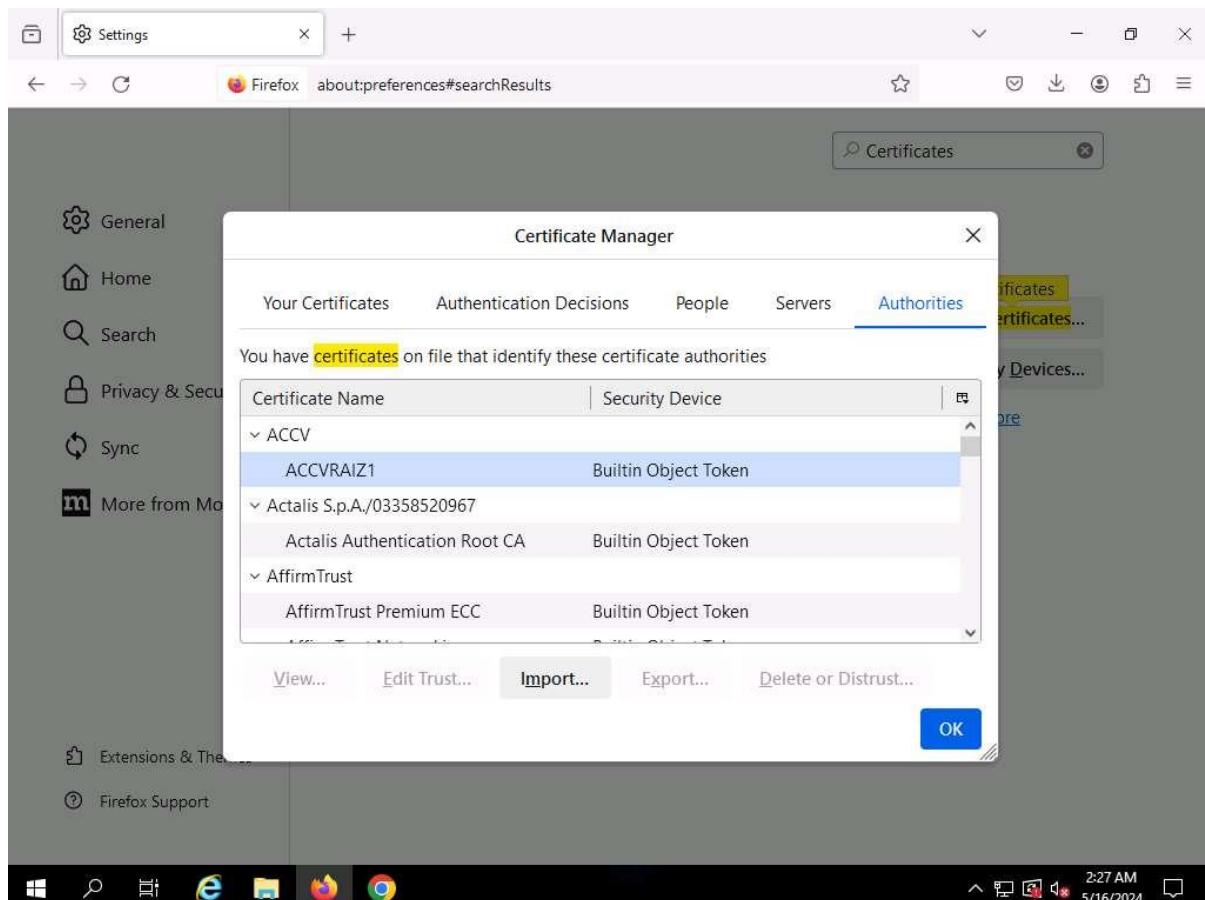


21. On the **Settings** page, search for **Certificates** and open **View Certificates**.

