Vulnerability of web-server and web-application and ways of invasion

Edris Lotfpouri

University of Kurdistan, Iran

Abstract

With the explosive growth of the internet network, searching and surfing through websites and web apps becoming our everyday habit, the point is, this websites and web-based applications that we use on our devices are getting information from us to improve and personalize their services for their clients; by using the word "us" I mean users, so this massive amount of data from different people all over the world may get some others attentions, like hackers. Despite the efforts of the companies or peoples who own a website or web app which people use, to protect their client privacy, there are people who can find a way to break into this privacy. In this article I have done my best to outline the main architecture of the network and describe its concept, how we get in danger and the type of the vulnerabilities, introduction to a series of famous attacks and lastly how to avoid as much as we can from this kind of threats.

I. Introduction

First we need to get acquainted with some definitions in the world of the Internet Network.

- **Vulnerabilities:** Let's begin with why web-servers and web-based applications are targets for attackers. First, they are always available via internet and a significant part of their functionality is available to anonymous users. Second, because of interfacing numbers of the web-based applications with back-end components, such as mainframes and product data bases, that might contain sensitive data such as credit card information. Last but not least, knowledge and technology used to implement, test and interact web-based applications is inexpensive, well known and widely available.
  One of the obvious reasons that create vulnerabilities is, the applications code is often developed by few programmers with little security training and under a strict time limit.
  Some vulnerabilities are due to architectural choices, such as use of relational data bases as back-ends for long-term storage which lead to vulnerabilities such as SQL injection and permanent Cross-Site-Scripting as known as XSS. Other causes of web-based vulnerabilities are the incorrect handling of trust relations between clients and servers.
- **Penetration Testing:** After the introduction to the vulnerabilities, now is the time for us to know about penetration testing. A penetration test is authorized, scheduled and systematic process of using known vulnerabilities in an attempt to perform an intrusion into host, network or application resources. Penetration testing or we can call it in short form "PenTest", stresses not only the operation, but the implementation and design of product or system.
  We have two type to conduct penetration test which known as internal penetration and external penetration tests.
  - **Internal penetration tests:** Tests intended to identify vulnerabilities with physical access or exposure to social engineering.

- **External penetration tests:** Tests intended to identify vulnerabilities that are present for connections that have been established through the organization connection to the Internet (also known as firewall or gateway).

Why penetration test is necessary?

If a vulnerability unfolded and utilized by an unauthorized individual to access company or person resources, those resources can be compromised. The objective of penetration test is to address vulnerabilities before they can be unfolded and utilized.

A large part of penetration testing is art rather than science. The effectiveness of penetration testing depends on skill and experience of the tester.

- **Hacking vs Ethical Hacking:** Hacking is the process of exploiting security flaws, bugs or vulnerabilities in a network or software in order to steal, destroy, change the data or interrupt normal operations. A person who performs hacking activities is called hacker.
  Ethical hacking is the process of exploiting security flaws, bugs and vulnerabilities in a network or software in order to identify loopholes and fix them before a malicious hacker find and exploit them.
  Table.1 show a comparison between hacking and ethical hacking.

TABLE 1
COMPARISON OF HACKING AND ETHICAL HACKING

Hacking	Ethical Hacking
Stealing sensitive user information	Identify and fixing security vulnerabilities to secure system
Destroying, changing and erasing data bases	Assess an organization security and quality
Bringing down a network by interrupting normal operation	Implementing data security regularity compliance

II. SYSTEM ARCHITECTURE

To understand how computers are connected and communicate through internet network, we bring up two model which are called OSI model and TCP/IP model.

- **OSI Model:** OSI is the short form of the “Open System Interconnection” and this model defines seven layers that show how applications running upon network-aware devices may communicate with each other.
  - **Layer 1: The Physical Layer:** This is the lowest layer of the OSI model and it activate, maintain and deactivate the physical connection. It is also responsible for transmission and reception of unstructured raw data over network. Eventually, they have to pass strings of ones and zeros down the wire.
  - **Layer 2: The Data Link Layer:** This layer synchronizes the information which is to be transmitted over the physical layer. The main function of this layer is to make sure

and it can manage the frame traffic data transfer is error free from one node to another over physical layer and it can manage the frame traffic control over the network.
It has two sub layers:

1. **MAC Address:** MAC is the short form of the Media Access Control and it is a twelve hexa-decimal digit which the first six digits are related to the factory and the other six are for the company.
2. **LLC Layer:** LLC is the short form of Logical Link Control and it has the duty to check out the data to be relevant with that system.

