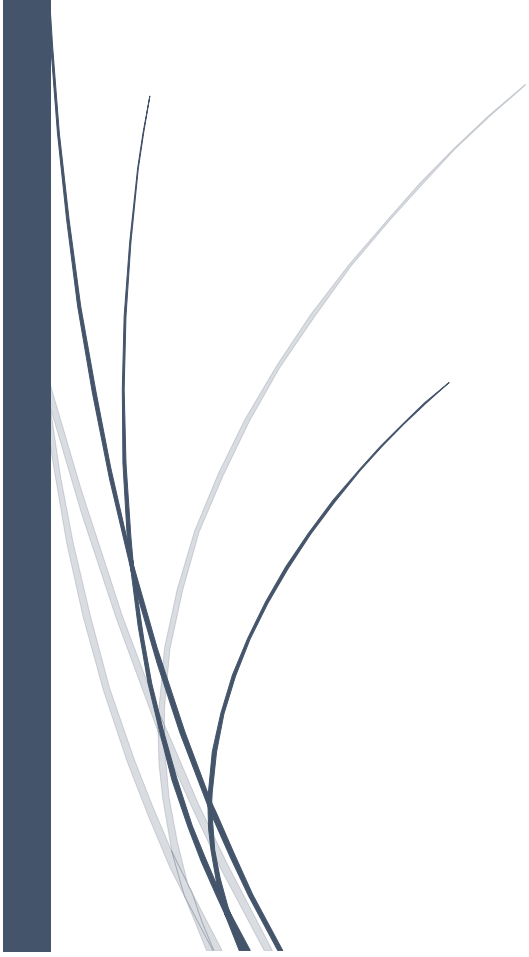
8/4/2021

WEB APPLICATION ATTACKS



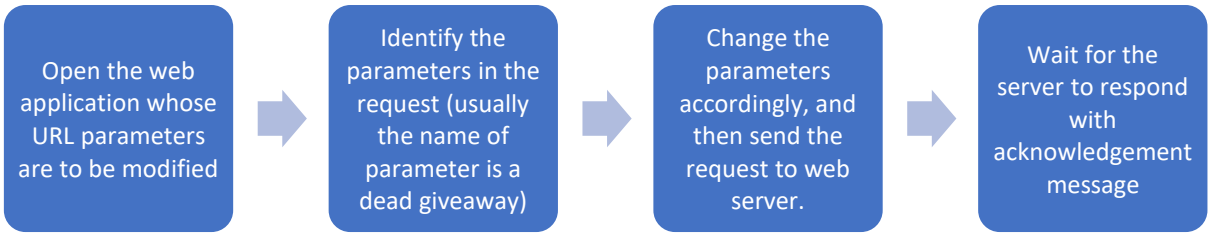
WEB APPLICATION ATTACKS

1 Parameter/Form Tampering

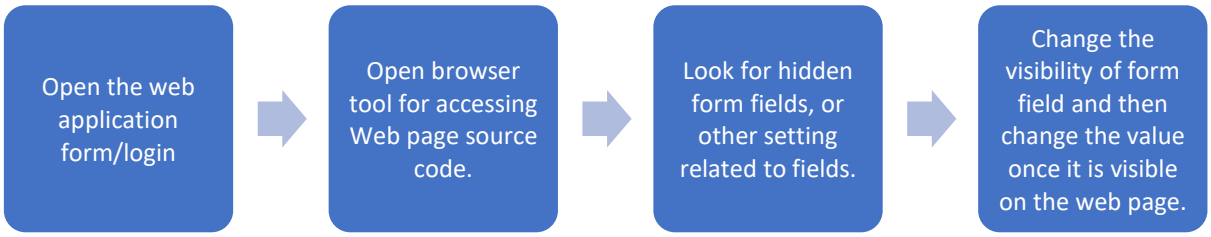
In this the attacker can tamper with the most places where the server requests an input from the user or which are accessible to the users, such as Form fields, URLs, HTTP Headers, etc. Such an attack can also led to cross site scripting or SQL injection also. If the attacker modifies the url parameters for any request they can extract data or make unauthorized changes to the database, or if they try to enter SQL queries in forms then that can extract data from the DB or in some cases bypass the login page with just queries.

❖ Flow Chart

❖ Parameter Tampering in Web Application URL :



❖ Form Tampering:



❖ Tools required:

This attack can be implemented without any other tools than the web browser. Although if required we can utilize Web server requesting tools such as Postman.

❖ Effectiveness of attack:

These attacks are less effective nowadays as most of the Web Applications are hiding the sensitive information from the URLs, which makes it difficult to change them that easily. Even with forms, most web applications have protection in backend to filter out SQL queries and actual data as well as change in form fields is detectable as well, which reduces the chances of form tampering being an effective method. Parameter/Form Tampering can still be utilized in some web applications which are not that secure yet, but it will be almost useless when trying out on major web applications.

----------XXXXX----------

2 Cookie Tampering

Cookies are small files stored within the browser by the server upon connection. These files contain information which help in identifying the user/system from all others, so that content specific to that user can be displayed by the server. Upon reconnection the server will read the cookie and then select what information to present based on the cookie.

They can also contain information about the user's session, relevant history or other information which can help develop targeted ads.

❖ Basic types:

Session -

- Used only during website usage
- Stays in RAM only
- Are removed after session ends.

Persistent -

- Remains on ROM
- Can be used to track history, and useful for auto login as well
- Some tend to have an expiration date.

❖ Flow Chart

Client-Side Cookie Tampering:

❖ Tools required:

This method can be implemented just using the web browser, as most current web browsers have the ability to modify cookies along with other elements.

❖ Effectiveness of attack:

This method of cookie tampering is not very effective. As most web applications have ceased storing sensitive data in cookies all together. There still might be some unsecure web applications where it can be utilized but not many. There are other cookie related attacks that are much more effective and useful, Session side jacking, session fixation, etc. These can acquire access to users account without even needing the password as using the appropriate session cookie will make them appear as the legitimate user for the web application.

----------XXXXX----------

3 Unvalidated Input and File Injection attacks

❖ What is Unvalidated Input?

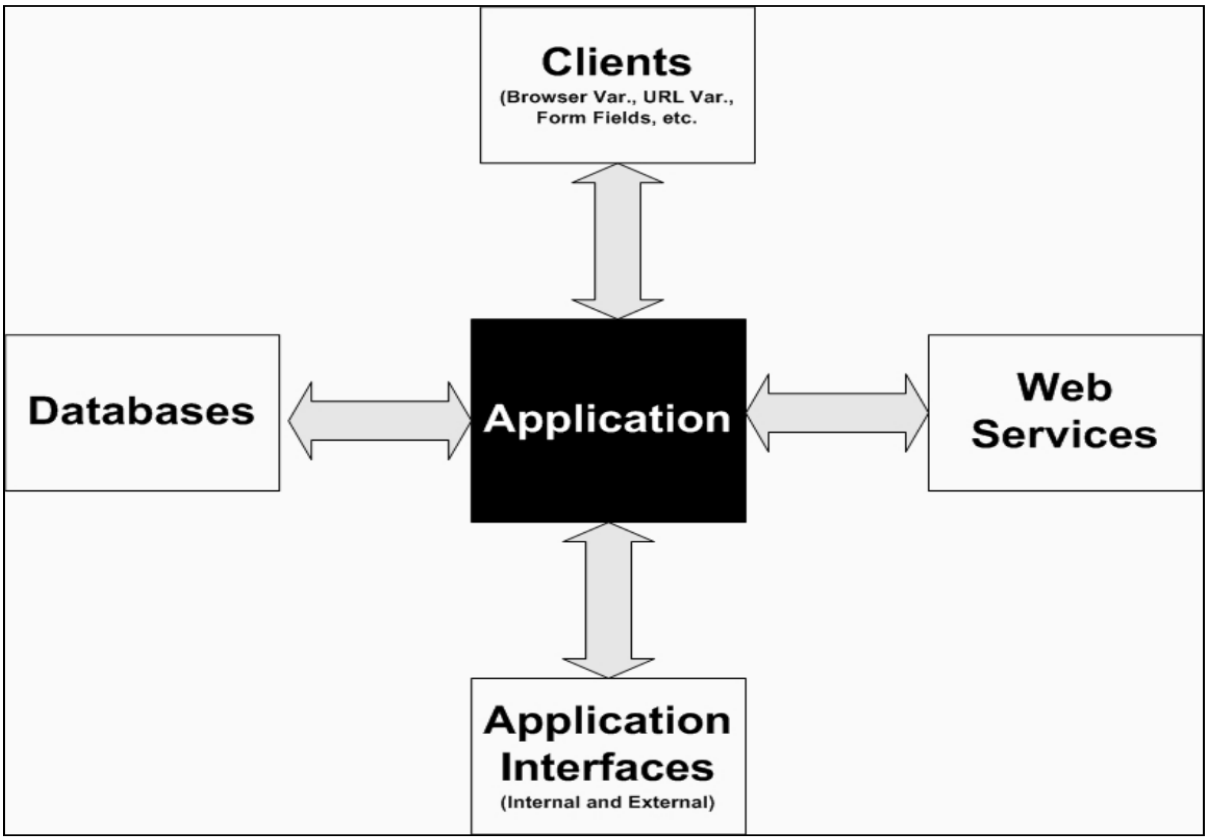
- As a rule, all information received by your program should be checked to ensure that the information is reasonable.
- An image of 200 by 300 pixels but cannot include an image of 200 by -1 pixels, can reasonably be included in the graphics file, for example. There is no way, however, that a file claims to contain such an image. A naive software program that tries to read such a file would try to allocate an incorrect buffer that could lead to a heap overflow attack or some other problem. You must therefore carefully monitor your input data. This process is usually referred to as input validation or health checks.
- Any input from an untrustworthy source received by your program is a potential attack target.