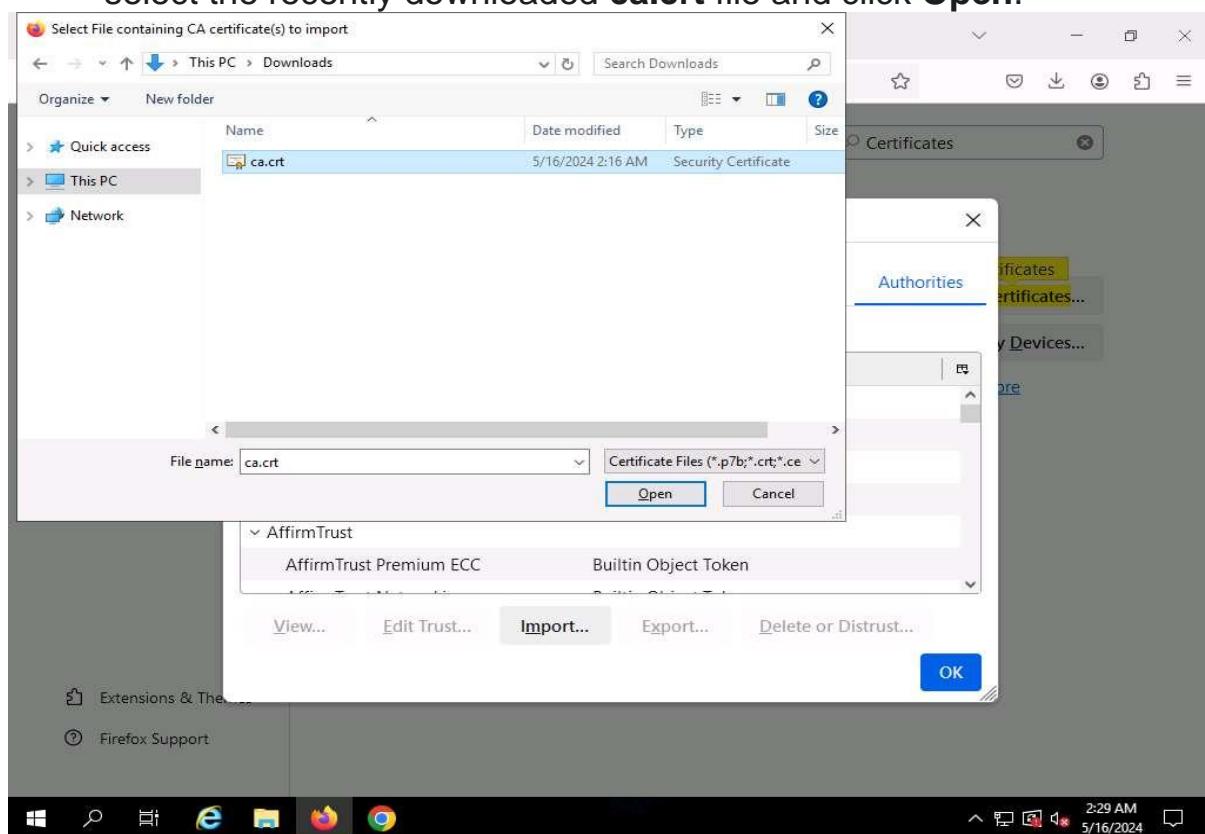


22. Navigate to **Authorities** tab and click on **Import...**

Name : kunal Jawale

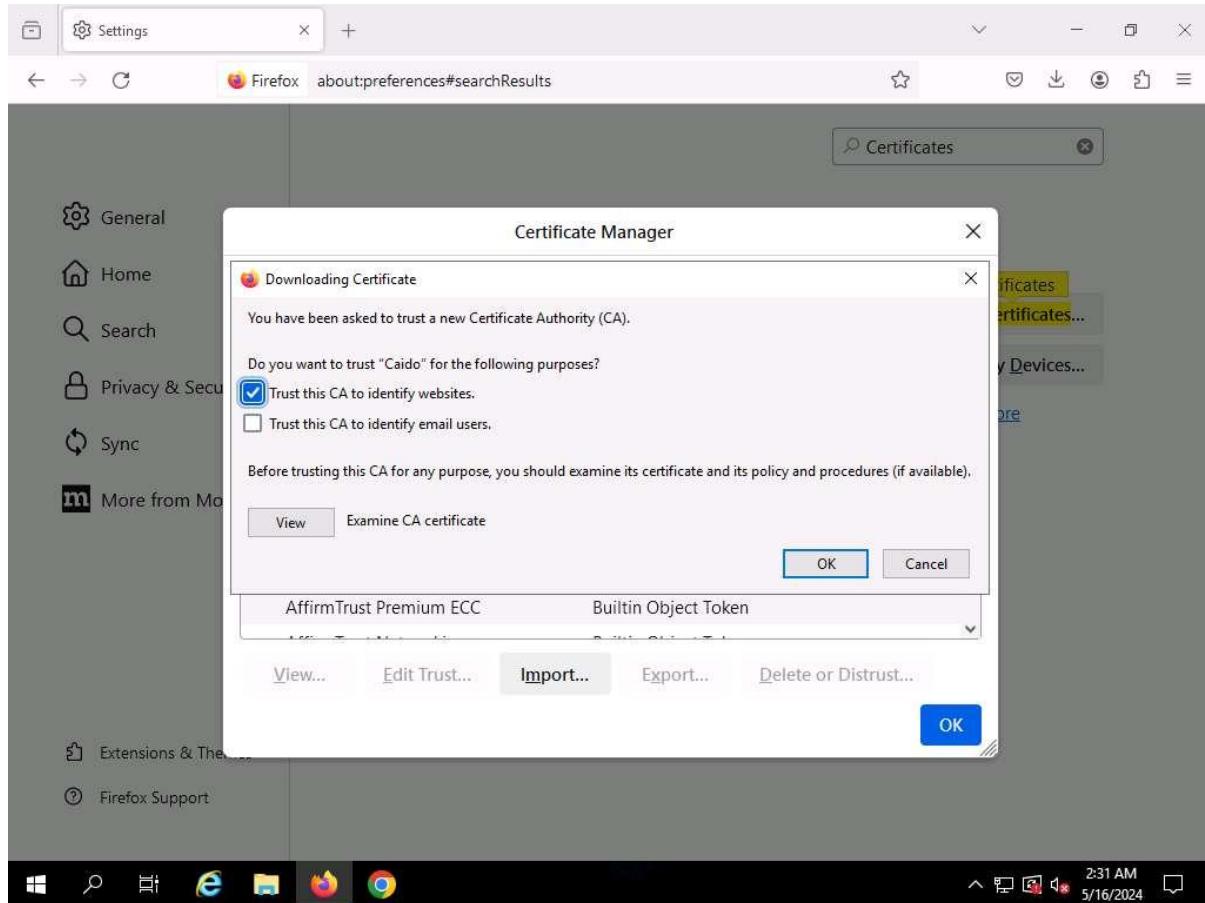


23. In Select File containing CA certificate(s) to import window, select the recently downloaded ca.crt file and click Open.