- **Layer 3: The Network Layer:** This layer routes the signal through different channels from one node to another and it decides by which route data should take. The IP protocol lives at this layer and also the firewall configuration is in this layer.
- **Layer 4: The Transfer Layer:** This layer breaks the data into small units so that they are handled more efficiently by the network layer. It also decides if data transmission should be on parallel path or single path.
- **Layer 5: The Session Layer:** This layer manages and synchronize the conversation between two applications and also check if the data are resynchronized properly, so the ends of the message are not cut prematurely and data loss is avoided.
- **Layer 6: The Presentation Layer:** This layer takes care that the data is sent in a such a way that the receiver will understand the data and be able to use it. It performs data compression, data encryption and data conversion.
- **Layer 7: The Application Layer:** Human-Computer interaction layer, where application can access the network services.

Fig.1 show the OSI Layers with a brief description.

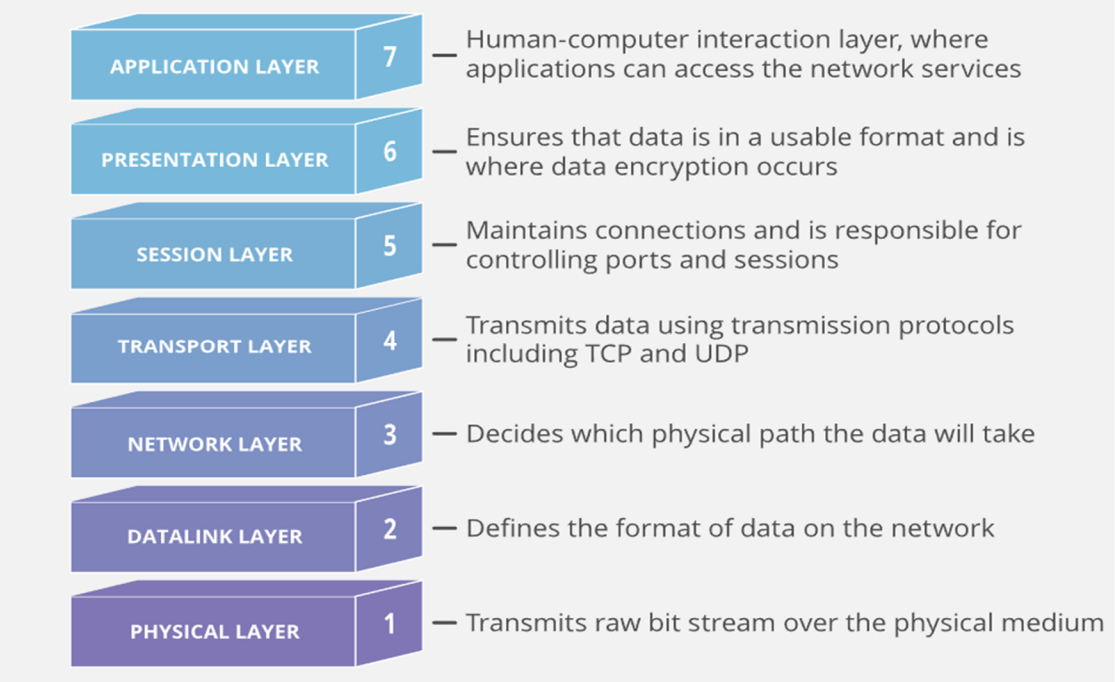

Fig. 1. Seven Layers of OSI Model

- **TCP/IP Model:** It stands for Transmission Control Protocol/Internet Protocol. It is the concise version of OSI model and it has four layers.
    - **Layer 1: Application Layer:** This layer performs the functions of the three top layers of OSI which are Application, Presentation and Session Layer. It is responsible for node-to-node communication and control the user-interface specification. Protocols such as HTTP, HTTPS, FTP, SSH and Telnet are present in this layer.
    - **Layer 2: Transport Layer:** This layer is also known as Host-to-Host layer and it is very similar to transport layer of OSI. It is responsible for end-to-end communications and error-free delivery of data.
    - **Layer 3: Internet Layer:** This layer is parallel with the OSI's Network Layer and it defines the protocols which are responsible for logical transmission of data over the entire network.
    - **Layer 4: Network Access Layer:** This layer corresponds to the combination of Data Link Layer and Physical Layer of the OSI model. It looks out for hardware addressing and the protocols present in this layer allows for the physical transmission of data.