- Whenever the input from an uncontrolled source is accepted by your program, a user has the potential to pass data that does not meet your expectations.
- If you don't validate the input, this could lead to issues such as program crashes that an attacker can execute his own code.
- An attacker can take advantage of invalidated input in several ways, including:
    - Buffer overflows
    - Format string vulnerabilities
    - URL commands
    - Code insertions
    - Social engineering
- Many Apple security updates fix input vulnerabilities such as some hackers used to "jailbreak" iPhones. Input vulnerabilities are common and often easy to use, but usually easy to correct.

❖ What is file injection attacks?

- A file inclusion vulnerability is a type of online vulnerability that typically affects web applications that use a scripting runtime. This problem arises when an application constructs a route to executable code using an attacker-controlled variable in such a way that the attacker has control over which file is executed at runtime.
- A file inclusion vulnerability differs from a generic directory traversal attack in that it subverts how a programme loads code for execution, whereas directory traversal is a method of acquiring unauthorized file system access.
- If a file inclusion vulnerability is successfully exploited, remote code execution on the web server that hosts the affected web application will occur.
- An attacker can construct a web shell on the web server via remote code execution, which can be exploited to deface a website.

❖ Flowchart of the attack



❖ Potential vulnerabilities

- **Improper HTML display:**
  - An unprofessional portrayal of your site would be the most low-impact display issue. Tailed input can make the display unusable at the other end of the spectrum.
  - Moreover, attackers can force errors to gain an idea of how diligent your developers tighten up their code in the application output. If the picture is easy to hack, it is a good bet that other application components can be crushed just as easily.

- **By-passed client-side validations:**
  - Validation on the client side isn't validation.
  - An attacker can deactivate script execution on her computer, enter harmful incorrect form data, and submit the form with ease.
  - Server crashes and the execution of rogue commands are just two of the conceivable results if there is no validation on the server side of the transaction.

- **Cross-site Scripting:**

- Unvalidated text fields may contain HTML tags such as <script> and <object>, which execute code unexpectedly.

- **Generation of informal error messages:**
  - An attacker can gain information about your system's application and operating system version and patch level by forcing specific types of error messages.
  - Footprinting is a popular application of this information in the early phase of an attack.

- **Buffer overflow:**
  - Buffer overflows can be used to crash a system or execute malicious code.

- **SQL injection:**
  - Company databases are vulnerable to unauthorized data change, deletion, or addition in this form of attack. It's created by an attacker inserting SQL code into a form field or changing query strings directly.

❖ List of tools:
- OWASP WebScarab

❖ Types of file inclusions

- **Remote file inclusion (RFI)**
  - When a web application downloads and runs a remote file, it is known as remote file inclusion (RFI). These remote files are often accessed by passing an HTTP or FTP URI to the web application as a user-supplied parameter.
- **Local file inclusion (LFI)**
  - Local file inclusion (LFI) is comparable to a remote file inclusion vulnerability in that only local file, i.e., files on the current server, can be included for execution instead of remote files. By providing a file with attacker-controlled data, such as the web server's access logs, this problem can still lead to remote code execution.

❖ List of configurations for file inclusions

- **PHP**
  - The usage of unvalidated user input with a filesystem function that contains a file for execution is the main cause in PHP. Include and must statements be the most notable.
  - Most of the flaws can be traced to inexperienced programmers who are unfamiliar with all the PHP programming language's features.

- **JavaServer Pages**

- JSP (JavaServer Pages) is a scripting language that allows you to include files that will be executed at runtime.

- **Server Side Includes (SSI)**
  - A Server Side Include is extremely rare, as it isn't usually enabled on a standard web server. On a vulnerable web server, a server-side inclusion can be leveraged to acquire remote code execution.

❖ My opinion on the effectiveness of the attack

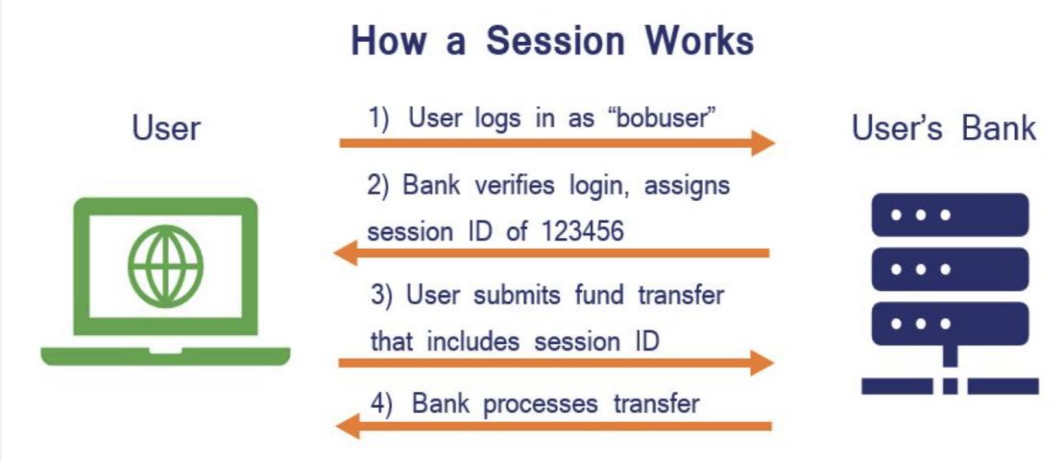
- In my opinion this attack does not impactful and couldn't find relevant information about it
- However, this attack is as unique as your development environment, your strategy to validating input will be.
- The development of appropriate policies, standards, guidelines, procedures, and developer knowledge is the one necessity that applies to all contexts.
- To protect against this flaw, all user input must be validated before being used.

----------XXXXX----------

4 Session hijacking

➢ What is session?

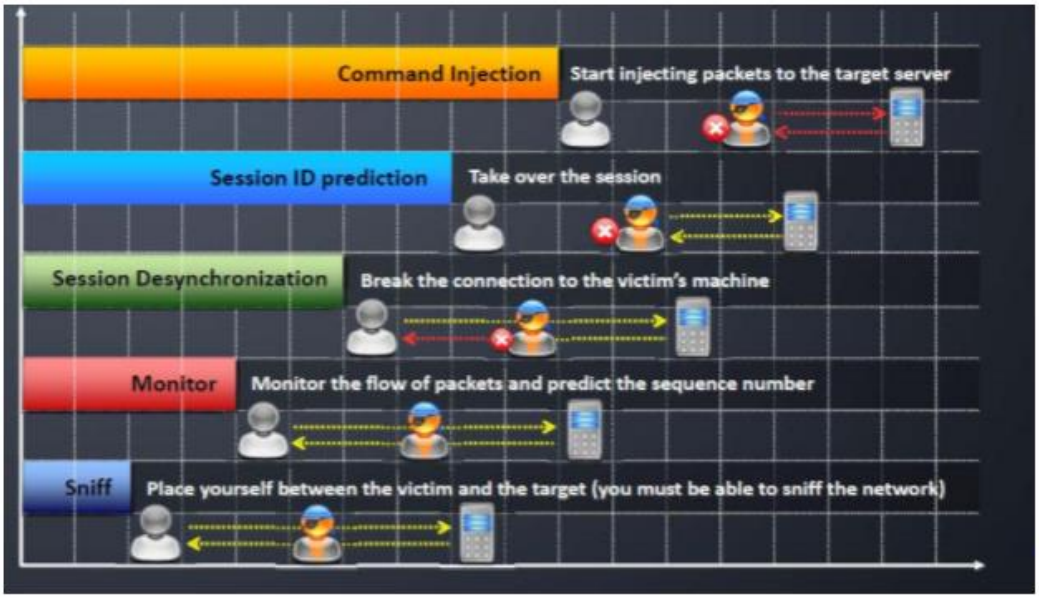
- HTTP is essentially stateless, which implies that each request is processed separately and without regard for prior requests. In practice, this means you'd have to input your username and password each time you visited a new page. As a result, rather than requiring users to re-authenticate between each click in a web application, the developers needed to devise a technique to track the state of several connections from the same user.



➢ What is session hijacking?

- A session hijacking attack occurs when an attacker takes control of a user's session. When you log into a service, such as your banking application, a session begins and ends when you log out. The attack is also known as cookie hijacking or cookie side-jacking since it relies on the attacker's knowledge of your session cookie.
- Although any computer session can be hijacked, browser sessions and online applications are the most usual targets.
- When you log into an online application, a temporary session cookie is often placed in your browser by the server to remember that the application is signed up. HTTP is a stateless protocol, and the most common means for the server to identify your browser or current session is through session cookies, which are added to every HTTP header.
- An attacker needs to know the victim's session ID to accomplish session hijacking (session key). This can be gained via stealing the session cookie or convincing the user to click on a malicious link with a session ID that has already been created. On the server, the attacker can hijack the session by using the same session ID for its own browser session after the user has been authenticated in both cases.
- The server is subsequently misled into thinking the attacker's connection is a valid session for the original user.

➢ Flowchart of the attack



- **Active Session sniffing**
  - The attacker then locates an active session between the target and another machine and enters it.
  - He catches the traffic with a sniffer like Wireshark and attempts to acquire information about the session.

- **Monitor**
  - He then scans the stream for vulnerable protocols such as HTTP, telnet, and looks for any legitimate authentication messages.

- **Session ID retrieval**
  - Using known information, the attacker attempts to predict the session id. Sequence number prediction is the next phase in the session hijacking process after a target has been picked.
  - Because failing to anticipate the proper sequence number will result in the server issuing reset packets and terminating the connection attempt, sequence number prediction is crucial.
  - If the attacker guesses the sequence numbers incorrectly several times, the attack is more likely to be detected.

- **Stealing**
  - Active assaults are used in application-level hijacking to steal the session ID. To steal the session id, man in the middle attacks, cross-site scripting, and sniffing are utilized.
  - While skilled attackers can perform number guessing by hand, software tools are available to automate the process.

