Unit-5:
5.1 Ethical Hacker
5.1.1 Roles and Responsibilities
5.1.2 Benefit of Ethical Hacking
5.1.3 Skills require to become Ethical hacker
5.2 Penetration testing concepts
5.2.1 Phases of Ethical hacking
5.2.1 Areas of penetration testing
5.3 SQL Injection:
5.3.1 Concepts of SQL Injection
5.3.2 Types of SQL Injection
5.3.3 Case study of SQL Injection
5.4 Firewall:
5.4.1 Concepts of Firewall
5.4.2 Types of Firewall
5.4.3 Working, Advantages and Importance of Firewall

A hacker is an unauthorized user who attempts to or gains access to an information System. Hacking is a crime even if there is no visible damage to the system, since it is an invasion in to the privacy of data. There are different classes of Hackers.

5.1 Ethical Hacker

An Ethical hacker also referred to as a ***"white hat hacker"***. An Ethical Hacker is the first to get access to the target system. Thus, the organization's security staff may patch a weakness in the system to prevent an intruder from abusing it. In short, Ethical hackers are well recognized in their profession for their job of protecting the system.

Hacking professionals adhere to four basic protocols:

- Stay within the bounds of the law: For example, perform a security evaluation only if you have the correct permissions.
- Determine the scope of the project: For example, if you're going to use an ethical hacker, you need to know the size of the investigation.
- Report flaws in the system: Provide a full report on how to cope with these problems.
- Keep in mind that the information you're dealing with is very sensitive. In certain cases, ethical hackers may be required to sign a non-disclosure agreement in addition to any extra terms and conditions imposed by the firm they are inspecting, depending on the sensitivity of the data they inspect.

5.1.1 Roles and Responsibilities

Ethical Hackers Responsibilities Role:

- **In-depth Knowledge of Security:** Ethical hackers should be aware with potential threats and vulnerabilities that can hack organisational systems. Ethical hackers are hired by organisations for their expertise skills and quick resolution to security vulnerabilities. They

should be cyber security professionals having knowledge of the computer systems, network and security.

- **Think like Hackers:** They are supposed to think like hackers who want to steal confidential data /information. Ethical hackers look for areas that are most likely to be attacked and the different ways in which attack can take place.
- **In-depth Knowledge of the Organisation they intend to provide Service:** Ethical hackers should be familiar with the services of the functional working of the organisation they are associated with. It should have the knowledge about the information that is extremely safe and needs to be protected. Ethical hackers should be capable of finding the attack methods for accessing the sensitive content of the organisation.

Ethical Hackers Responsibilities:

- **Hacking their own Systems:** Ethical hackers hack their own systems to find potential threats and vulnerabilities. They are hired to find vulnerabilities of the system before they are discovered by hackers.
- **Diffuse the intent of Hackers:** Ethical hackers are hired as a Precautional Step towards Hackers, who aim at breaching the security of computers. Vulnerabilities when detected early can be fixed and safe confidential information from being exposed to hackers who have malicious intentions.
- **Document their Findings:** Ethical hackers must properly document all their findings and potential threats. The main part of the work they are hired by the organisations is proper reporting of bugs and vulnerabilities which are threat to the security.
- **Keeping the Confidential Information Safe:** Ethical hackers must oblige to keep all their findings secure and never share them with others. Under any kind of situation they should never agree to share their findings and observations.
- **Sign Non-Disclosure Agreements:** They must sign confidential agreements to keep the information they have about the organisations safe with them. This will prevent them to give - out confidential information and legal action will be taken against them if they indulge in any such acts.
- **Handle the loopholes in Security:** Based on their observations, Ethical hackers should restore/ repair the security loopholes. This will prevent hackers from breaching the security of the organisation from attacks.

Ref: https://www.geeksforgeeks.org/ethical-hacker-required-skills-roles-and-responsibilities/

5.1.2 Benefit of Ethical Hacking

Ethical hackers are well recognised in their profession for their job of protecting the system. Below are the advantages of being an ethical hacker:

- Prevent harmful cyber attacks.
- Prevent penetration attacks of intruders.
- Find loopholes in the system and repair them with their expertise.
- Establish security and safety measures within the system.
- Prevent cyber terrorism and hacks from taking place.

Advantages of Ethical Hacking :

Following are the advantages of Ethical Hacking as follows.