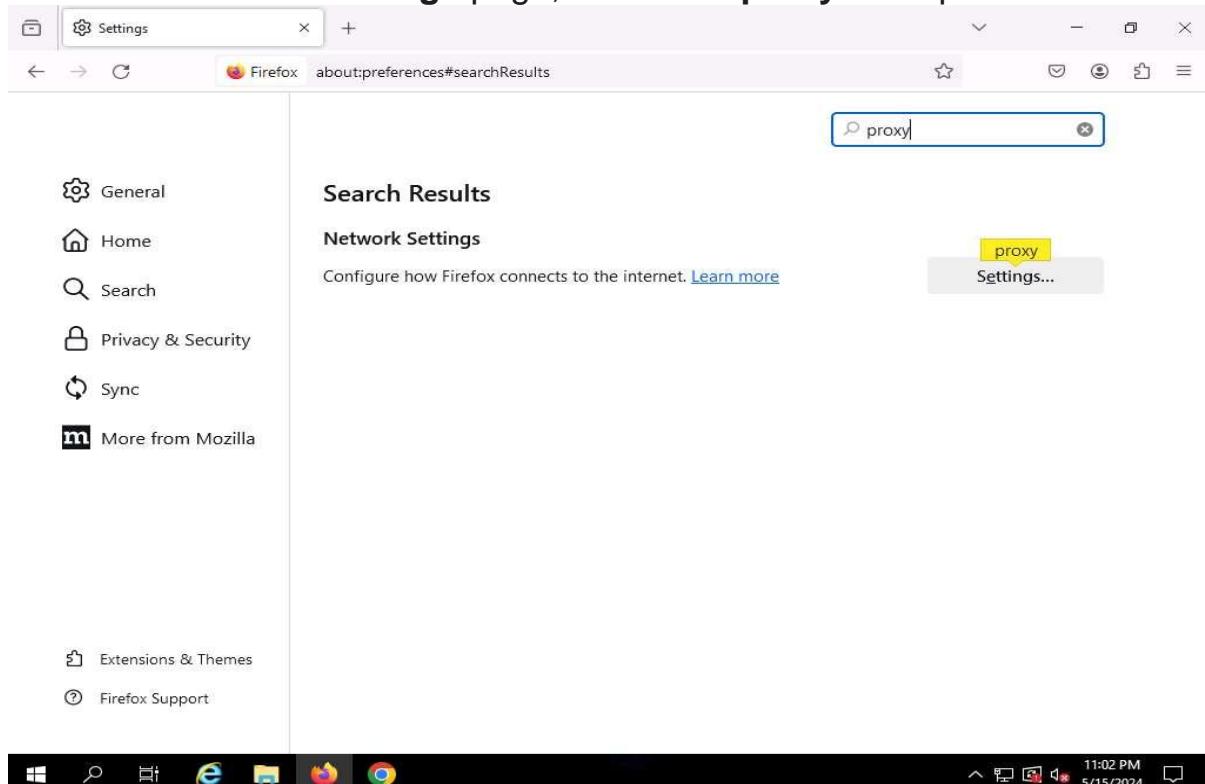


Name : kunal Jawale

24. When prompted, click the **Trust this CA to identify websites** checkbox and click on **OK**. Click **OK** in the **Certificate Manager** window.