- **Taking target offline**

- One of the targets must be muted once a session has been chosen and sequence numbers predicted.
- This is usually accomplished by a denial-of-service attack.
- The attacker must keep the client computer offline for the duration of the attack, or the client computer will start sending data across the network, prompting the workstation and server to try to synchronize their connections repeatedly, resulting in an ACK storm.

- **Taking over the session and maintaining the connection**
  - Taking control of the communication session between the workstation and the server is the final part of the session hijack attack.
  - To evade detection, the attacker will impersonate their client IP address and include a sequence number that was previously predicted.
  - The attacker has successfully assaulted the communication session if the server accepts this information.

➢ Common strategies of session hijacking

Fixation Attack:

- Session fixation attacks make the most of a flaw associate exceedingly in a very} system that permits somebody to fixate (or realize or set) the session ID of another user.
- this type of attack focuses on phishing tries to induce websites to simply accept session IDs from URLs.

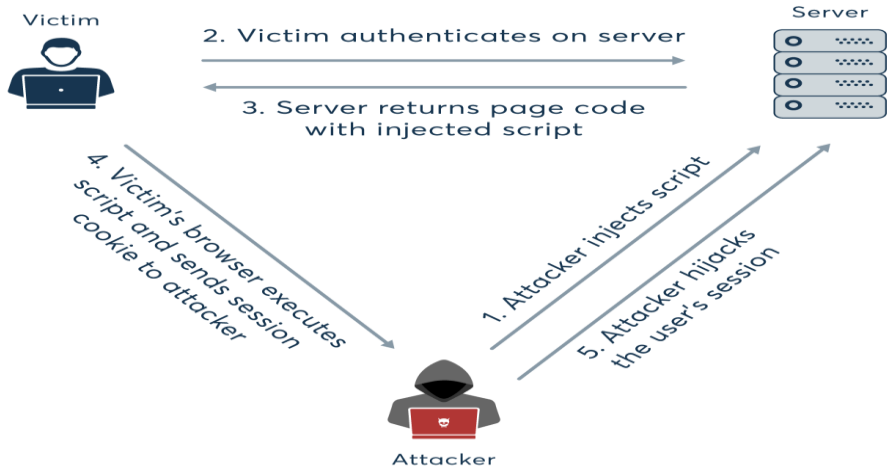
➢ Session sniffing:

- once a hacker uses a packet individual like Wireshark to capture and log packets as they flow over a network connection, this is often called session sniffing. Session cookies are a part of this communication, and an attacker will realize and steal them victimization session sniffing.

- Session hijacking can happen if SSL/TLS isn't utilized on the rest of the site. Hackers are going to be able to monitor the traffic of everybody else on the network, together with session cookies, via packet sniffing.

- this type of session hijacking attack is especially vulnerable on public Wi-Fi networks. because of there's no user authentication for the network, a hacker can monitor abundant of the network traffic just by work on and employing a packet sniffer. A hacker may additionally construct their own access purpose and use man-in-the-middle attacks to gain session IDs and commit session hijacking.

➢ Cross-site scripting (XSS):



- A cross-site scripting (XSS) attack deceives the user' machine into running malicious code, although it seems to come back from a trustworthy source. once the script is launched, the hacker will take the cookie.
- Client-side scripts (typically JavaScript) are injected into webpages via server or application vulnerabilities, inflicting the browser to execute the code when the page is loaded. Malicious scripts can get your session ID if the server doesn't set the HttpOnly property in session cookies.

Malware:

- Session hijacking may be caused by malware and different dangerous third-party apps. Hackers programmed the malware to smell packets and hunt for session cookies.
- It then steals it and sends it to the assaulter once it finds one. The malware is doing a session sniffing attack on the user.
- Gaining access to the user' machine, whether or not via malware or by physically connecting to that domestically or remotely, is another additional direct technique of collection session IDs.
- The attacker can then visit the browser' temporary native storage folder, or "cookie jar," and choose Associate in cookie they want.

Brute force:

- Finally, an assaulter will try and guess a user' active session key, that is just potential if the applying utilizes short or sure session identifiers.
- sequent keys were once a typical flaw; however, session IDs are currently prolonged and generated randomly in recent applications and protocol versions.
- to stop brute force assaults, the key generation method should offer sudden numbers with enough entropy to render guess attacks difficult.

➢ List of session hijacking tools:

- **Burp Suite**
  - **https://portswigger.net**
- **Ettercap**
  - http://ettercap.github.io
- **OWASP ZAP**
  - **https://www.owasp.org**
- **BetterCAP**
  - **https://www.bettercap.org**
- **netool toolkit**
  - **https://sourceforge.net**
- **WebSploit Framework**
  - **https://sourceforge.net**
- **sslstrip**
  - **https://pypi.python.org**
- **JHijack**
  - **https://github.com/yehgdotnet/JHijack**
- **Cookie Cadger**
  - **https://www.cookiecadger.com**
- **Firesheep**
  - **http://codebutler.github.io/firesheep/**

➢ My opinion on the effectiveness of the attack

- Session hijacking, in my opinion, encompasses a vital impact because of it's outlined as taking control of a lively TCP/IP communication session while not the user' permission. once properly executed, attackers war the identity of the compromised user Associate have access to a similar resource because the compromised user.
- Moreover, session hijacking encompasses a kind of consequences, together with identity thievery, info theft, and the theft of sensitive data.
- Session hijacking permits cyber attackers to completely lead of a machine, each at the network and application level. Multiple applications are place in danger when hackers get access to an SSO (single sign on).
- SSO maintains credentials for all applications, together with those who handle sensitive personal data, in cookies.
- On the opposite hand, security has been tightened au fait social networking platforms like Facebook, Google, Associate in Nursingd Instagram, with layers of authentication added. once trying to enter an approved account, hackers, for example, incur the danger of frequently submitting wrong information. As a result, an application may be locked, and the account owner will be alerted.
- To summarize, session hijacking may be a probably dangerous situation, however preventing measures have always been advantage.

----------XXXXX----------

5 SQL injection attack:

5.1 What is SQL injection?

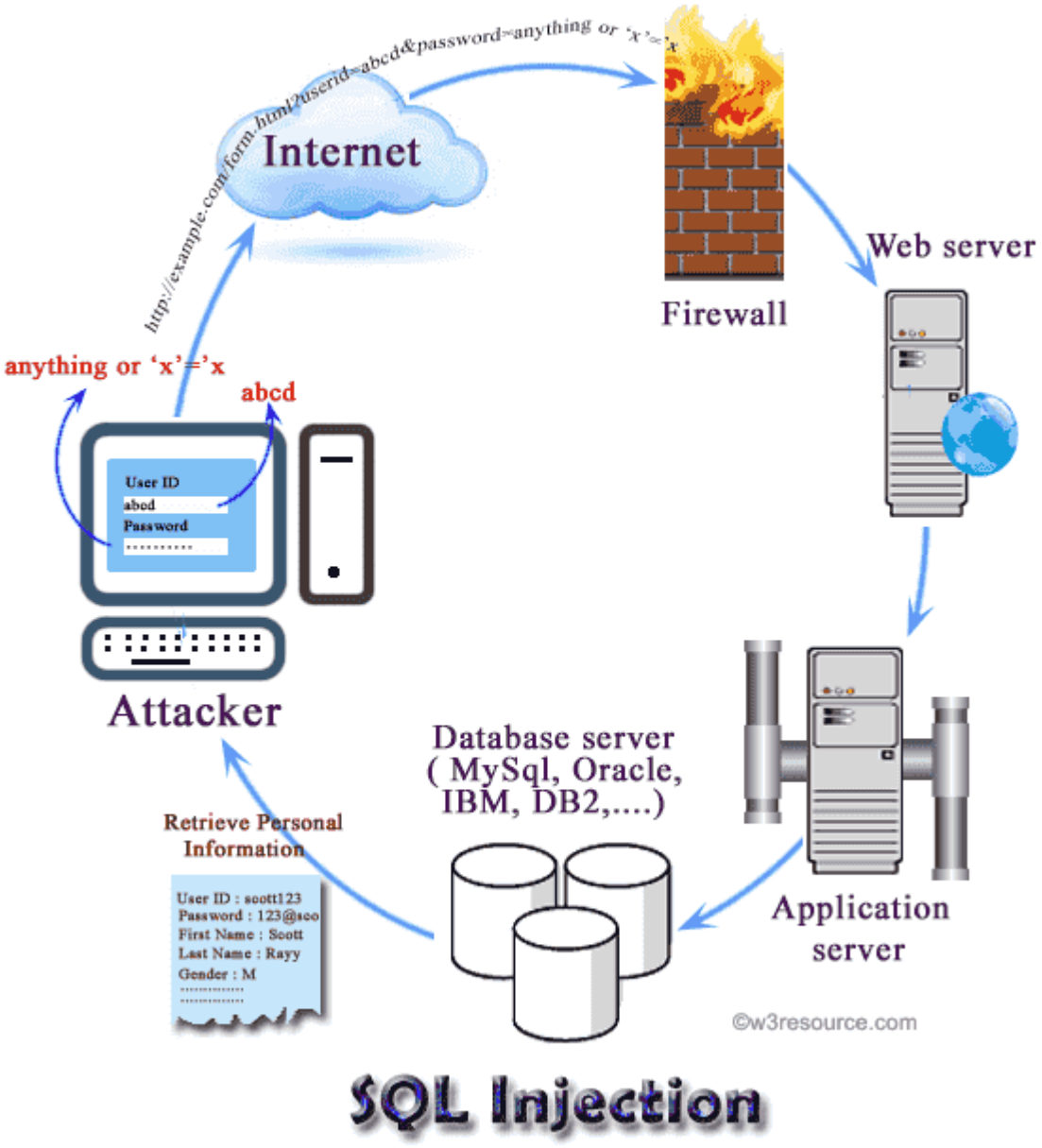
- SQL injection is a vulnerability in web security that allows an attacker to intercede with queries made by the user to the database.
- It allows the attacker to view the data from the database which is not usually allowed to be seen and retrieved.
- This data can be any user data or application data.
- What the attacker can do is edit, modify or delete this data, which can cause changes to the application or database content or behavior.
- An attacker can even escalate the attack to compromise the database server or can perform a denial-of-service attack.
- A successful attack results in unauthorized access to sensitive data, including passwords, card details, or personal user information.
- This is very common with PHP and ASP apps because of older interfaces.