- This helps to fight against cyber terrorism and to fight against national security breaches.
- This helps to take preventive action against hackers.
- This helps to build a system that prevents any kinds of penetration by hackers.
- This offers security to banking and financial establishments.
- This helps to identify and close the open holes in a computer system or network.

Disadvantages of Ethical Hacking :

Following are the disadvantages of Ethical Hacking as follows.

- This may corrupt the files or data of an organization.
- They might use information gained for malicious use. Subsequently, trustful programmers are expected to have achievement in this framework.
- By hiring such professionals will increase costs to the company.
- This technique can harm someone's privacy.
- This system is illegal.
- It hampers system operation

5.1.3 Skills require to become Ethical hacker

An ethical hacker finds the weak points or loopholes in a computer, web applications, or network and report them to the organization. So, let's explore the skills required to become an ethical hacker. **Basic Skills Required to Become an Ethical Hacker are** Networking Knowledge ,Skills in Linux, Skills in Programming, Reverse Engineering Basics, Cryptography knowledge, DB Knowledge, Resolving Issues…etc

1. Computer Networking Skills

One of the most important skills to become an ethical hacker is networking skills. Understanding networks like DHCP, Sub netting, and more will provide ethical hackers to explore the various interconnected computers in a network and the potential security threats that this might create, as well as how to handle those threats.

2. Computer Skills

Basic computer skills include data processing, managing computer files, and creating presentations. Advanced computer skills include managing databases, programming, and running calculations in spreadsheets. Some of the most essential computer skills are MS Office, Spreadsheets, Email, Database Management, Social Media, Web, Enterprise systems, etc. An ethical hacker needs to be a computer systems expert.

3. Linux Skills

The main reason to learn Linux for an ethical hacker is, in terms of security, Linux is more secure than any other operating system. It does not mean that Linux is 100 percent secure it has some malware for it but is less vulnerable than any other operating system. So, it does not require any anti-virus software.

4. Programming Skills

Another most important skill to become an ethical hacker is Programming Skills. Before one writes code he/she must choose the best programming language for his/her programming. Here is the list of programming languages used by ethical hackers along with where to learn these programming language.

- **Python**: Python Programming Language
- **SQL**: SQL Tutorial
- **C**: C Programming Language
- **JavaScript**: JavaScript Tutorials
- **PHP**: PHP Tutorials
- **C++**: C++ Programming Language
- **Java**: Java Programming Language
- **Ruby**: Ruby Programming Language
- **Perl**: Perl Programming Language

5. Basic Hardware Knowledge

suppose one wants to hack a machine that is controlled by a computer. First, he needs to know about the machine or how it works. Last, he has to get access to the computer that controls the machine. Now, the machine will have a very good software security system; however, hackers don't care about hardware security, so he can play with the hardware if he can access it. If one doesn't know about hardware, then how will he/she know how the motherboard works, how USBs to transfer data, or how CMOS or BIOS work together, etc.? So one must have basic hardware knowledge also to become an ethical hacker.

6. Reverse Engineering

Reverse Engineering is a process of recovering the design, requirement specifications, and functions of a product from an analysis of its code. It builds a program database and generates information from this. In software security, reverse engineering is widely used to ensure that the system lacks any major security flaws or vulnerabilities. It helps to make a system robust, thereby protecting it from hackers and spyware. Some developers even go as far as hacking their system to identify vulnerabilities – a system referred to as ethical hacking.

7. Cryptography Skills

Cryptography deals with converting a normal text/message known as plain text to a non-readable form known as cipher text during the transmission to make it incomprehensible to hackers. An ethical hacker must assure that communication between different people within the organization does not leak.

8. Database Skills

DBMS is the root of creating and managing all databases. Accessing a database where all the information is stored can put the company in a tremendous threat, so ensuring that this software is hack-proof is important. An ethical hacker must have a good understanding of this, along with different database engines and data schemas to help the organization build a strong DBMS.

9. Problem-solving Skills
Problem-solving skills help one to determine the source of a problem and find an effective solution. Apart from the technical skills pointed above, an ethical hacker also must be a critical thinker and dynamic problem solver. They must want to learn new ways and ensure all security breaches are thoroughly checked. This requires tons of testing and an ingenious penchant to device new ways of problem-solving.

