**Browser analysis during normal browsing sessions-**

1. Internet explorer -Temp File Directory files (Content.IE, History.IE5, Cookies, Recovery, Custom Destinations, Index.dat) are created, modified, and deleted
2. Google chrome - Directory Chrome\User Data (Safe Browsing Whitelist, Default\ Cache, Current Session, Default\History, Default\Session Storage) files are created, modified, and deleted
3. Firefox- Directory Firefox\Profiles (Cache, jumpListCache, etc.) and Win CustomDestinations, files are created, modified, and deleted
4. Safari - Directory AppleComputer\Safari (Cache, History, Webpage Previews, Cookies, WebpageIcons.db) files are created, modified, and deleted

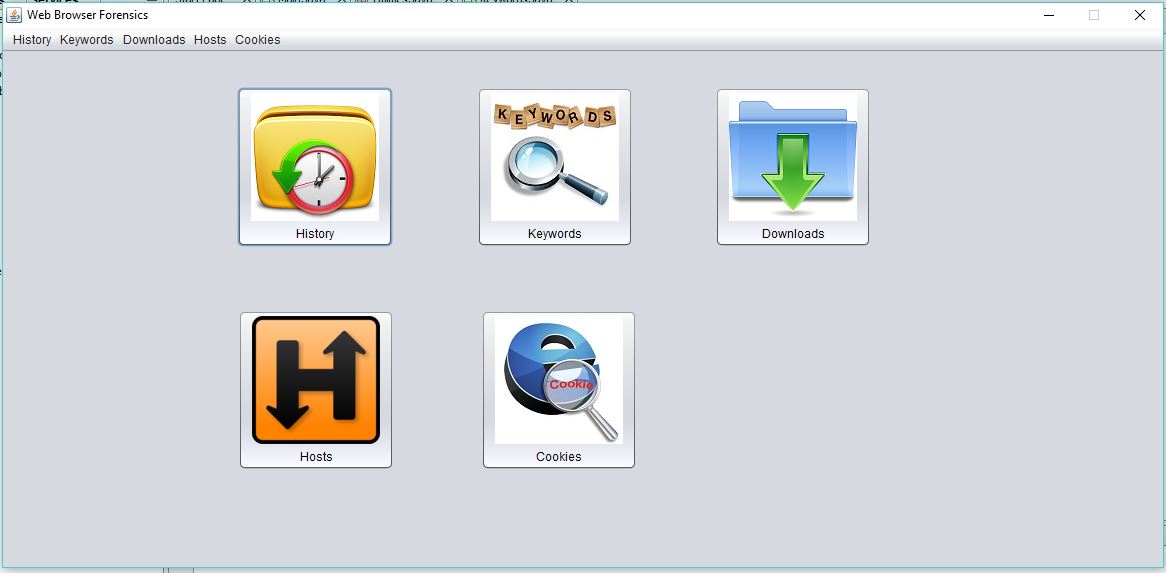
**Browser analysis during private browsing sessions-**

1. IE - InPrivate Browsing Everything gets deleted when exiting the browser and the entire session is terminated
2. Google Chrome - Incognito Mode Safe Browsing databases, Cookies, and History are modified, no changes during session but the chrome\_shutdown\_ms.txt is replaced with a new timestamp when session ends
3. Firefox - Private Browsing Safe Browsing database gets modified, nothing appears to be written while surfing, but when session ends, some Firefox\Profile files are modified
4. Safari - Private Browsing Only NTuser.dat appears to be modified