5.2 Types of SQL injection attacks are:

- Retrieving Hidden Data
- Union attacks
- Examining the database
- Blind SQL injection
- Subverting Application Logic

5.3 Flowchart of the attack

1. This attack can bypass the firewall and impact the secured system.



2. First, the attacker uses the poorly filtered or incorrect escaped characters embedded in SQL statements into parsing variable data from user input.
3. Next, he injects arbitrary data generally a DB query, which is executed by the database through a web application.
4. Through this the attacker obtains unauthorized access to a database.
5. The attacker can also create, read, update or delete the data stored in the database server.

6. All databases such as ORACLE, MySQL, MSSQL Server, MS Access are currently vulnerable to SQL injection attacks.

5.4 Impact of SQL injection

- Steal Credentials
- Access Databases
- Alter Data
- Delete Data
- Access Networks

5.5 Tools Required for SQL injection Attack

Several tools are available to perform SQL injection attacks, a few of them are listed below:

- SQLMap
- jSQL Injection
- Whitewidow
- DSSS
- Leviathan
- BBQSQL
- Explo
- NoSQLMap

Another thing, required for the attack to materialize is entering unintended data to the program from an untrusted source.

5.6 My opinion on the effectiveness of the attack

I believe SQL injection attack has vast effectiveness because a successful attack can release sensitive data including, user data, credit card data, result in deletion of tables or an attacker can administratively right to the database and these all are highly calamitous to the company or organization.

- Another thing is SQL attacks are pervasive means they can evade firewalls and other perimeter defenses.
- Moreover, the organizations are not familiar with the techniques used by the attackers.
- The measures required to be taken to prevent SQL injection are also lacking.
- I believe, the SQL injection threat should be taken seriously because the recent attacks for the past 12 months I have studied the majority of them evaded perimeter defenses.

So, to conclude the SQL attacks are highly devastating and effective.

----------XXXXX----------

6 Directory Traversal Attack

6.1 What is Directory Traversal Attack

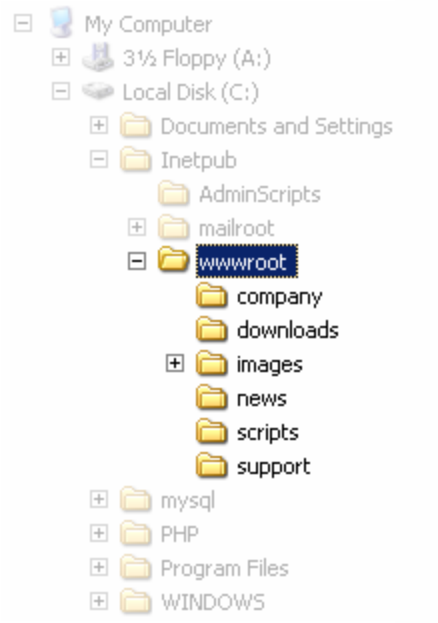
The main focus of this HTTP attack is to gain access to files and directories that are stored outside the web-root folder. The attack is also known as path traversal attack.

Access to access arbitrary files and directories/folders stored on the file system which include application source code or configuration is made possible by manipulation of variables referencing files with “dot-dot-slash (../)” and by using absolute file paths.

The Attack is also known as directory climbing and backtracking.

6.2 Tools and Configuration Required for the attack

- To perform the attack, the attacker need is a web browser and knowledge of where he can find any default files and directories on the system.
- Two types of web server security levels are:
  1. ACLs
  2. Root Directory



- ACL is used to authorize which groups a user can access, modify or delete files on the server.
- The root directory prevents the users from accessing folders and files such as C:\WINDOWS/system32/win.ini.
- An attacker can make use of the root directory access vulnerability to step out of the root directory and access other folders present on the file system.
- This provides an opportunity to the attacker to view other restricted files
- This further can aid the attacker with other useful information to compromise the system.
- Type of directory traversal attacks are:
    - Via web application code
    - Via web server

How to prevent Directory Traversal Attacks?

- Install latest web server software, install all latest patches.
- Filter user input. Remove anything other than good and required input data.
- Remove metacharacters from user input.
- Use a web vulnerability Scanner like Acunetix, to get detailed reports, and use these reports to remove the vulnerability.
- Carefully write the code of the web application or website.

How Effective is Directory Traversal Attack?

In my opinion, a directory traversal attack once successful is very damaging to the organization.

- The attack can be easily executed if there are vulnerabilities in the code and application.
- The attacker can download server configuration files, containing important and sensitive information and use this to explore more vulnerabilities in the server.
- The attacker later can gain access to confidential information or even gain full control of the webserver.

So, for me, this attack is very effective once successful.

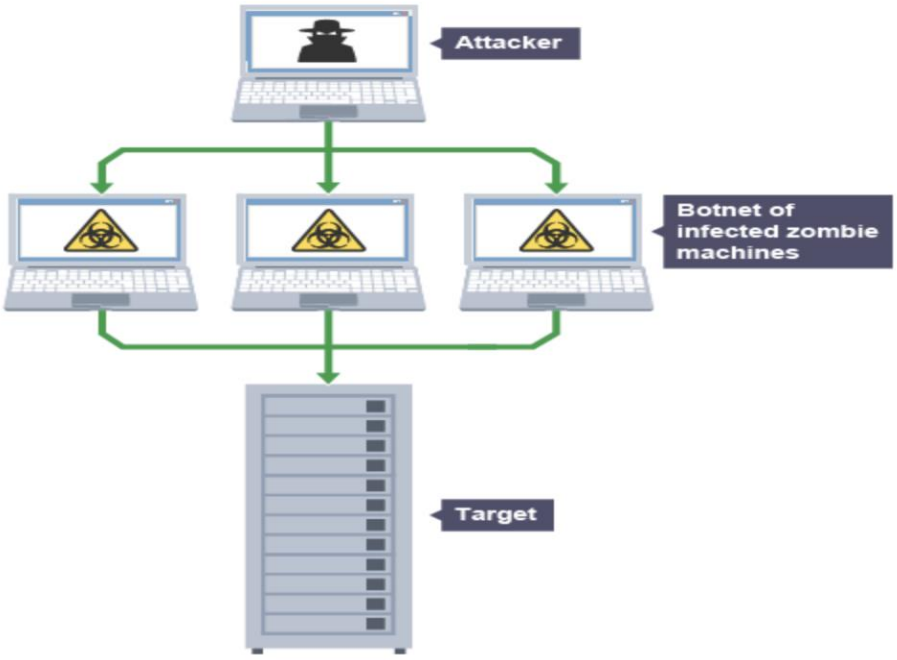
----------XXXXX----------

7 Denial-of-service Attack

7.1 What is denial-of-service attack?

- A Service Denial attack only uses a small number (perhaps just one) of attacking systems for overloading the target. One notable exception here could be industrial control systems with low bug tolerance or connected via easily saturated bandwidth links. A Denial-of-Service (DoS) attack occurs if legitimate users are prevented by a malicious cyber threat actor from accessing information systems, devices, or other network resources.
- Email, websites, online accounts (e.g., banking), and other services that rely on the compromised machine or network may be disrupted.
- The targeted host or network is flooded by a denial-of-service attack until it is unable or simply unsuccessful to respond, preventing authentic users access. DoS attacks can cost a company both time and money while its resources and services are unavailable.
- When a website is attacked by DoS, your perspective will rely on the obvious effect. The site simply stops displaying content appears to the user on average.
- For companies, this may have meant that they have stopped reacting online systems. Sense data may not be recovered, or critical processes controlled due to the symptoms of a DoS attack on industrial systems.
- DoS attacks may last for more than one site or system at a time. When an attack is carried out from multiple computers (or vectors), it becomes a "distributed Denial of Service," known as "DDoS," rather than just one.

7.2 Flowchart of the attack?



7.3 How dos attack works?

These are the five common types of attacks

- **Ping of Death**
  - The ping command is commonly used to check if a network resource is available. It communicates with the network resource by sending tiny data packets. The death ping is used by transmitting data packets that are greater than the TCP/maximum IP limit (65,536 bytes).
  - TCP/IP fragmentation splits up packets and sends them to the server in small bits. The server may freeze, reboot, or crash because of the delivered data packages being greater than the server's capacity.
- **Smurf**
  - This sort of attack targets an Internet Broadcast Address with a huge amount of Internet Control Message Protocol (ICMP) ping traffic. The targeted victim's IP address is spoofing the reply to IP address.
  - Instead of the IP used for the pings, all responses are provided to the victim. A smurf attack amplifies a single ping 255 times because a single Internet Broadcast Address can serve a maximum of 255 hosts. This has the effect of slowing down the network to the point that it is no longer usable.
- **Buffer overflow**
  - A buffer is a temporary storage space in RAM used to keep data while the CPU manipulates it before writing it back to the disc. Buffers have a maximum size.
  - This form of attack overflows the buffer with data it can't handle. This causes the buffer to overflow, causing the data it carries to be corrupted. Sending emails with 256-character file names is an example of a buffer overflow.
- **Teardrop**
  - Larger data packets are used in this type of attack. TCP/IP decomposes them into fragments, which the receiving host assembles.

