Experiment no. 7

Aim: To implement Session Hijacking attack.

Theory:

Session:
The session refers to certain time period that communication of two computer systems or two parts of a single system takes place. When one logins to a password protected system, the session is used. The session will be valid up to the end of the communication. In some cases, such as in the above described case, the session is user-initiated. There is technology initiated sessions also. Various email clients use the sessions and these are examples for the sessions initiated by the technology. However, many of the active sessions will be hidden from the users. They will not know when a session starts and ends. The session is an important factor in the Internet communications.

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. The most useful method depends on a token that the Web Server sends to the client browser after a successful client authentication. A session token is normally composed of a string of variable width and it could be used in different ways, like in the URL, in the header of the http requisition as a cookie, in other parts of the header of the http request, or yet in the body of the http requisition.

Session hijacking, sometimes also known as cookie hijacking is the exploitation of a valid computer session—sometimes also called a session key—to gain unauthorized access to information or services in a computer system. In particular, it is used to refer to the theft of a magic cookie used to authenticate a user to a remote server. It has particular relevance to web developers, as the HTTP cookies[1] used to maintain a session on many web sites can be easily stolen by an attacker using an intermediary computer or with access to the saved cookies on the victim's computer (see HTTP cookie theft). After successfully stealing appropriate session cookies an adversary might use the Pass the Cookie technique to perform session hijacking. The Session Hijacking attack compromises the session token by stealing or predicting a valid session token to gain unauthorized access to the Web Server.

The session token could be compromised in different ways; the most common are:

- Predictable session token
- Session Sniffing
- Client-side attacks (XSS, malicious JavaScript Codes, Trojans, etc);
- Man-in-the-middle attack
- Man-in-the-browser attack

Session Hijacking Prevention:

As we've seen earlier, the method often used to steal session id is by installing a malicious code on the client website and then the cookie is stealing. The best way to prevent session hijacking is enabling the protection from the client side. It is recommended that taking preventive measures for the session hijacking on the client side. The users should have efficient anti-virus, anti-malware software, and should keep the software up to date. There is a technique that uses engines which fingerprints all requests of a session. In addition to tracking the IP address and SSL session id, the engines also track the http headers. Each change in the header adds penalty points to the session and the session gets terminated as soon as the point exceeds a certain limit. This limit can be configured. This is effective because when intrusion occurs, it will have a different http header order. These are the recommended preventive measures to be taken from both the client and server sides in order to prevent the session hijacking attack.

Example:

Session sniffing: In the example, as 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.