Fig. 2 is show a brief comparison of OSI and TCP/IP model.

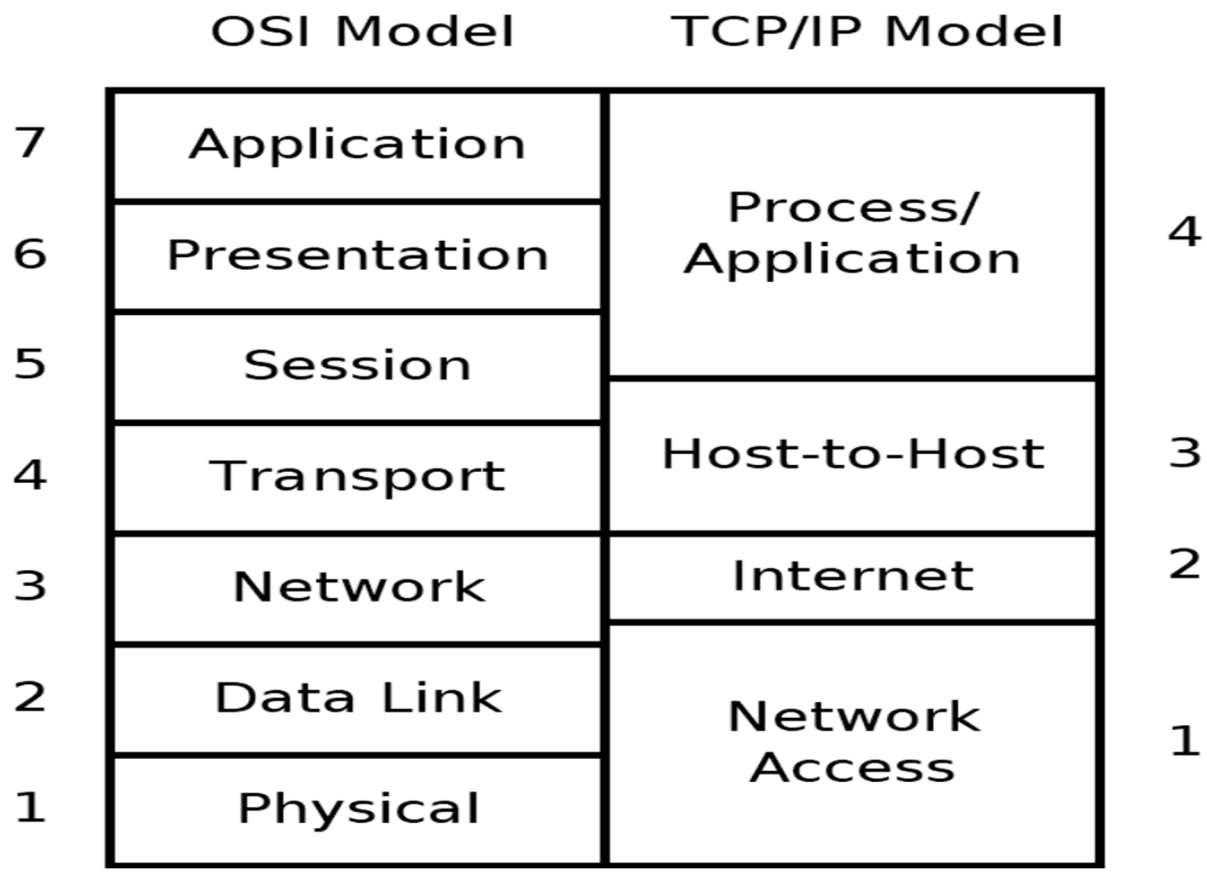

Fig. 2. Comparison of OSI and TCP/IP model.

III. VULNERABILITIES OF WEB-SERVER AND WEB-SERVER ATTACKS

We have studied the architecture of the computers connection and communication over the network now is time for knowing vulnerabilities of the network web-servers and types of the threats for them.