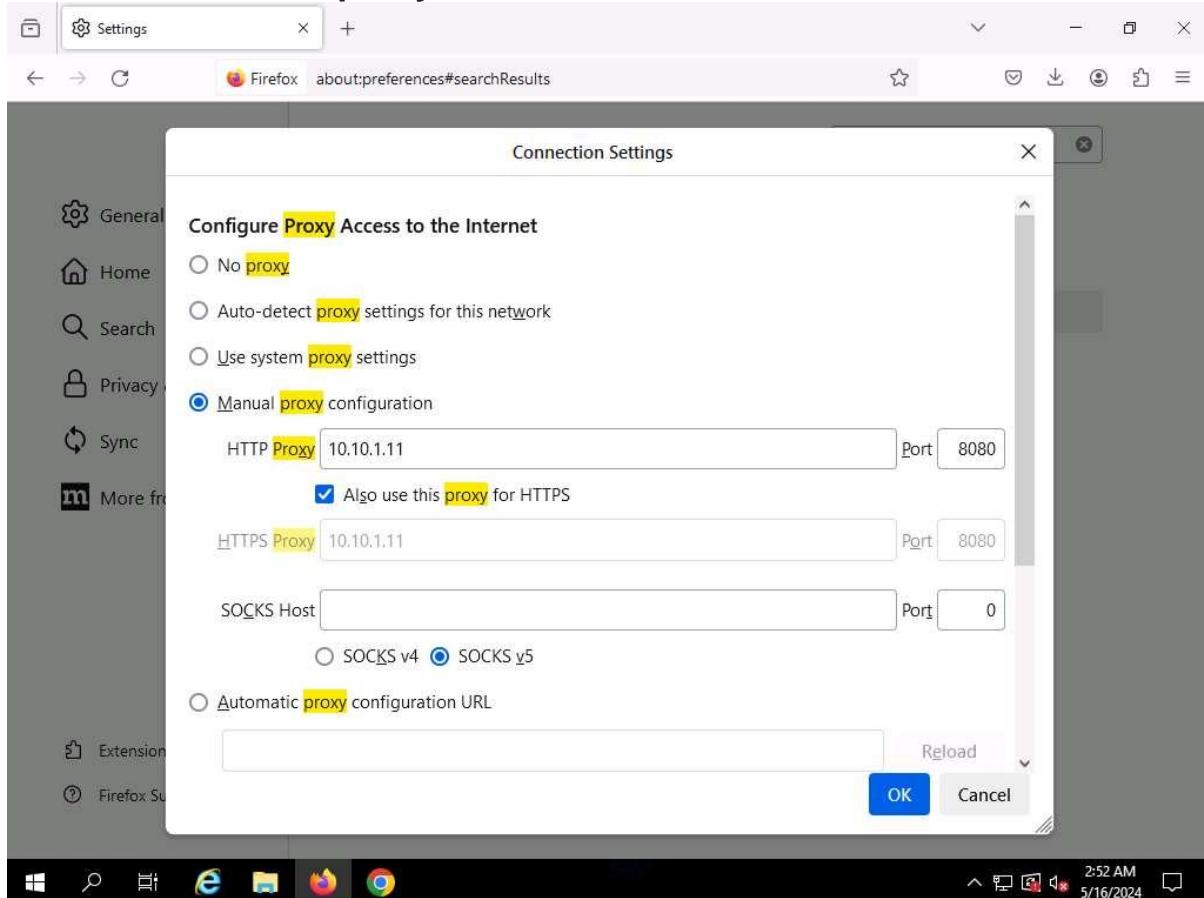
25. On the **Settings** page, search for **proxy** and open it.



Name : kunal Jawale

26. **Connection Settings** page appears and click **Manual proxy configuration** to configure a proxy.

27. Set HTTP Proxy to **10.10.1.11** and port to **8080**, check the **Also use this proxy for HTTPS** box and click **OK**.



28. After saving, close the **Settings** and browser windows. You have now configured the proxy settings of the victim's machine.

29. Open a new tab in **Firefox** web browser and place your mouse cursor in the address bar, type **www.moviescope.com** and press **Enter**.

30. If a message appears, stating that **Your connection is not private**. Click the **Advanced** button.

31. On the next page, click **Proceed to www.moviescope.com (unsafe)** to open the website.

Name : kunal Jawale

32. Now, click Windows 11 to switch back to the attacker machine (**Windows 11**) and observe that **Caido** has begun to capture the requests of the victim's machine.

The screenshot shows the Caido proxy tool running on a Windows 11 desktop. The left sidebar contains various menu items like Sitemap, Scope, Filters, Proxy, Intercept (highlighted), HTTP History, WS History, Match & Replace, Testing, Replay, Automate, Workflows, Assistant, Logging, Search, Findings, Exports, Workspace, and Files. The main area has tabs for Requests and Responses, with Requests selected. A table lists captured requests with columns for ID, Host, Method, Path, Query, and Extension. Most requests are from 'detectportal.firefox.com' and 'www.google.co...'. Below the table is a detailed view of a request to 'http://detectportal.firefox.com'. The bottom status bar shows the date and time as 5/16/2024 3:00 AM.

33. On the **Requests** tab, for all **www.moviescope.com** requests, modify **www.moviescope.com** to **www.goodshopping.com** in all the captured GET requests and **Forward** all the requests.