|  |  |  |
| --- | --- | --- |
| **Browser Evidences and Possible attacks on it** | | |
| **Path** | **Attack** | **Description** |
| 1. Cache | 1. Cache Poisoning | Cache poisoning is a type of attack in which corrupt data is inserted into the cache database of the Domain Name System (DNS) name server. The Domain Name System is a system that associates domain names with IP addresses. |
|  | | |
|  | 1. Cache Sniffing | First the attacker uses a sniffer to capture a valid cache then made changes into cache. |
|  | | |
| 1. Cookies | 1. Session Hijacking | The Session Hijacking attack consists of the exploitation of the web session control mechanism, which is normally managed for a session token. Because http communication uses many different TCP connections, the web server needs a method to recognize every user’s connections. |
|  | | |
|  | 1. Session Sniffing | We can see, first the attacker uses a sniffer to capture a valid token session called “Session ID”, then he uses the valid token session to gain unauthorized access to the Web Server. |
|  | | |
|  | Cross-site script attack | The attacker can compromise the session token by using malicious code or programs running at the client-side. The example shows how the attacker could use an XSS attack to steal the session token. If an attacker sends a crafted link to the victim with the malicious JavaScript, when the victim clicks on the link, the JavaScript will run and complete the instructions made by the attacker. |
|  | | |
| 1. History | 1. Denial of Service | Show error message to access the page |
|  | | |
|  | 1. DDoS | DDoS attacks are quick to start killing performance on the server. The first clue that you’re under an attack is a server crash. With IIS, the server often returns a 503 “Service Unavailable” error. It usually starts intermittently displaying this error, but heavy attacks lead to permanent 503 server responses for all of your users. |
|  | | |
| 1. Bookmarks | 1. Drive-by attack | Generally we save bookmarks for fast access, and when needed that pull up your laptop and check your favorite news website. Once you’re on the page, the malicious code embedded into the website (usually an [exploit kit](https://heimdalsecurity.com/glossary/exploit-kit)) starts scanning your computer. |
|  | | |
|  | 1. Malware inserted in computer | **This**[happens all the time](https://heimdalsecurity.com/blog/compromised-websites-ransomware/)**.** Not because website owners want to imperil their visitors, but because software is not flawless and websites get hijacked. |
|  | | |
| 5) Downloads | 1. **malware infection without clicking or downloading anything** | This was the norm until a few years ago. As cybercriminals refined their tactics, they found ways to spread their malicious software without requiring user/victim interaction. |
|  | | |
|  | 2. **Attackers compromise websites and embed malicious elements inside** | These elements can range from [malicious JavaScript code injects](https://heimdalsecurity.com/blog/javascript-malware-explained/), to [malvertisements](https://heimdalsecurity.com/glossary/malvertisement" \t "_blank), malicious redirects, [cross-site scripting attacks](https://heimdalsecurity.com/glossary/cross-site-scripting) (also known as XSS), malicious iFrames that execute invisibly or other subtle attack techniques that potential victims can’t spot on their own. |