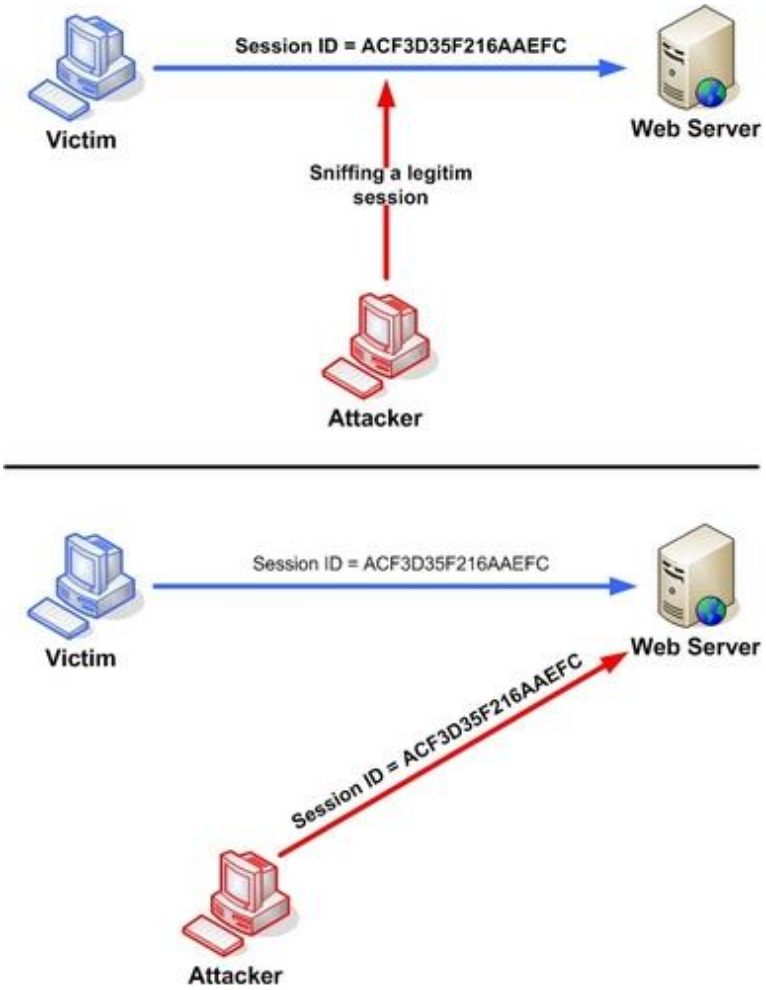


Fig 1: Manipulating the token session executing the session hijacking attack.

PROCEDURE :

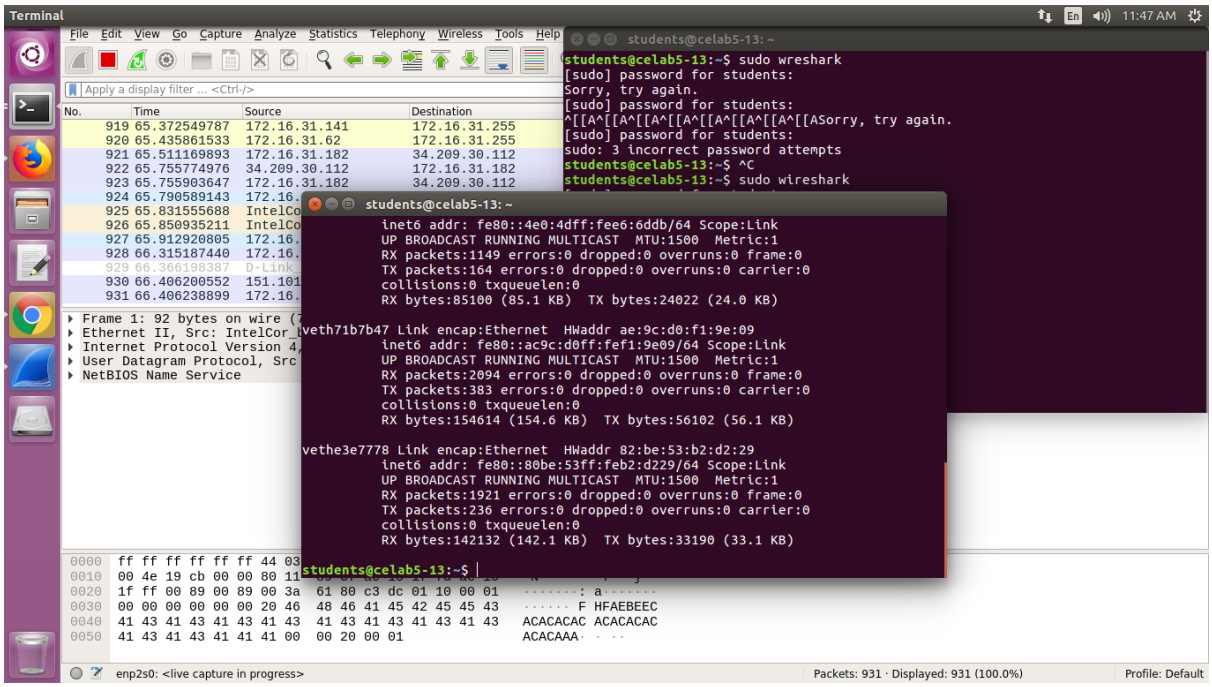
1. 