- The attacker tampers with the packets as they are sent, causing them to overlap. As the targeted victim attempts to reassemble the packets, it may crash.

- **SYN attack**
  - Synchronize is abbreviated as SYN. To establish connection using TCP, this form of attack takes use of the three-way handshake. The victim of a SYN attack is bombarded with incomplete SYN messages.
  - As a result, the victim machine allocates RAM resources that are never used, and legitimate users are denied access.

7.4 List of DoS attack tools:

- **Namesy**: Random packets can be generated with this utility. It's compatible with Windows.
  - http://packetstormsecurity.com/files/25599/nemesy13.zip.html
- **Land and LaTierra:** This programme can be used to spoof IP addresses and open TCP connections.
- **Blast:**
  - http://www.opencomm.co.uk/products/blast/features.php
- **Panther:** This program can be used to send UDP packets into a victim's network.
- **Botnets:** A distributed denial of service attack can be carried out using a large number of infected computers on the Internet.

7.5 My opinion on the effectiveness of the attack

- In my view, attacks are the most dangerous one because they can harm someone financially or personally.
- Moreover, by flooding servers, websites, and web services with an immense number of requests, such attacks cause them to malfunction. When resources aren't built to handle large loads, they stop working, making them unavailable to users. These attacks also take advantage of flaws in the network protocol and application layers.
- To defend against denial-of-service attacks, security patches for operating systems, router configuration, firewalls, and intrusion detection systems can be utilized.
- Inquire with your ISP to see if there is an outage on their end, or if their network is being attacked and you are an unwitting victim. They might be able to give you some advice on how to proceed.

- To summarize, denial-of-service attack may be a probably dangerous situation, however preventing measures always been advantageous.

----------XXXXX----------

8 Cross Site Scripting (XSS)

8.1 What is Cross Site Scripting (XSS)?

Cross Site Scripting (XSS) is one of the most common and susceptible attacks that all experienced testers are aware of. It is considered one of the most dangerous attacks against web applications and It can bring very serious consequences.

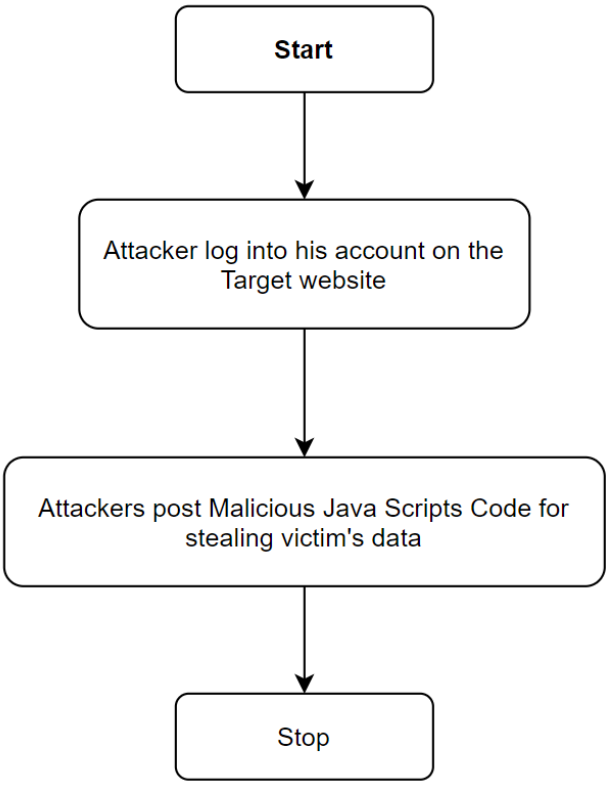
XSS attack is a piece of malicious code, to exploit an XSS vulnerability, the hacker will insert malicious code through scripts to execute them on the client side. The main purpose of this attack is to steal user's identifying data such as: cookies, session tokens and other information.

8.2 Types of XSS Attacks

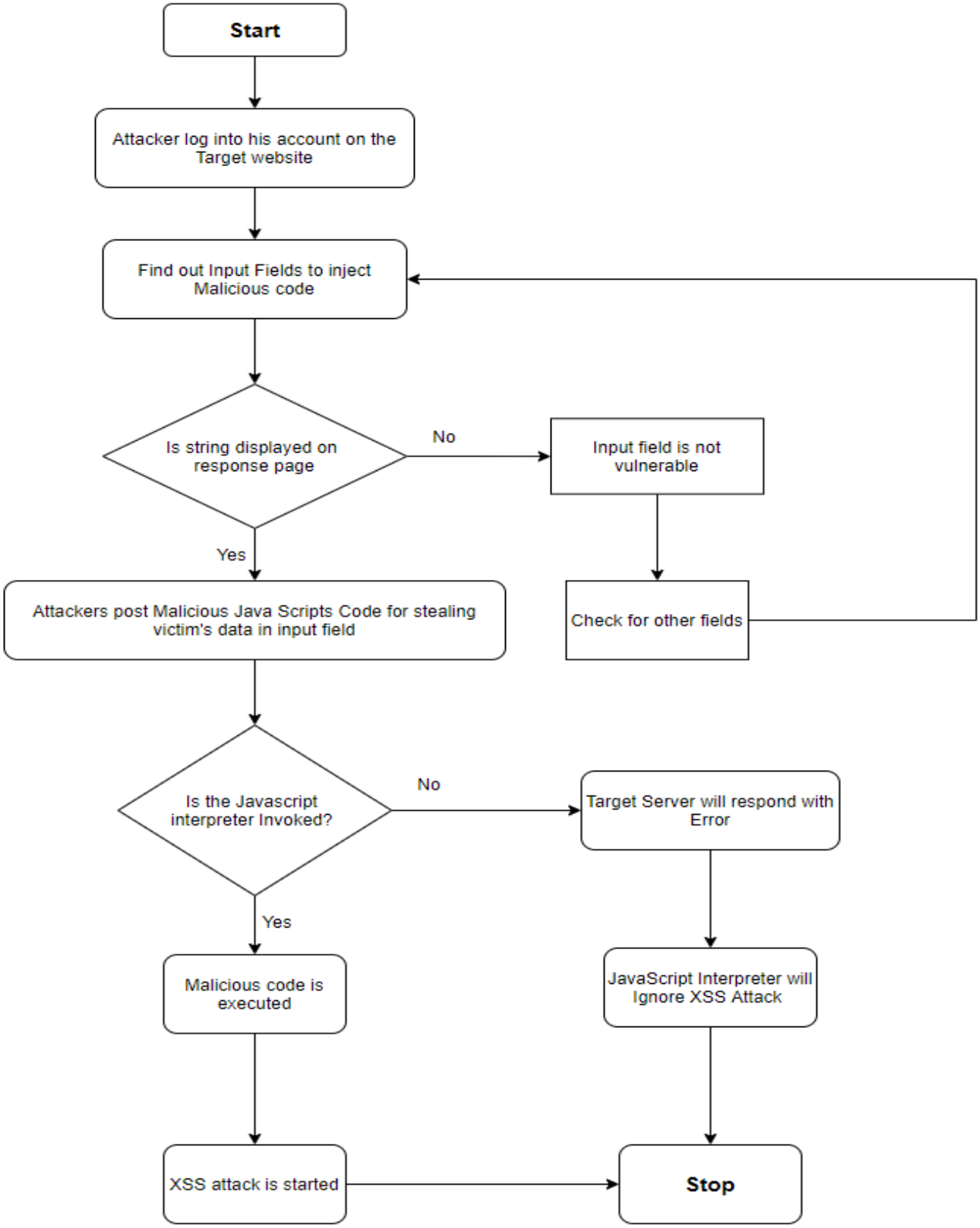
There are 3 main types of XSS attacks as follows:

1. Reflected XSS
2. Stored XSS
3. DOM Based XSS

8.3 Exploitation of XSS Vulnerability on Target website



8.4 Detailed Exploitation of XSS Vulnerability flowchart on Target website

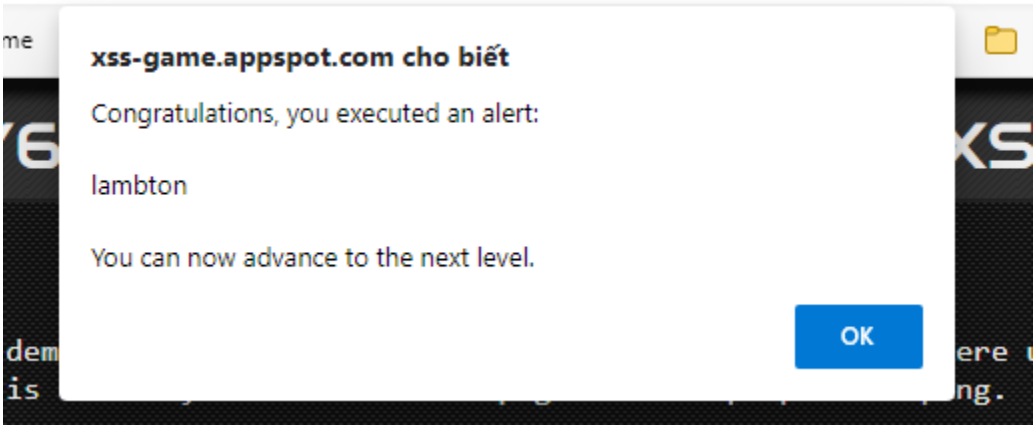

Start
Attacker log into his account on the Target website
Find out Input Fields to inject Malicious code
Is string displayed on response page
No
Input field is not vulnerable
Yes
Attackers post Malicious Java Scripts Code for stealing victim's data in input field
Check for other fields
Is the Javascript interpreter Invoked?
No
Target Server will respond with Error
Yes
Malicious code is executed
JavaScript Interpreter will Ignore XSS Attack
XSS attack is started
Stop

8.5 Tools/Configuration

Since the XSS attack lies in taking advantage of the data input filtering vulnerability, we can use tools to scan or test it ourselves.