- **Vulnerabilities of web-servers:** The first thing that we can mention as a vulnerability is the bad configuration of the web-server. Second thing to mention is that the vulnerability may be create because of problems on the OS or software. Third thing is vulnerability of the demos version of OS or software. Most of the todays vulnerabilities with web-servers are ascribed to the creativity of the hacker community to find ways to handle and change data on the web-server in ways the web-server developer did not intend and handle.
- **Web server invasion:** Defacement is the thing that hackers do usually when they successfully got into a web-server but maybe they don't do it and just take the available information. We have got different ways to attack a web-server but some of these ways are very popular and useful for hackers. Now we are going to introduce some of the popular ones.
    - **Getting The Administrator Badge:** As we know the administrator is the highest level in a web-server or a computer which has access to everything. The hackers try to get this badge by attacks like Man-in-the-middle.
    - **Get Administrator Password:** Because of the same reason which I mentioned in the top hackers try to get administrator password. The hackers try to get this password by an attack which is called Brute-force.
    - **DNS:** The hackers use DNS to take users to another web-server to steal their info.
    - **Routing The Client:** The hackers route the users to somewhere else which is belong to hacker after attacking to firewall or router.
    - **Using Telnet or SSH:** The hackers use these remote access protocols to get access to the web-server information.
    - **URL Poisoning:** The hackers add an identification number to the page address line of the web browser and when a user visits a particular site. They use this for tracking the user and finding the things which user interest in.

IV.HOW TO SAFE OUR WEB-SERVER

There are ways to avoid and reduce the threats by the hackers which are going to introduce it. Doing these things are so effective but it is not going to solve the problem completely because the hackers especially those who are very creative can find new ways which is called zero-day vulnerabilities.

1. Changing the name of the Administrator and use complex password.
2. Deactivate default website and FTP site.
3. Deactivate directory browsing in web-server setting (use to show the directories).
4. Check user entries for attacks like Buffer overflow and SQL injection.
5. Deactivate remote managing to avoid attacks by Telnet and SSH.
6. Using a script to routing an inappropriate entry to 404 (File not found).
7. Install patches and service packs for OS and software.
8. Deactivate NETBIOS which is in the Session layer of the OSI model to avoid using API and SMB (Server Message Block).

These are some of the ways to handle, block or reduce the threats of the hackers.

V. VULNERABILITIES OF WEB APPLICATIOS AND IMPORTANT ATTACKS

The goal of hacking web applications is to get private information of the users like identity, phone number or credit card information and password. There are five steps for doing attacks on web applications which is shown at the fig. 3.

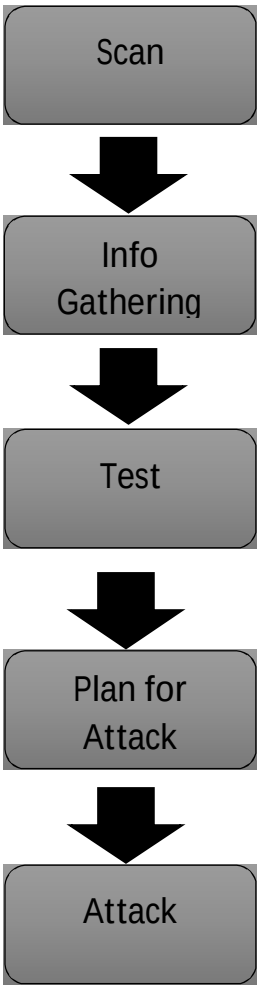

Fig.3. Five steps of the web-app attacks

- **Important attacks:** There are ways to attack web applications but some of them are very common and important. So, we are going to study these attacks with detail.

  Cross-Site-Scripting (XSS): In this attack the hacker uses the web-app to send destructive JS (Java Script) code. This could be used to collect form data from the web-page, information from the session, information from the server which "owns" the web-page, install Trojan or information about the user that is viewing the web-page. The destructive script forward the collected information to the place which hacker has

chosen. Even if the XSS attack does not collect data, the technique can be used for defacing the page. Fig.4 show that how XSS works.

- **SQL Injection:** Another common and important attack is SQL injection which the hacker using it to get a direct access to data base. It works through input box of the user, the input box is the place which user enter username and password in a web-site or adding some data to the URL to search for a particular word in an application. Hacker could use vulnerable web-app to go around the security checkouts. SQL injection has another type which is called Blind SQL Injection.
- **Blind SQL Injection:** This kind of attack works based on a regular error. The web application get data from client and run it in SQL queries without checking it. With this attack the hacker can add, remove, change or get content in the data base.
- **Buffer Overflow:** Buffer is a sequential section of the memory allocated to content anything from character string or an array of integers. Buffer overflow occurs when more data put into a fix-length buffer than the buffer can handle. The extra information can overflow into adjacent memory space, corrupting or overwriting the data held in that space. This overflow usually results in a system crash but it also is an opportunity for hacker to run arbitrary code or manipulate the coding to prompt malicious actions. The SQL injection and Buffer overflow caused for same reason and that reason is invalid parameter.
- **Cookie Poisoning and Snooping:** Cookies used to save sessions condition, by poisoning the cookies hackers can inject destructive content and get private information. In this attack hackers poison cookies or stealing them.