Fig 2: Terminal Screenshot

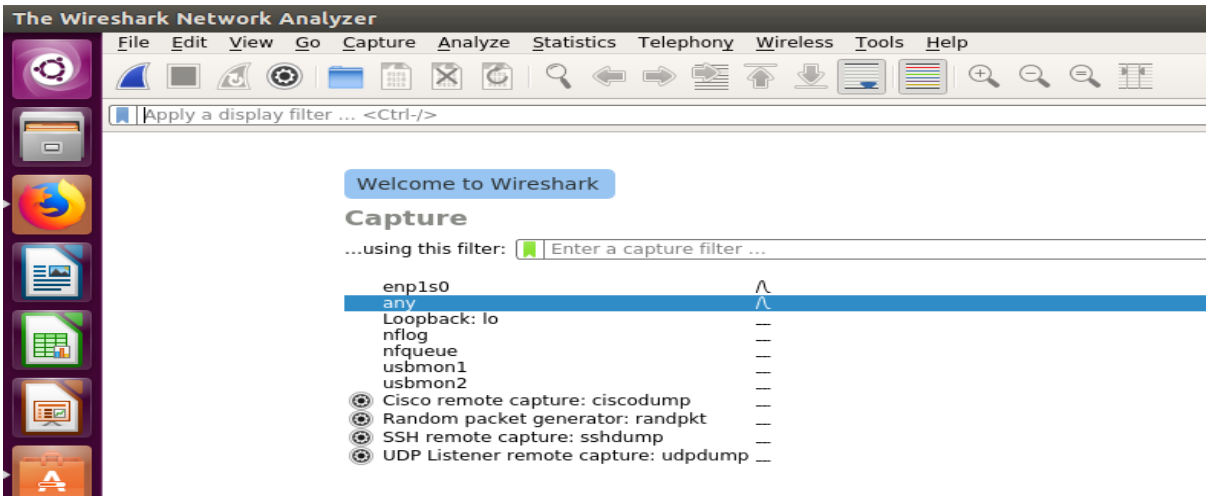
2. 

Fig 3: Wireshark wireless adapter Screenshot

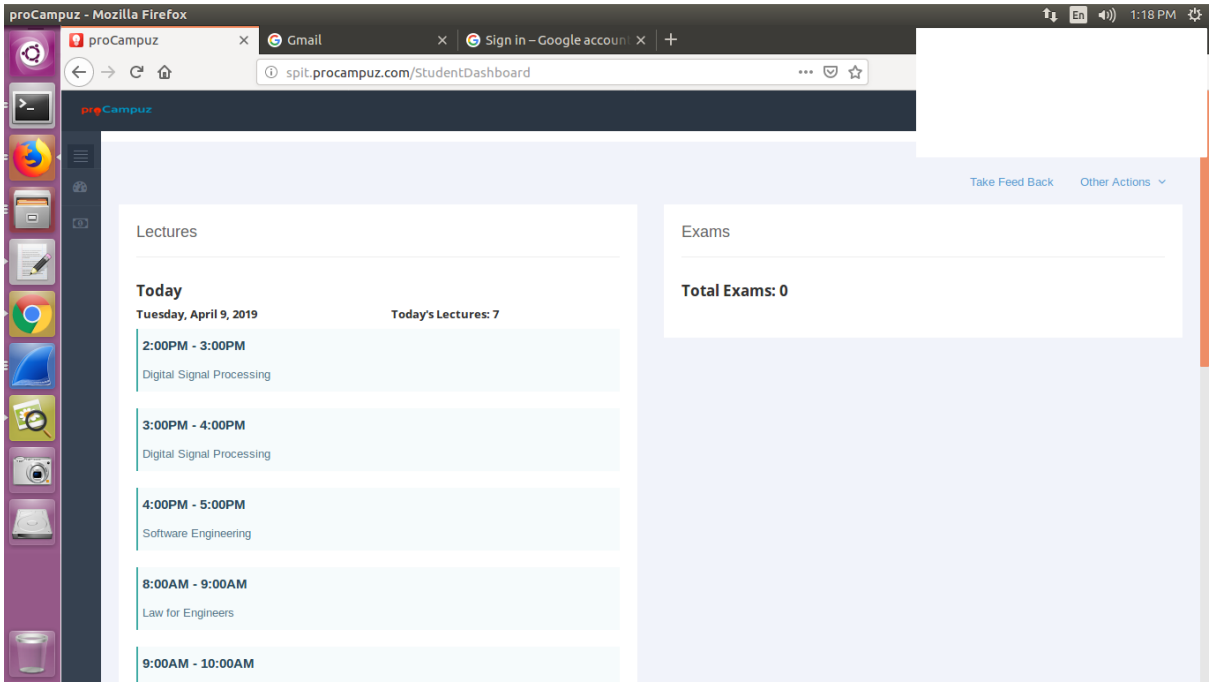
3. 