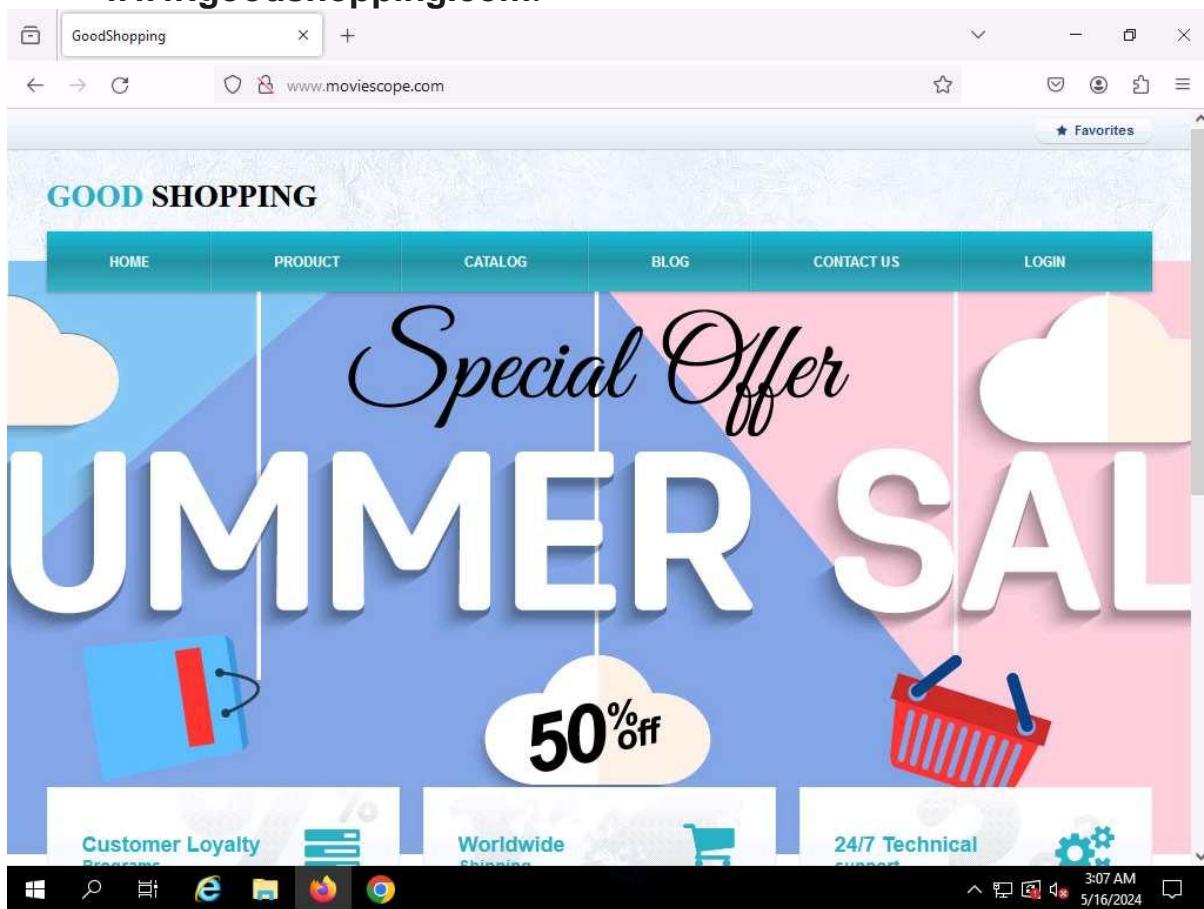
This screenshot shows the Caido interface after modifying the host for 'www.moviescope.com' requests. The 'Intercept' section in the sidebar shows 88 requests. The Requests table lists numerous GET requests from 'www.moviescope.com' to various paths like '/font/fontawesome-webfont.woff', '/images/shadow_right.jpg', etc. Below the table, a detailed view shows a request to 'http://www.moviescope.com'. The host has been changed to 'www.goodshopping.com'. The bottom status bar shows the date and time as 5/16/2024 3:05 AM.

34. In a similar way, modify every **GET** request captured by **Caido** until you see the **www.goodshopping.com** page in the victim's machine. You will need to switch back and forth from the victim's machine to see the browser status while you do this.

If you do not receive any request or you see a blank Requests tab then switch to **Windows Server 2019** machine and refresh the browser to capture the request again.