We can test by using our personal knowledge to attack by entering a simple script like this " <script>alert(‘lambton’)</script> " or other scripts to check.

Then after pressing the “Search” button, the entered script will be executed



Or we can also scan and perform this vulnerability attack on tools like:

- Acunetix Web Vulnerability Scanner
- Qualys Web Application Scanner
- XSS Scanner
- XSSer
- XSSniper

- XSStrike

8.6 Personal opinion about the effectiveness of XSS attack

XSS attack is considered one of the most dangerous attacks, it can affect web applications, because when a website is attacked with XSS, it will damage both website owners and users on the web. that website, reducing the reputation of that website.

However, this vulnerability is easy to fix and if the programmer uses famous frameworks such as Spring, Django, ASP.NET MVC, then there is no need to worry about this vulnerability because these frameworks all have built-in prevention methods. . Even if the programmer doesn't know anything about XSS, just using the latest framework can prevent quite a few errors. This also greatly affects the effectiveness of exploiting this vulnerability.

----------XXXXX----------

9. Buffer Overflow Attack

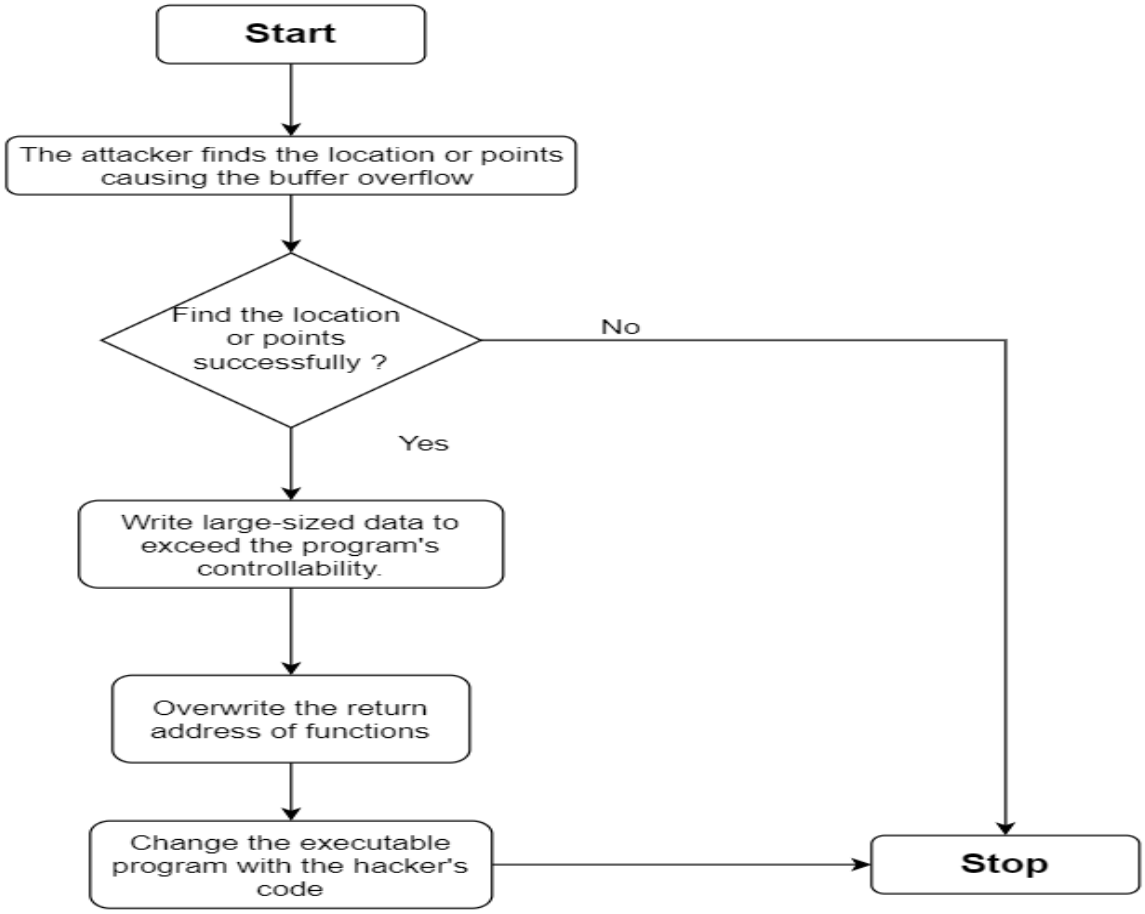
9.1 What are Buffer Overflow Attacks?

Buffer overflow attack is when memory is overwritten multiple times on the stack. This error frequently occurs because the user sends a large amount of data to the application server, which causes the data to be forced to overwrite those contiguous memory locations. This is a programming error that can cause a computer memory access exception and the program to be terminated, or, when deliberately sabotaged by the user, can take advantage of this bug to circumvent system security.

9.2 Common Buffer Overflow Error Types

- Exploiting a buffer overflow on the stack
- Exploiting a buffer overflow on the heap
- Exploit based on software vulnerabilities
- Exploit sites with user interaction but no data binding

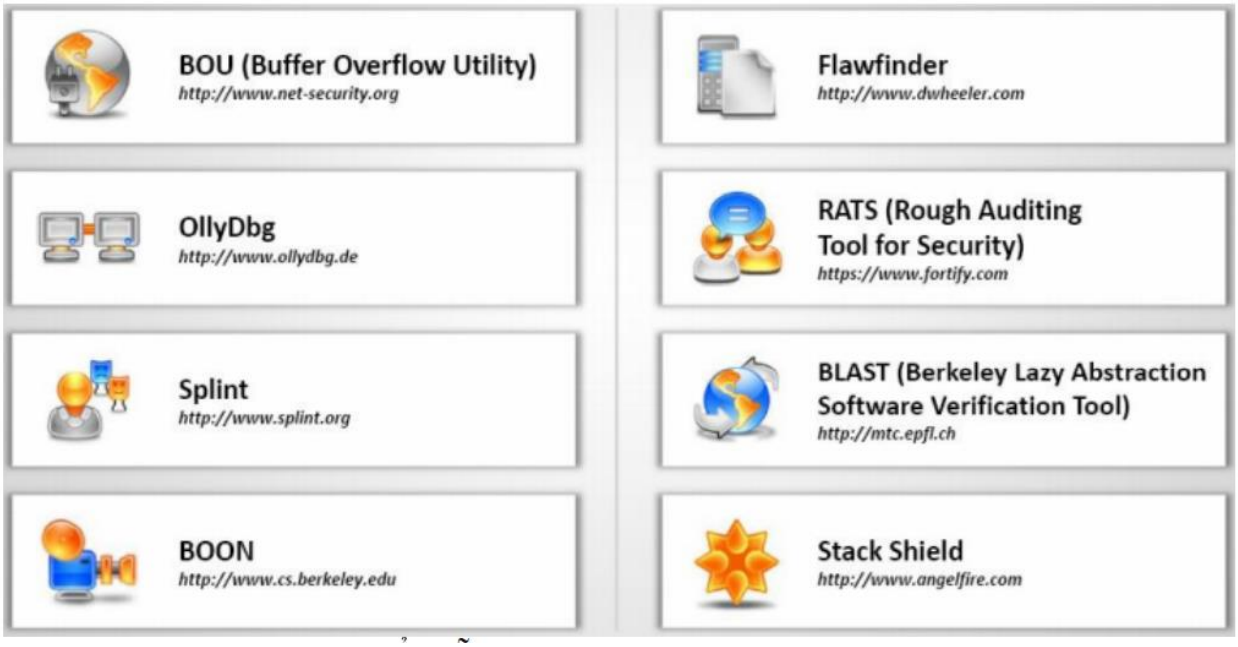
9.3 Exploitation of Buffer Overflow on Target website



9.4 Tools/Configuration

To find buffer overflow errors, hackers can use the BOU (Buffer Overflow Utility) tool to check if the web sites have buffer overflow errors or not. On the other hand, we can also use OllyDbg tool to determine the overwritten addresses.

In addition, we can also use IDA Pro to identify applications with buffer overflow; the following is a list of programs used to check for Buffer Overflow errors



9.5 Personal opinion about the effectiveness of Buffer Overflow attack

Although buffer overflow attack is very dangerous. When these vulnerabilities exist and are attacked, they can lead to application downtime, data loss, or even help attackers take control of the system or give attackers an opportunity to perform many different tricks.

However, these vulnerabilities have been greatly reduced as a result of the emergence of languages with built-in protection and modern operating systems that have added runtime protection capabilities. These measures have been shown to be effective against buffer overflow attacks. That has resulted in the effectiveness of exploiting this vulnerability being significantly reduced.