Fig 4: Procampuz login

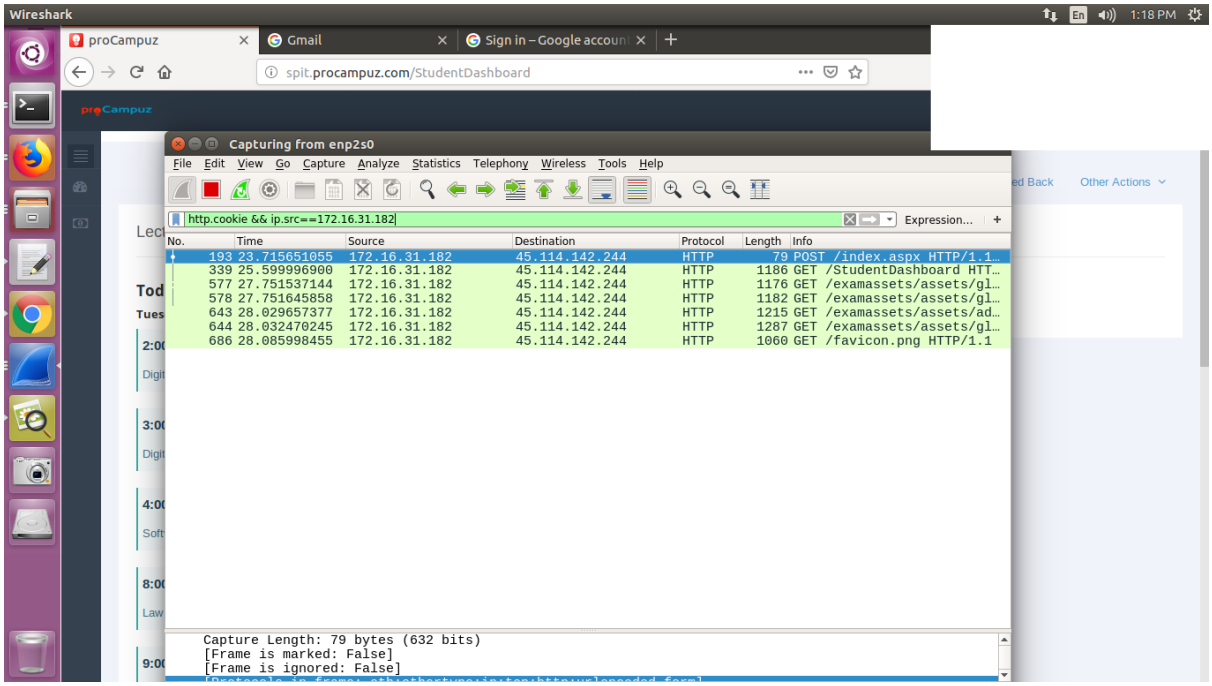
4. 