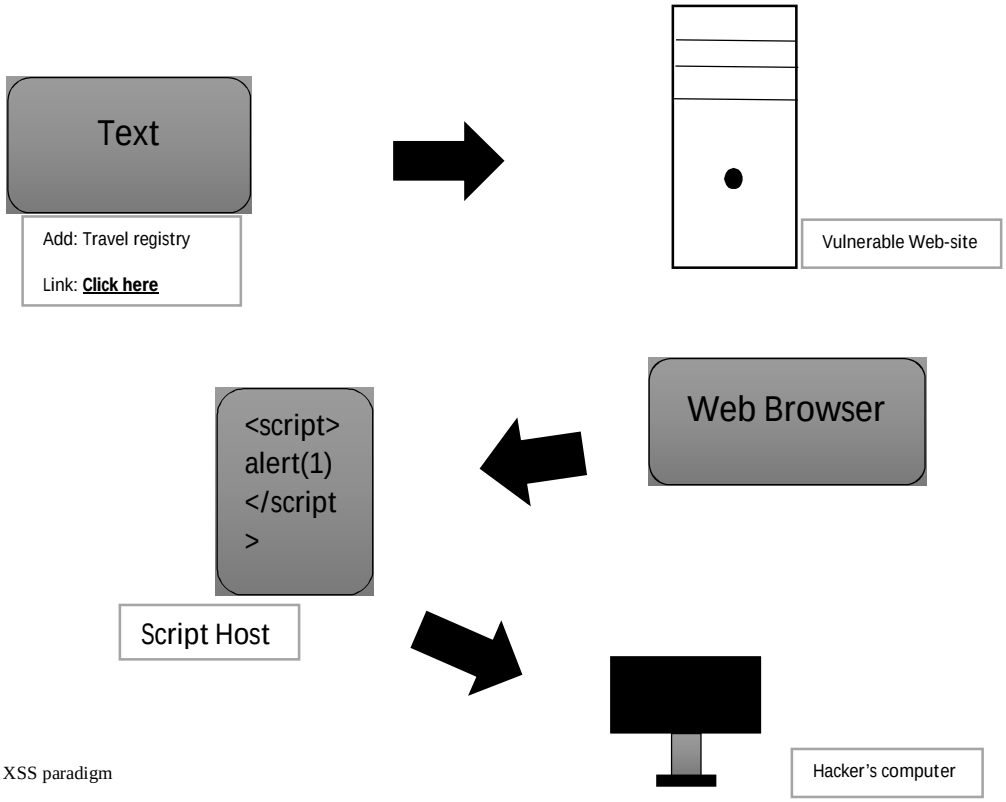

Fig. 4. The XSS paradigm

VI. HOW TO PREVENT ATTACKS

We have introduced the attacks and now we are going to study ways to prevent this attacks.

- **Prevent Cross-Site-Scripting (XSS):** Preventing XSS is trivial in some cases but can be much harder depending on the complexity of the application and the ways of handling user data. To prevent this vulnerability, the programmer must filter input on arrival, encode data on output and use appropriate response headers.
- **Prevent SQL Injection and Blind SQL Injection:** If programmers do not spend enough time to checkout variables on the user entry, there will be un predictable consequences. To prevent this vulnerability, the programmer must not let the user to change the syntax of the SQL. The app should use safe interface like JDBC or ADO for processing on the data. There are other ways to handle this problem.
    1. Reduce the level of the connection between user and database.
    2. Use strong and complex password for Administrator an SA.
    3. Deactivate descriptive messages to avoid sending unnecessary information to hacker.
    4. Never connect to the database with the admin account because your system maybe infected.
    5. Use encryption and cryptography for classified information.
    6. Do not accept wrong entries from user.
- **Prevent Buffer Overflow:** The simple way to prevent this vulnerability is using languages which they do not let this thing happen like Java and Python. C language allow these vulnerabilities through direct access to memory. Completely changing the language is not always possible but we can use secure alternatives and when that is not possible, it is necessary to perform manual bounds checking and null termination when handling string buffers.
- **Prevent Cookie Poisoning:** We can prevent this vulnerability by doing things below:
    1. Do not save un hashed password into cookies.
    2. Set time out for cookies which after that time the cookie does not work anymore.
    3. Identity information of cookies must be along with IP.