----------XXXXX----------

10. Cross Site Request Forgery (CSRF) Attack

10.1 What the attack is:

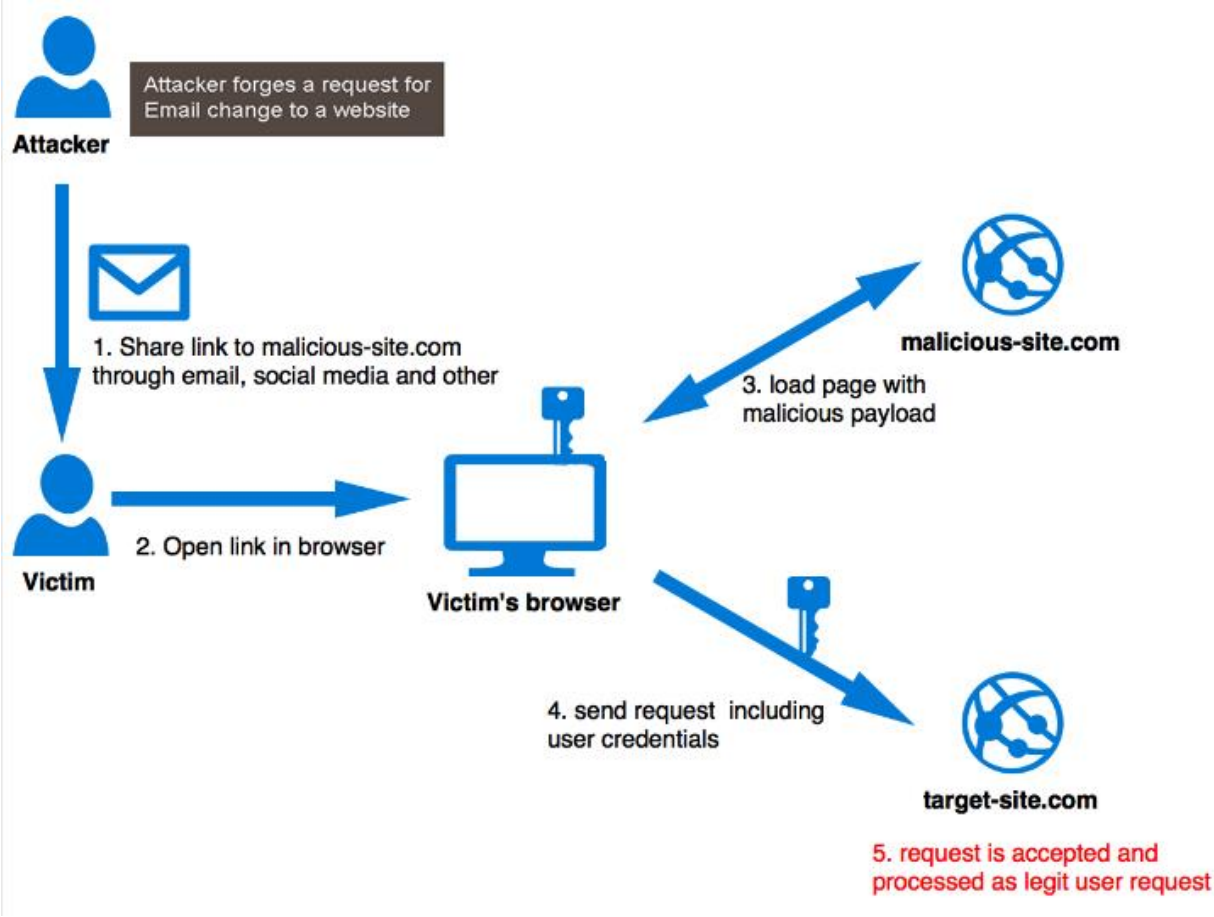
- It is an attack that manipulates the users into doing an unwanted action on a web application where they are authenticated users.
- It is a website exploit which makes a trusted user submit a piece of malware to the website unknowingly.
- This is quite different from the cross-site scripting attack as here the website trusts the user rather than the user trusting the website.
- It is basically using the user to conduct an attack on a website which utilizes HTTP request methods.

10.2 Tools used:

- Burp
- XSRFProbe
- Nexploit
- OWASP's ZAP
- Pinata-csrf-tool

10.3 Flowchart:

- The attack begins with the user receiving a spoofed email, text message or any other form of internet messaging from an unknown source. A good majority of CSRF attacks are social engineering attacks.
- When the user initially sees the message in their browser, it will not take them directly to the target website but will redirect the user to a malicious website.
- This website will load its payload, in this case which would be HTTP requests and then again redirect to the intended target website.
- Once there when the user uses their credentials to access the website and its services, the payload would be released.
- The payload's HTTP request methods with the verbs GET, POST, PUT and DELETE can do the rest of the work such as credential theft, information theft and other such malicious activities.



10.4 Impact of the attack:

- The biggest implication is on the safety of the user's data
- The attacker could very well take over the entire website and perform activities that the user did not know about or did not start

10.5 Opinion:

- This is a very malicious attack.
- It is one of those attacks that cannot be defended against neither by the user or the website due to its rather inconspicuous nature
- A better solution would be to practice safe browsing practices and ignoring any such messages coming from an unidentified source.

----------XXXXX----------

11. Command Injection Attacks

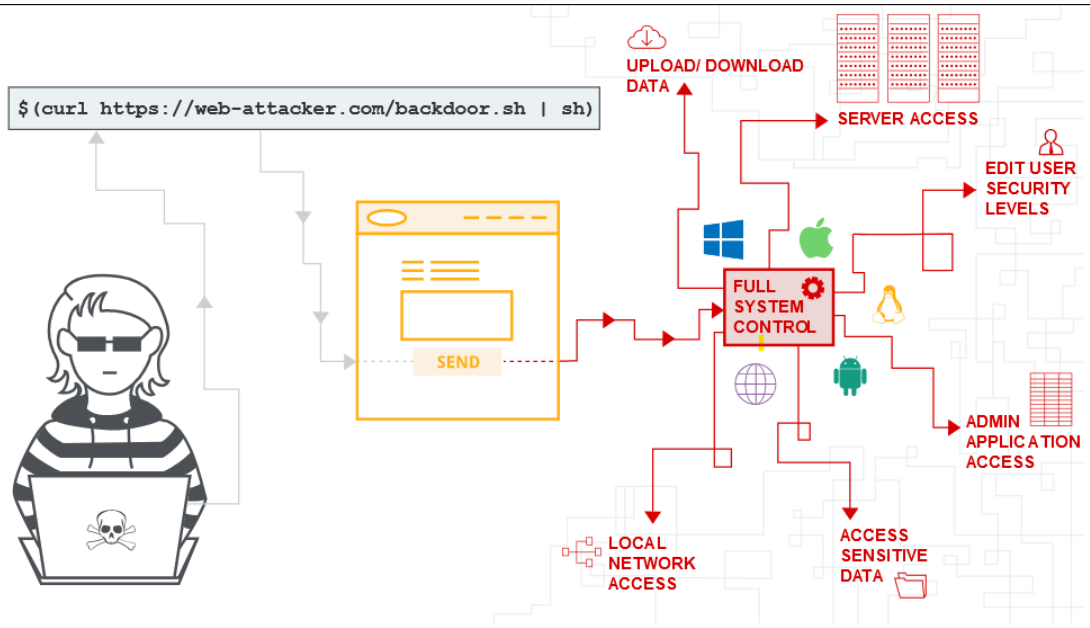
11.1 What the attack is:

- Generally, in any operating system, the user sends some input and based on that the user will get the output.
- But in this type of attack, the attacker can use an application vulnerability, insecure data transmission or any other such vulnerability to inject a set of commands into the system shell.
- When the user takes an action, the output will be sent to the attacker as well.
- The attack will basically let the user over-extend the functionality of any application and instead of using malicious code to inject the commands, use this over -extended application to get to the system shell.
- Overall, this attack will give more control to the attacker.

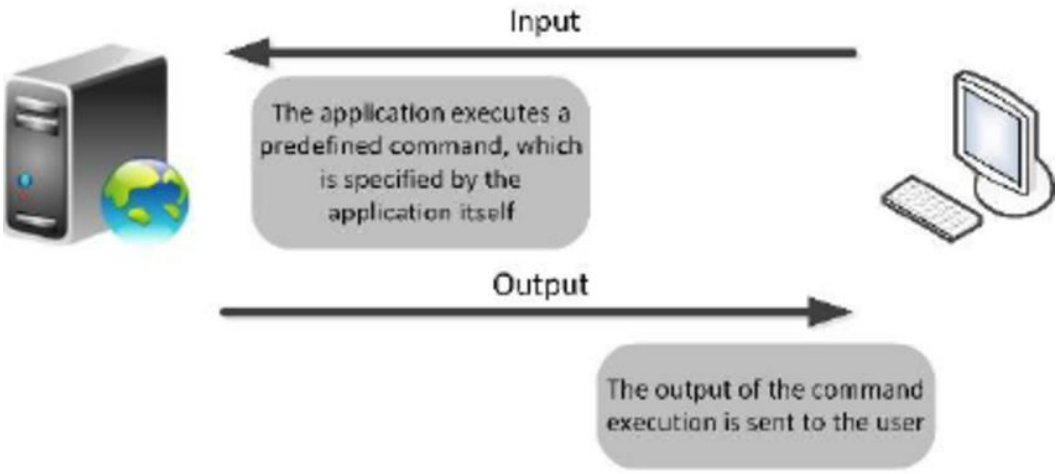
11.2 Tools used for this attack:

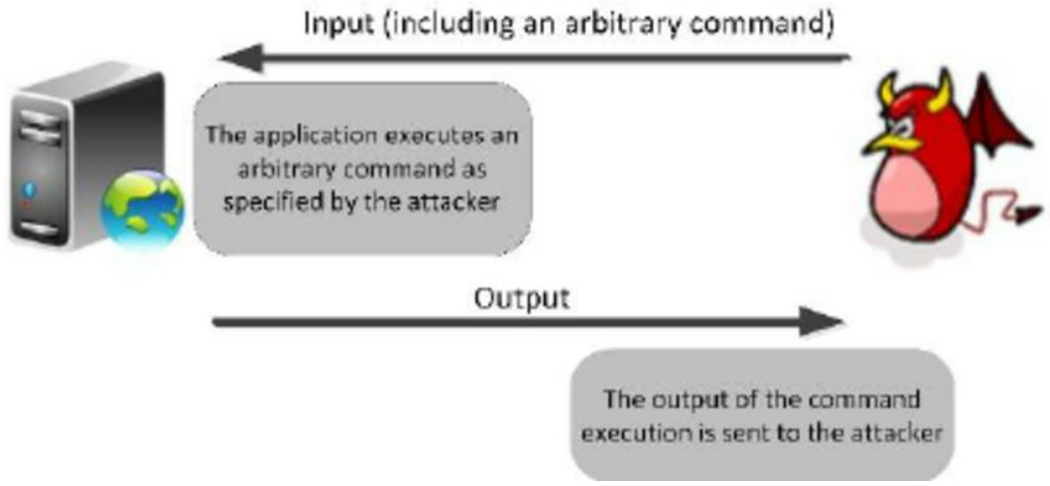
- Burp
- Commix
- Metasploit
- MimiKatz

11.3 Flowchart



- The attacker will target any system or server by first doing some reconnaissance, looking for open ports and other details, through a compromised web application.
- The attacker sets up a listener which will report back to him whenever their arbitrary commands have been executed.
- When he finds the necessary data, the attacker will inject the malicious commands to the intended target system.
- Once sent, the commands will be accepted in the system shell and from there the attacker can use the listener already set up to find the relevant data and move forward with the attack
- An easier way to look at this is