Fig 5: Wireshark capture

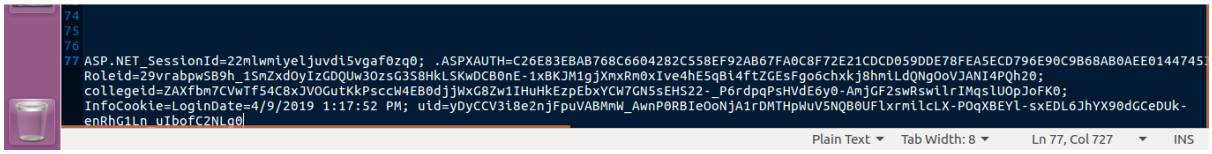
5. 

Fig 6: Cookie details Screenshot

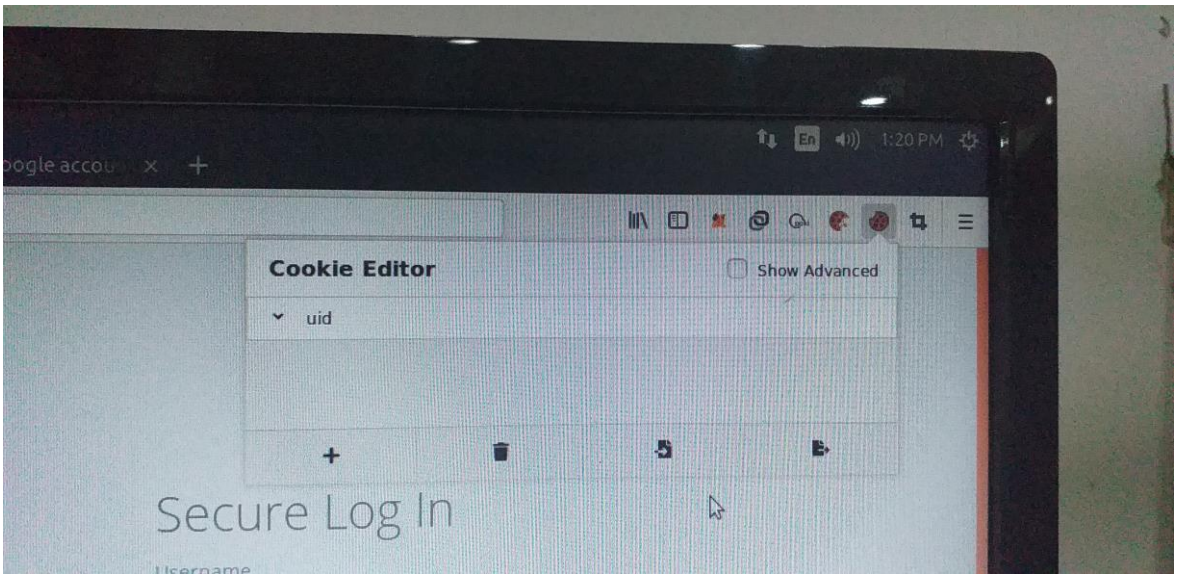
6. 

Fig 7: CookieEditor

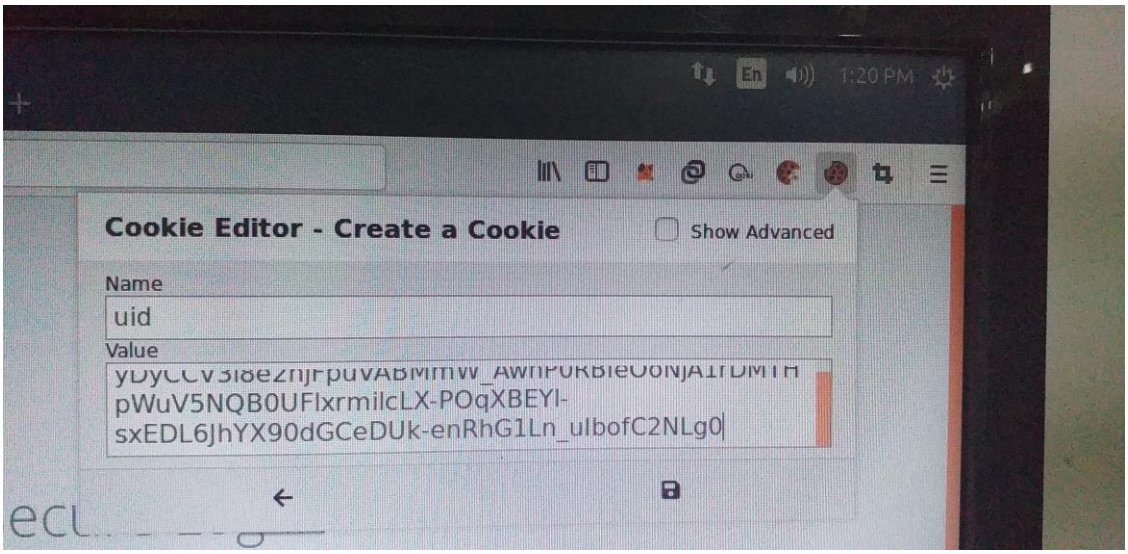
7. 