5.2 Penetration testing concepts

A penetration test, also known as a pen test, is a simulated cyber attack against your computer system to check for exploitable vulnerabilities. In the context of web application security, penetration testing is commonly used to augment a web application firewall (WAF).
OR
"Penetration testing which is also known as pen-test is a part of ethical hacking, where it focuses explicitly on penetrating only the information systems."

Now, how is penetration testing different from ethical hacking?
Penetration testing focuses exclusively on information systems, while ethical hacking is a broad area to protect the systems. Ethical hacking has more job roles and responsibilities than penetration testing.

Types Of Penetration Testing:
Six main types of penetration testing are as follows:

- Network Services
- Web Application
- Client-Side
- Wireless
- Social Engineering
- Physical Penetration Testing

Penetration Testing Tools:
These are some of the more popular tools that are frequently used by hackers:

1. BeEF
2. Metasploit
3. NMAP
4. Nessus Vulnerability Scanner
5. WIRESHARK
6. SQLMap
7. BackTrack
8. John the Ripper

5.2.1 Phases of Ethical hacking

Phases of Ethical Hacking and Penetration Testing are same.
Ethical Hacking has numerous applications. The way professionals utilize these 6 phases of Ethical Hacking are discussed below-

These are:

1. **Reconnaissance:** The attacker uses various hacking tools (NMAP, Hping) to obtain information about the target
2. **Scanning:** Using tools such as NMAP and Nexpose, the attacker tries to spot vulnerabilities in the system
3. **Gain access:** Here, the attacker attempts to exploit the vulnerability using the Metasploit tool
4. **Maintain access:** Now, the attacker tries to install some backdoors into the victim's system for future access (Metasploit is used again to achieve this)
5. **Clear tracks:** In this stage, the attacker clears all evidence of the attack as no attacker likes to get caught
6. **Reporting:** Finally, the ethical hacker documents a report which consists of the vulnerabilities spotted, the tools used to exploit, and the success rate of the operation

5.2.2 Areas of penetration testing

1. Network services: It finds weaknesses and vulnerabilities in the security of the network infrastructure (for example, firewall testing)
2. Web application: Security vulnerabilities or weaknesses will get discovered in web-based applications (for example, Outlook)
3. Client-side: It finds vulnerabilities in software on a client computer, such as an employee workstation (for example, media player)
4. Wireless: This test examines all the wireless devices which are used in a corporation (for example, tablets or smartphones)
5. Social engineering: Getting confidential information by tricking an employee of the corporation to reveal such items (for example, phishing)

5.3 SQL injection

5.3.1 Concepts of SQL Injection:

SQL injection is a code injection technique that might destroy your database. SQL injection is one of the most common web hacking techniques. SQL injection is the placement of malicious code in SQL statements, via web page input.

An SQL injection attack is the execution of a malicious SQL query to alter data stored in a database or to access data without authentication or authorization. Websites or web applications using SQL databases are vulnerable to SQL injection attacks. The most common approach to launching an SQL injection attack is via user input fields. Hence, it is very important to validate data entered by users before sending it to the server.

A SQL injection attack consists of insertion or "injection" of a SQL query via the input data from the client to the application. A successful SQL injection exploit can read sensitive data from the database, modify database data (Insert/Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the content of a given file present on the DBMS file system and in some cases issue commands to the operating system. SQL injection attacks are a type of injection attack, in which SQL commands are injected into data-plane input in order to affect the execution of predefined SQL commands.

SQL injection usually occurs when you ask a user for input, like their username/userid, and instead of a name/id, the user gives you an SQL statement that you will **unknowingly** run on your database.

In general,

- SQL injection is a code injection technique that might destroy your database.
- SQL injection is one of the most common web hacking techniques.
- SQL injection is the placement of malicious code in SQL statements, via web page input.

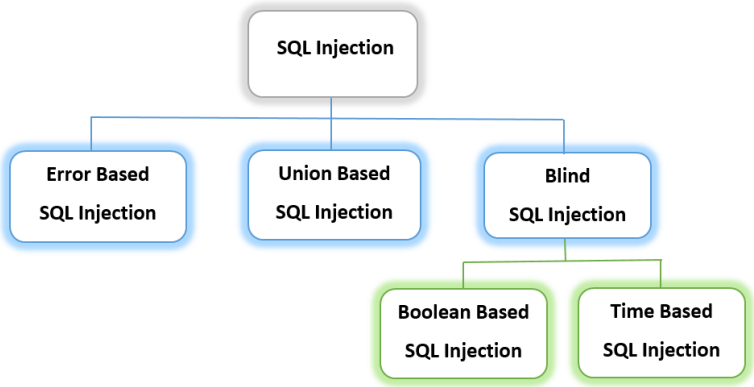
What is the impact of a successful SQL injection attack?