11.4 Types of command injections:

- Arbitrary command injections
- Arbitrary file uploads
- Insecure serialization
- Server-Side Template injection
- XML external entity injection

11.5 Impact of the attack:

- The entire attack represents a program error in the OS
- It can lead to privilege escalation
- An attacker can steal or even alter sensitive files leading to chaos
- Upload a malware file and infect the server
- Potential ransomware attacks
- Disruption of services

11.6 Opinion:

- I couldn't find too much information on the impact of this attack, but from what I have found, this attack has very serious consequences.
- There is no telling what an attacker can potentially do if this attack succeeds.
- It can be a nightmare scenario if an attacker has complete system access to a server and the security team must get the control back.

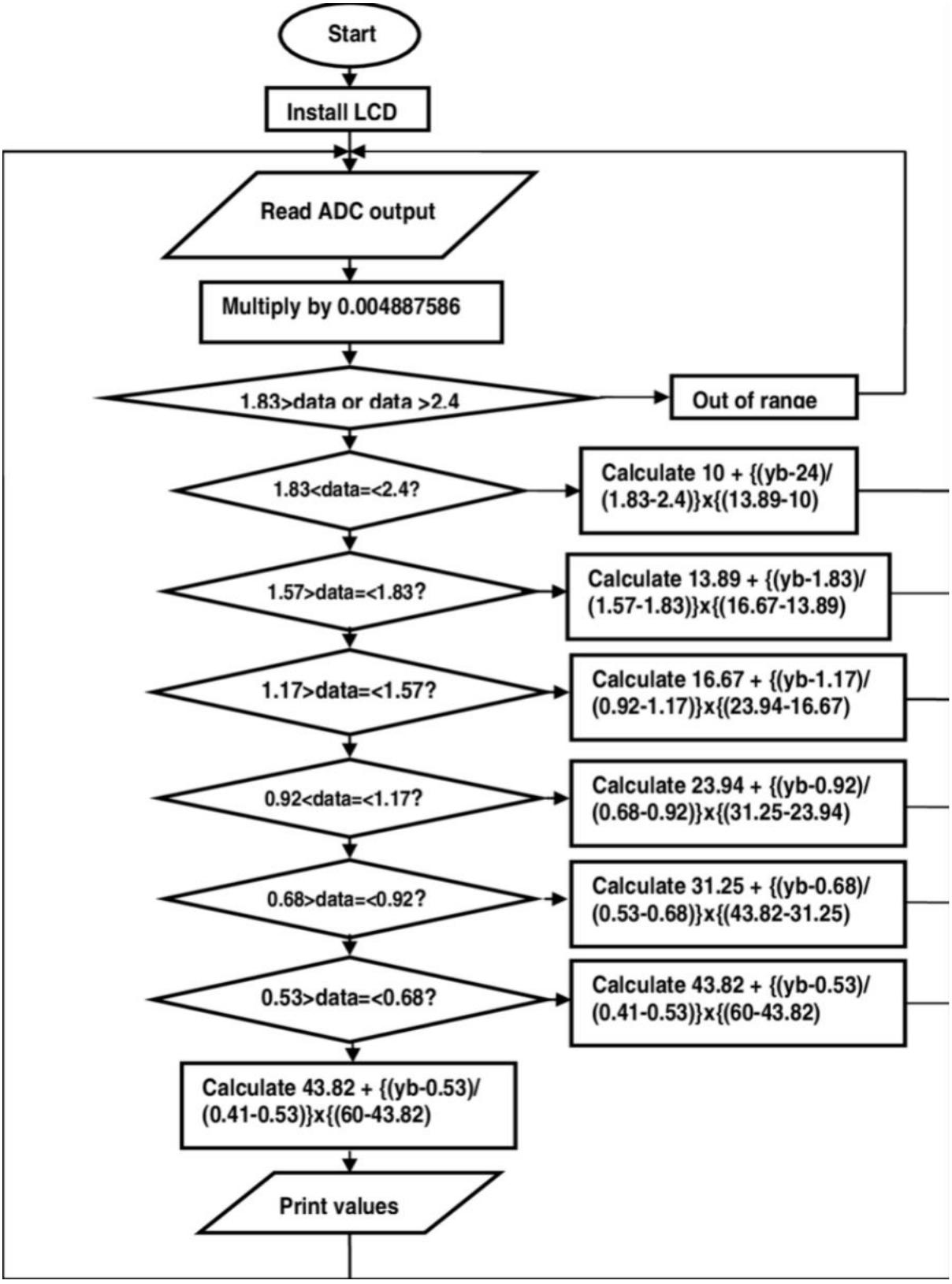
----------XXXXX----------

12 Source code

12.1 What is source code?

- Source code frequently contains some type of delicate data whether it be design related data (for example data set certifications) or just data on how the web application capacities.
- Whenever uncovered, such data might conceivably be utilized by an aggressor to find coherent imperfections and grow into an ensuing chain of assaults which would not be conceivable without approaching the application's source code.
- These might incorporate assaults like SQL infusion, information base takeovers, and distant code execution.
- It is normal practice for web applications to serve non-HTML records, like PDFs, picture documents, and Word reports that are tweaked for a particular client.
- Contingent upon the source code, data set association strings, username, and passwords, the inner functions and business rationale of use may be uncovered. With such data, an assailant can mount the accompanying sorts of assaults:
    - 1. Access the different information assets. Contingent upon the advantages of the record got from the source code, it very well might be feasible to peruse, refresh or erase discretionary information from the data set.
    - 2. Access secret phrase secured managerial instruments like dashboards, the board consoles and administrator boards, subsequently overseeing the application.
    - 3. Foster further assaults by exploring the source code for input approval blunders and rationale weaknesses.

12.2 Flowchart of the attack



12.3 Tools and configuration required for the attack to materialize:

- ABHSH
- Astree
- Attack flow
- Boon

12.4 My opinion on effectiveness of the attack

- Contingent upon the source code, information base association strings, username and passwords, the inward functions and business rationale of the application may be uncovered. With such data, an assailant can mount the accompanying kinds of assaults:
    - Access the information base or different information assets. Contingent upon the advantages of the record acquired from the source code, it could be feasible to peruse, refresh or erase discretionary information from the data set.
    - Access secret phrase secured regulatory components like dashboards, the executive consoles and administrator boards, thus overseeing the application.
- **Prevention**
    - Confirm exactly what aspects of the source code are actually disclosed; due to the limitations of these types of vulnerability, it might not be possible to confirm this in all instances. Confirm this is not an intended functionality.
    - If it is needed by the application, alter its permissions to secure public users from accessing it. If it is not, then remove it from the web server.
    - Ensure that the server has all the current security patches applied.
    - Remove all temporary and backup files from the web server.

----------XXXXX----------

REFERENCES

https://www.greycampus.com/opencampus/ethical-hacking/session-hijacking-process

https://diarium.usal.es/pmgallardo/2020/10/26/list-of-session-hijacking-tools/

https://www.netsparker.com/blog/web-security/session-hijacking/

https://securityboulevard.com/2020/11/the-ultimate-guide-to-session-hijacking-aka-cookie-hijacking/

https://us.norton.com/internetsecurity-id-theft-session-hijacking.html

https://www.thesslstore.com/blog/the-ultimate-guide-to-session-hijacking-aka-cookie-hijacking/

https://www.guru99.com/ultimate-guide-to-dos-attacks.html

https://us-cert.cisa.gov/ncas/tips/ST04-015

https://medium.com/geekculture/simple-but-powerful-denial-of-service-dos-attack-8c7dfd60045f

https://www.dotmagazine.online/issues/new-directions/dangerous-ddos-attacks

https://www.ncsc.gov.uk/collection/denial-service-dos-guidance-collection

https://en.wikipedia.org/wiki/Cross-site_request_forgery

https://owasp.org/www-community/attacks/csrf

https://www.netsparker.com/blog/web-security/command-injection-vulnerability/

https://owasp.org/www-community/attacks/Command_Injection

https://www.imperva.com/learn/application-security/command-injection/

https://portswigger.net/web-security/os-command-injection

https://en.wikipedia.org/wiki/File_inclusion_vulnerability

http://docshare02.docshare.tips/files/3804/38049226.pdf

http://www.infosecwriters.com/text_resources/pdf/Unvalidated_Input_TOlzak.pdf

https://developer.apple.com/library/archive/documentation/Security/Conceptual/SecureCodingGuide/Articles/TypesSecVuln.html

https://developer.apple.com/library/archive/documentation/Security/Conceptual/SecureCodingGuide/Articles/ValidatingInput.html#//apple_ref/doc/uid/TP40007246-SW3

https://www.offensive-security.com/metasploit-unleashed/file-inclusion-vulnerabilities/

https://www.softwaretestinghelp.com/tools/top-40-static-code-analysis-tools/

https://www.nist.gov/itl/ssd/software-quality-group/source-code-security-analyzers