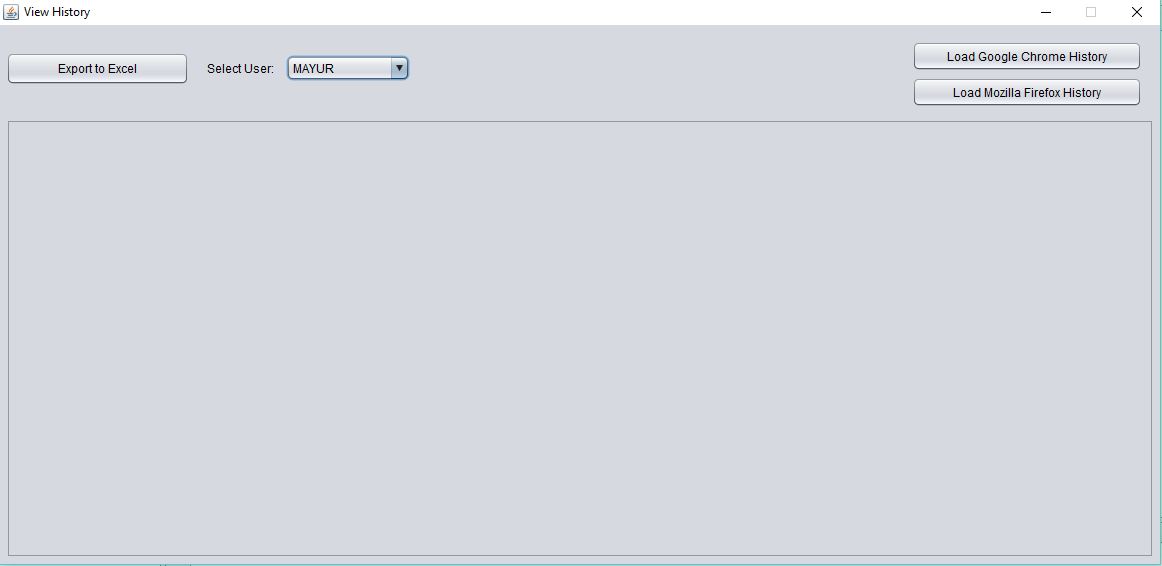
**Graphical User Interface**

**1) Mainpage**

****

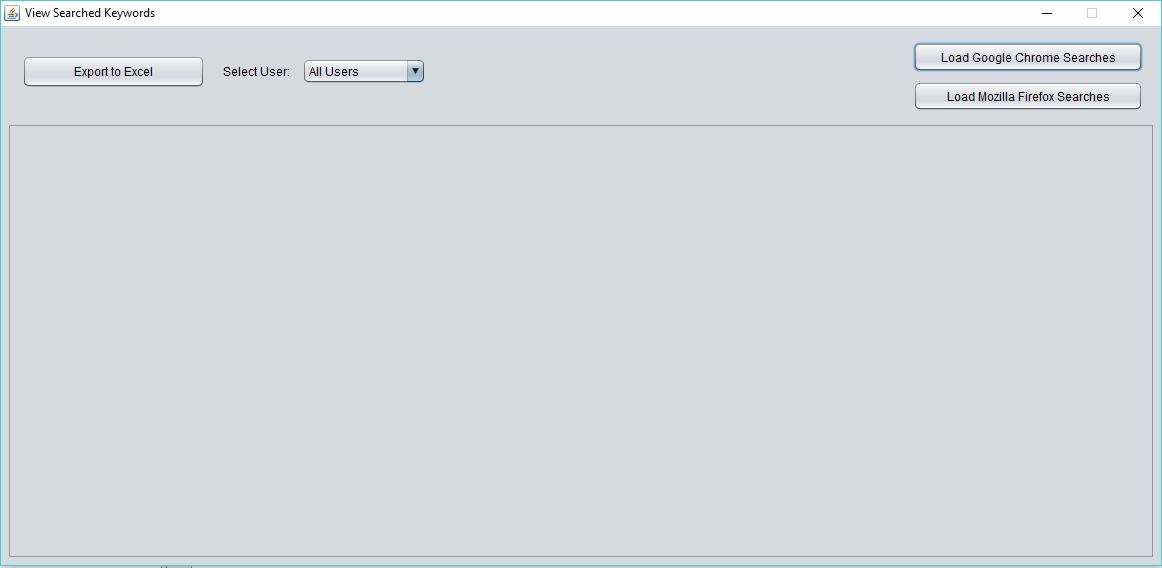
**This is Starting page of our system. This page contains the buttons that redirects to History, Keywords, Downloads, Hosts, Cookies page.**

**2) History**

****

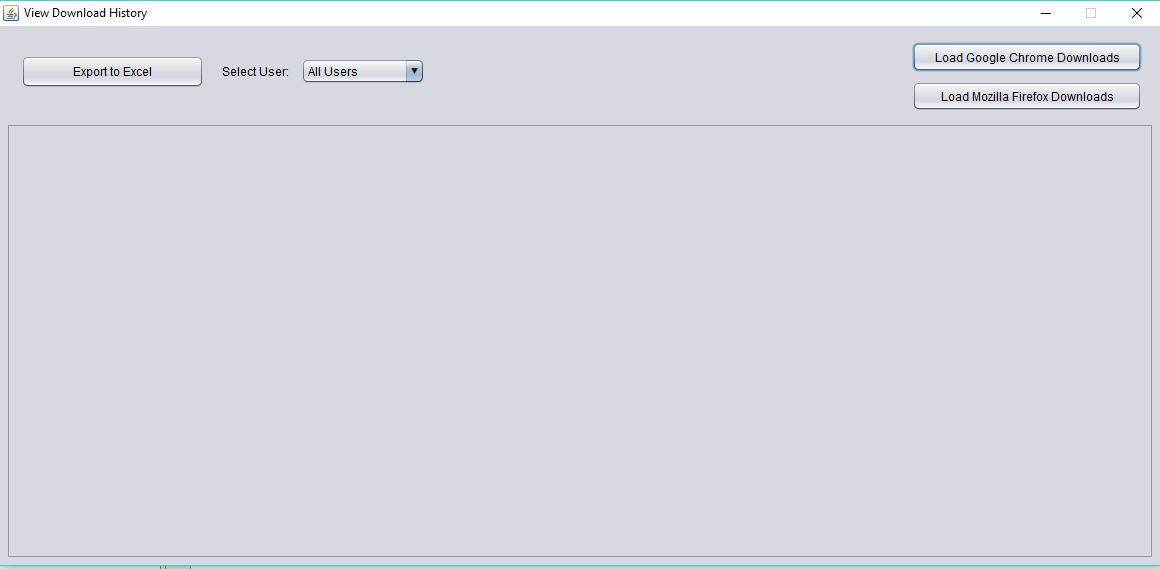
**This page opens after clicking on History button of Mainpage. This contains various browsers names, user drop down list and export to excel button that converts the table form data into excel format data.**