A successful SQL injection attack can result in unauthorized access to sensitive data, such as passwords, credit card details, or personal user information. Many high-profile data breaches in recent years have been the result of SQL injection attacks, leading to reputational damage and regulatory fines. In some cases, an attacker can obtain a persistent backdoor into an organization's systems, leading to a long-term compromise that can go unnoticed for an extended period.

5.3.2 Types of SQL Injection:

There are following types of SQL injection:

1. Error Based SQL injection
2. Union based SQL injection
3. Blind Based SQL injection



1. Error Based SQL injection:

Error-based SQL Injections obtain information about the database structure from error messages issued by the database server.

Here, an attacker tries to insert malicious query in input fields and get some error which is regarding SQL syntax or database. Attackers can use these error messages to gain information about the database. A hacker might try writing a SQL command in any input field like a single quote, double-quote, or any other SQL operator like OR, AND, NOT.

For Example, for a URL of a site that takes a parameter from the user,

then in that case: https://www.example.org/index.php?item=123

Then here attacker can try inserting any SQL command or operator in the passes value,

as: https://www.example.org/index.php?item=123′

In this case, a database could return some error. This error message gives the attacker information like the database used in SQL, the syntax that caused an error, and where the syntax occurred in the query. For a professional hacker with experience, this will be enough to tell him that the server is insecurely connected to a database and can plan additional SQL injection attacks that will cause damage.

2. Union based SQL injection:

Union-based SQL injection involves the use of the UNION operator that combines the results of multiple SELECT statements to fetch data from multiple tables as a single result set. The malicious UNION operator query can be sent to the database via website URL or user input field.

The UNION operator is used for combining 2 tables or performing 2 select queries at the same time. In union operators, they remove duplicate row or column which we try to execute at the same time.

Query:

```
SELECT EMP_ID, EMP_DOJ FROM EMP
UNION SELECT dept_ID, dept_Name FROM dept;
```

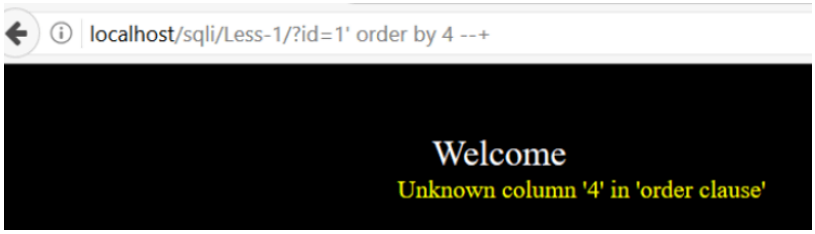
This SQL query will produce a single result set with two columns, including values from EMP columns EMP_ID and EMP_DOJ and dept columns dept_ID and dept_Name.

Two important needs must be met for a UNION query to function:

- Each query must return the same number of columns.
- The data types must be the same, i.e., it is not changed after query execution.

To determine the no of columns required in an SQL injection UNION attack, we will Inject a sequence of ORDER BY clauses and increment the provided column index until an error is encountered. " - -" at the end ignores all subsequent statements.

- ?id=1' order by 1 --+ no error
- ?id=1' order by 2 --+ no error
- ?id=1' order by 3 --+ no error
- ?id=1' order by 4 --+ we get error