35. Now, click on Windows Server 2019 to switch to the victim's machine (**Windows Server 2019**); the browser displays the website that the attacker wants the victim's machine to see (in this example, www.goodshopping.com).

36. The victim has navigated to **www.moviescope.com**, but now sees **www.goodshopping.com**; while the address bar displays **www.moviescope.com**, the window displays **www.goodshopping.com**.



37. Now, we shall change the proxy settings back to the default settings. To do so, in the **Firefox** browser, select **Settings** from the context menu. On the **Settings** page, search for **proxy** and open it. **Connection Settings** page appears, check **No Proxy** radio button and click **OK**.

1. Detect Session Hijacking using Wireshark

Launch MITM attack

1. Run **bettercap -iface eth0** to set the network interface.

-iface: specifies the interface to bind to (here, **eth0**).

```
[attacker@parrot]~$ sudo su
[sudo] password for attacker:
[root@parrot]#cd
[root@parrot]#bettercap -iface eth0
bettercap v2.32.0 (built for linux amd64 with go1.19.8) [type 'help' for a list of commands]

10.10.1.0/24 > 10.10.1.13 » [09:02:10] [sys.log] [inf] gateway monitor started ...
10.10.1.0/24 > 10.10.1.13 »
```

2. Type **net.probe on** and press **Enter**. This module will send different types of probe packets to each IP in the current subnet for the **net.recon** module to detect them.
3. Type **net.recon on** and press **Enter**. This module is responsible for periodically reading the system ARP table to detect new hosts on the network.

Name : kunal Jawale

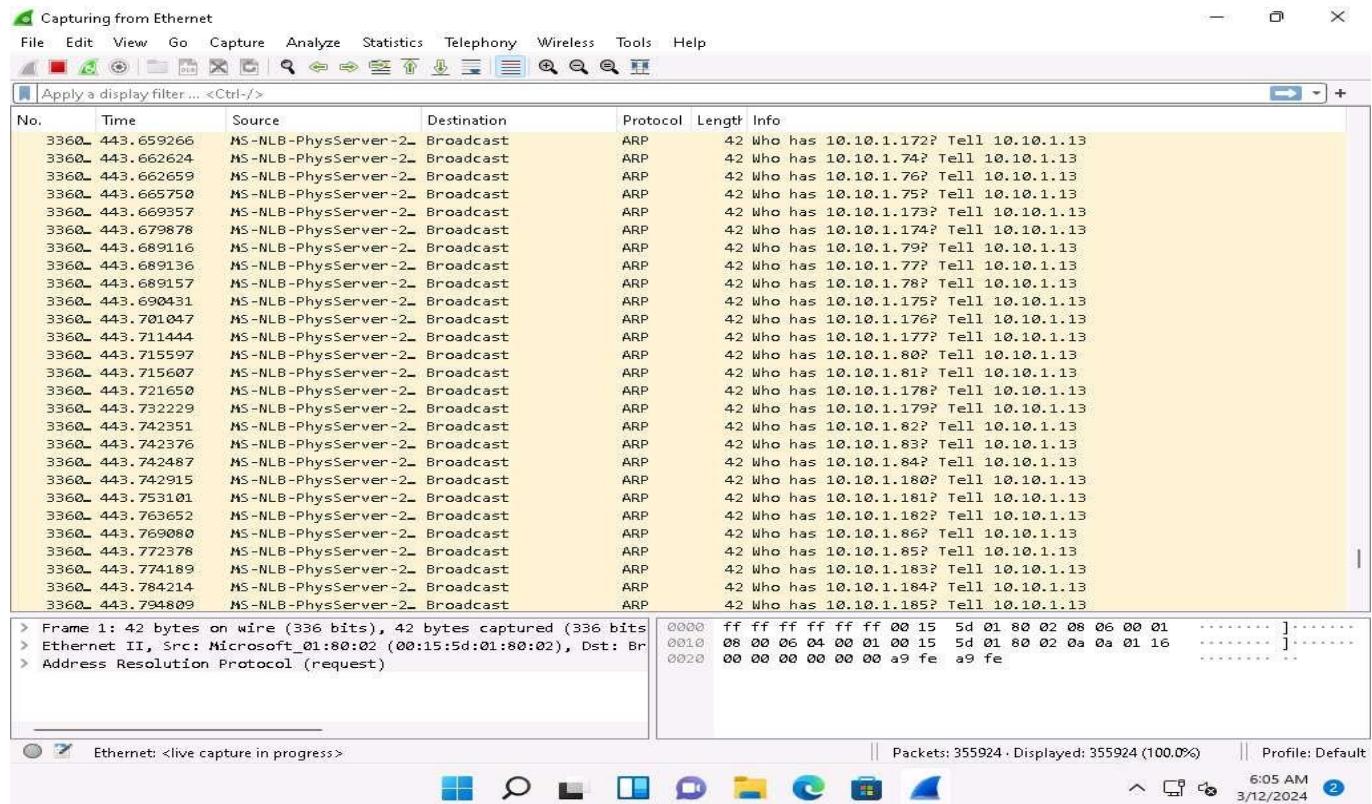
The net.recon module displays the detected active IP addresses in the network. In real-time, this module will start sniffing network packets.