**3) Keywords**

****

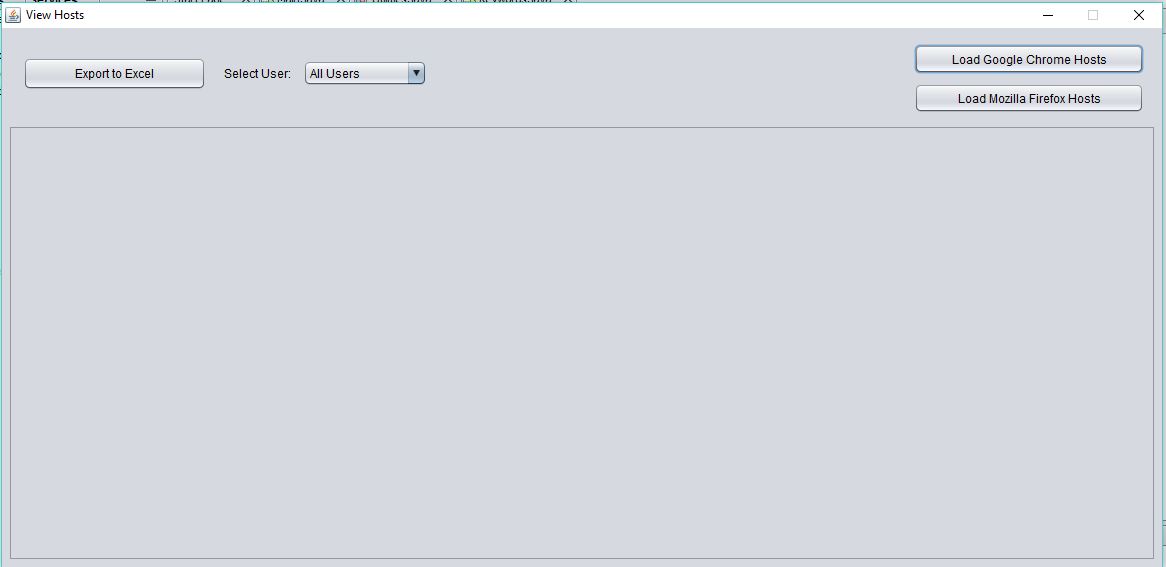
**This page opens after clicking on Keywords button of Mainpage. This contains various browsers names, user drop down list and export to excel button that converts the table form data into excel format data.**

**4)Downloads**

****

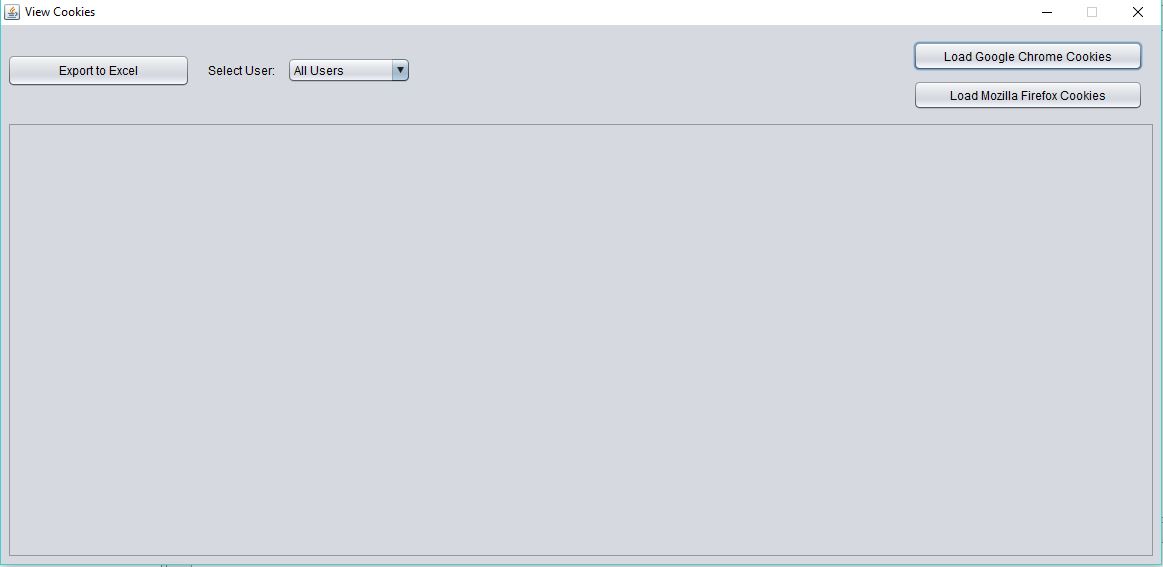
**This page opens after clicking on Downloads button of Mainpage. This contains various browsers names, user drop down list and export to excel button that converts the table form data into excel format data.**

**5)Hosts**

****

**This page opens after clicking on Hosts button of Mainpage. This contains various browsers names, user drop down list and export to excel button that converts the table form data into excel format data.**

**6)Cookies**

****

**This page opens after clicking on cookies button of Mainpage. This contains various browsers names, user drop down list and export to excel button that converts the table form data into excel format data.**