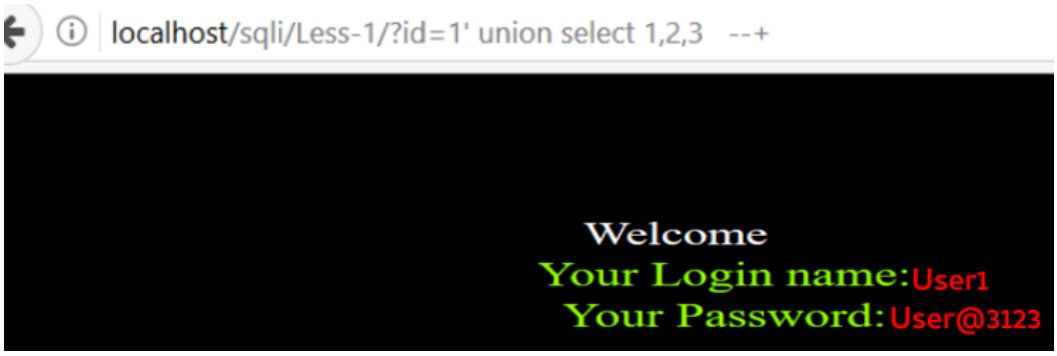


This demonstrates that the query lacks the fourth column. So we now know that the query in the backend has three columns.

Now we will use the UNION statement in order to join two queries and to be able to discover the vulnerable columns.

Query:

?id=1‘ UNION SELECT 1,2,3 --+



3. Blind SQL injection:

It is also referred to as Inferential SQL Injection. This is a type of SQL injection where we don’t have a clue as to whether the web application is vulnerable to injection attack or not.

The attacker sends data payloads to the server and observes the response and behavior of the server to learn more about its structure. This method is called blind SQLi because the data is not transferred from the website database to the attacker, thus the attacker cannot see information about the attack in-band.

Blind SQL injections rely on the response and behavioral patterns of the server so they are typically slower to execute but may be just as harmful.

Blind SQL injections can be classified as follows:

(a) Boolean-based SQL Injection:

It works by submitting a SQL query to the database and forcing the application to produce a different response depending on whether the query returns TRUE or FALSE. Only correct queries show the result, wrong queries do not return anything. Attackers should try to generate logically correct queries.

Example:

The attacker will use blind SQL injection to ensure that the inject query returns a true or false result.

?id=1' AND 1=1 --+

Now, the database checks if 1 is equal to 1 for the supplied condition. If the query is legitimate, it returns TRUE;

(b) Blind Time-Based SQL Injections:

Time-based SQL Injection works by sending a SQL query to the database and forcing it to wait for a predetermined length of time (in seconds) before answering. The response time will tell the attacker if the query result is TRUE or FALSE.

Depending on the outcome, an HTTP response will either be delayed or returned immediately. Even though no data from the database is returned, an attacker can determine if the payload used returned true or false. Because an attacker must enumerate a database character by character, this attack is often slow (particularly on big databases).

5.3.3 Case study of SQL Injection:

Case Study 1:

For this SQL injection example, let's use two database tables, Users and Contacts. The Users table may be as simple as having just three fields: ID, username, and password. The Contacts table has more information about the users, such as UserID, FirstName, LastName, Address1, Email, credit card number, and security code.

The Users table has information used for logins like:

1. jsmith,P@$$w0rd
2. sbrown,WinterIsComing!
3. kcharles,Sup3rSecur3Password$

When someone wants to log in, they'll go to the login page and enter their username and password. This information is then sent to the webserver, which will construct a SQL query and send that query to the database server. An example of what that query looks like might be:

Select ID from Users where username='jsmith' and password='P@$$w0rd'

The way SQL works is that it will then perform a true or false comparison for each row that the query requests. In our example, the query says to check the Users table and give back the ID value for every row where the username is jsmith and the password is P@$$w0rd. Often, the webserver will then see what is returned by the database server and if it is a number. In our case, the webserver would receive back a 1 and let the user past the login page.

But, what if we want to get malicious with this? Because the database server performs that true-or-false check, we can trick it into believing that we have successfully authenticated. We can do this by adding an OR to the password. If we log in with x' or 1=1 as our password, that will create a new SQL query that looks like:

Select ID from Users where username='jsmith' and password='x' or 1=1

This will work for us, because while x is not jsmith's password, the database server will then check the second condition. If x isn't jsmith's password, then does 1 equal 1? It does! The ID will be sent back to the application and the user will be successfully authenticated.

Another technique we can use for blind SQL injection, the one where no data is sent back to the screen is to inject other hints. Similar to our ' or 1=1 condition, we can tell the server to sleep. We could add: " ' or sleep(10) " and this will do what it seems like. It will tell the database server to take a 10-second nap and all responses will be delayed.