VII. AUTHENTICATION

Web servers and web applications supporting various ways of authentication to secure their connection and communications with users, the most popular and common one is HTTP authentication. There are two types of HTTP authentication:

1. **Basic:** The username and password send out without being hashed.
2. **Digest:** The username and password are hashed with a model called Challenge-Response.

In addition to HTTP authentication, the web servers use NTLM, certificated-based, token based and biometric authentication too. We are going to study those authentications but before that we are going to introduce an authentication protocol which is very common and useful.

- **Challenge-Response Protocol:** In computer security, challenge-response authentication works like this, which the party provides a question (challenge) and another party must provide a valid answer (response) to be authenticated. The simple example for this is password which we use to enter our accounts.

- **NTLM Authentication:** NTLM is the short form of New Technology Lan Manager. It uses a challenge-response authentication protocol which is using three messages and a fourth additional message if integrity is desired.
    - First, client establishes a network path to the server and send out a NEGOTIATE_MESSAGE showing it capabilities.
    - Second, the server response with a CHALLENG_MESSAGE which is used to establish the identity of the client.
    - Third, the client response to the challenge with AUTHENTICATE_MESSAGE.
- **Certificate-Based Authentication:** This authentication uses digital-certificate to identify a user. Digital-certificate is a digital document that includes the public key bound to an individual, organization or computer. Public-key cryptography or asymmetric cryptography is an encryption scheme that uses two mathematically related but not identical keys one named Public key and the other is Private key. Alike symmetric key algorithm which is relying on one key to both encrypt and decrypt, in asymmetric cryptography each one of the keys performs unique function, the public key is used to encrypt and private key is used to decrypt. These certificates are issued by certificate authorities who have documented policies for determining owner identity and distributing certificates.
- **Token Based Authentication:** This authentication is a hardware device which provides a code for sixty second to login to your account.

Using authentication is very helpful to prevent vulnerabilities.

VIII. CONCLUSION

This article describes web-server and web application vulnerabilities, also the ways to use this vulnerabilities and ways to prevent them. We have started with describing the architecture of ho computers connect and communicate in order to understand better that, how we get in danger and how hackers use those vulnerabilities. But just before that we have compare hackers and ethical hackers to know that, how they deal with vulnerabilities. After the architecture of the system we have get into a section which it's goal was to introduce the vulnerabilities of the web-servers and attacks related to them. The next step was ways to get web-server safer and less vulnerable. We have continued till we get to introducing section of web applications vulnerabilities and describing the important attacks for this kind of applications. Then we talked about the ways to prevent these attacks in detail. At the end the authentication part which is really important for us to be more safe, therefore we have discussed the ways of authentication.

REFERENCES

[1] https://docs.oracle.com/cd/E74890_01/books/Secur/secur_ssoauth009.htm

[2] https://www.globalsign.com/en/ssl-information-center/what-is-public-key-cryptography

[3] https://www.cybrary.it/study-guides/what-is-certificate-based-authentication/

[4] https://www.globalsign.com/en/blog/what-is-certificate-based-authentication

[5] https://en.wikipedia.org/wiki/NT_LAN_Manager

[6] https://www.exploit-db.com/docs/14475

[7] https://sites.cs.ucsb.edu/~vigna/publications/2007_cova_felmetsger_vigna_webvuln.pdf

[8] https://dl.acm.org/doi/pdf/10.1145/366173.366183

[9] https://www.sans.org/reading-room/whitepapers/testing/penetration-testing-you-265

[10] https://devcount.com/hacking-vs-ethical-hacking/

[11] https://www.studytonight.com/computer-networks/complete-osi-model

[12] https://www.cloudflare.com/learning/ddos/glossary/open-systems-interconnection-model-osi/

[13] https://www.geeksforgeeks.org/tcp-ip-model/

[14] https://whatis.techtarget.com/definition/URL-poisoning-location-poisoning

[15] https://www.giac.org/paper/gsec/3729/web-server-vulnerabilities-defense-in-depth-strategy-squid-proxy/105970

[16] https://www.veracode.com/security/buffer-overflow

[17] CEH V10: EC-Council Certified Ethical Hacker Complete Training Guide with Practice Labs: Exam: 312-50