Fig 8: Editing in Cookie Editor

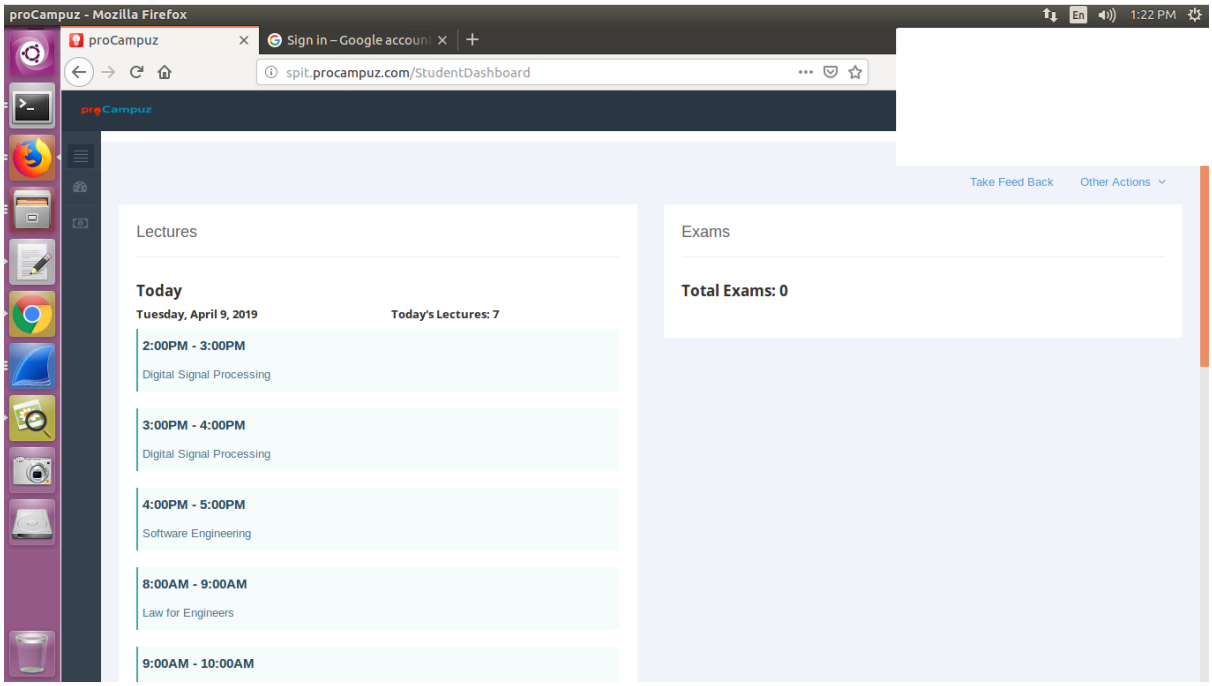
8. 

Fig 9: Gaining access via url

Conclusion:

To implement session hijacking, wireshark is used to analyse the network and to get cookie which than can be edited in browser, and the hacker can acts as victim on the respective website and do malicious activity. To prevent it, make sure the network to which you are connecting is secured and use private incognito mode or vpn service.