4. Type **net.sniff on** and press **Enter**. This module is responsible for performing sniffing on the network.
5. You can observe that bettercap starts sniffing network traffic on different machines in the network, as shown in the screenshot.

```
[root@parrot]~[~/home/attacker]
└─#cd
[root@parrot]~[~]
└─#bettercap -iface eth0
bettercap v2.32.0 (built for linux amd64 with go1.19.8) [type 'help' for a list of commands]

10.10.1.0/24 > 10.10.1.13 » [09:02:10] [sys.log] [inf] gateway monitor started ...
10.10.1.0/24 > 10.10.1.13 » net.probe on
[09:04:08] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
10.10.1.0/24 > 10.10.1.13 » [09:04:08] [sys.log] [inf] net.probe probing 256 addresses on 10.10.1.0/24
10.10.1.0/24 > 10.10.1.13 » [09:04:08] [endpoint.new] endpoint 10.10.1.9 detected as 02:15:5d:45:41:2f.
10.10.1.0/24 > 10.10.1.13 » [09:04:08] [endpoint.new] endpoint 10.10.1.11 (WINDOWS11) detected as 00:15:5d:01:80:00 (Microsoft Corporation).
10.10.1.0/24 > 10.10.1.13 » [09:04:09] [endpoint.new] endpoint 10.10.1.19 (SERVER2019) detected as 02:15:5d:45:41:2e.
10.10.1.0/24 > 10.10.1.13 » [09:04:09] [endpoint.new] endpoint 10.10.1.22 (SERVER2022) detected as 00:15:5d:01:80:02 (Microsoft Corporation).
10.10.1.0/24 > 10.10.1.13 » net.recon on
10.10.1.0/24 > 10.10.1.13 » [09:04:21] [sys.log] [err] module net.recon is already running
10.10.1.0/24 > 10.10.1.13 » net.sniff on
10.10.1.0/24 > 10.10.1.13 » [09:04:46] [net.sniff.mdns] mdns fe80::15:5dff:fe45:4130 : Android.local is fe80::15:5dff:fe45:4130
10.10.1.0/24 > 10.10.1.13 »
```

A huge number of Arp requests, indicate attack in progress.



Lab 2: Detect Session Hijacking

Overview of Detecting Session Hijacking

There are two primary methods that can be used to detect session hijacking:

- **Manual Method:** Involves using packet sniffing software such as Wireshark and SteelCentral Packet Analyzer to monitor session hijacking attacks; the packet sniffer captures packets being transferred across the network, which are then analyzed using various filtering tools.
- **Automatic Method:** Involves using Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) to monitor incoming network traffic; if a packet matches any of the attack signatures in the internal database, the IDS generates an alert, and the IPS blocks the traffic from entering the database.

Name : kunal Jawale