Case Study 2:

An attacker wishing to execute SQL injection manipulates a standard SQL query to exploit non-validated input vulnerabilities in a database. There are many ways that this attack vector can be executed, several of which will be shown here to provide you with a general idea about how SQLI works.

For example, the above-mentioned input, which pulls information for a specific product, can be altered to read http://www.estore.com/items/items.asp?itemid=999 or 1=1.

As a result, the corresponding SQL query looks like this:

```
SELECT ItemName, ItemDescription

FROM Items

WHERE ItemNumber = 999 OR 1=1
```

And since the statement 1 = 1 is always true, the query returns all of the product names and descriptions in the database, even those that you may not be eligible to access.

Attackers are also able to take advantage of incorrectly filtered characters to alter SQL commands, including using a semicolon to separate two fields.

For example, this input http://www.estore.com/items/iteams.asp?itemid=999; DROP TABLE Users would generate the following SQL query:

```
SELECT ItemName, ItemDescription
FROM Items
WHERE ItemNumber = 999; DROP TABLE USERS
```

As a result, the entire user database could be deleted.

Another way SQL queries can be manipulated is with a UNION SELECT statement. This combines two unrelated SELECT queries to retrieve data from different database tables.

For example, the input http://www.estore.com/items/items.asp?itemid=999 UNION SELECT user-name, password FROM USERS produces the following SQL query:

```
SELECT ItemName, ItemDescription
FROM Items
WHERE ItemID = '999' UNION SELECT Username, Password FROM Users;
```

Using the UNION SELECT statement, this query combines the request for item 999's name and description with another that pulls names and passwords for every user in the database.

Another example,

Suppose we have an application based on student records. Any student can view only his or her own records by entering a unique and private student ID. Suppose we have a field like below: **STUDENT_ID:**

And the student enters the following in the input field:

12222345 or 1=1.

So this basically **translates to :**

SELECT * from STUDENT where

STUDENT_ID == 12222345 or 1 = 1

Now this **1=1** will return all records for which this holds true. So basically, all the student data is compromised. Now the malicious user can also delete the student records in a similar fashion.
Consider the following SQL query.

SELECT * from USER where

USERNAME = "" and PASSWORD=""

Now the malicious can use the '=' operator in a clever manner to retrieve private and secure user information. So instead of the above-mentioned query the following query when executed, retrieves protected data, not intended to be shown to users.

Select * from User where

(Username = "" or 1=1) AND

(Password="" or 1=1).

Since **1=1** always holds true, user data is compromised.

Preventing SQL Injection

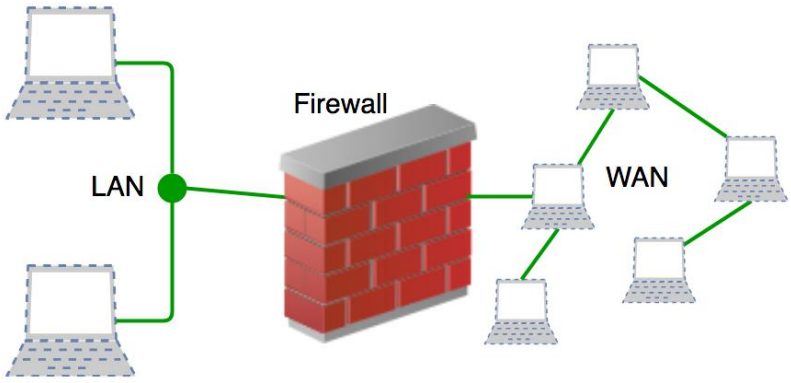
- User Authentication: Validating input from the user by pre-defining length, type of input, of the input field and authenticating the user.
- Restricting access privileges of users and defining as to how much amount of data any outsider can access from the database. Basically, user should not be granted permission to access everything in the database.
- Do not use system administrator accounts.

5.4 Firewall:

5.4.1 Concepts of Firewall :

With the increasing number of cyber crimes with every passing day, individuals and companies must secure their information. A firewall is one such security device that can help you safeguard your network and device from an outsider.

A firewall can be defined as a special type of network security device or a software program that monitors and filters incoming and outgoing network traffic based on a defined set of security rules. It acts as a barrier between internal private networks and external sources (such as the public Internet).



Firewalls prevent unauthorized access to networks through software or firmware. By utilizing a set of rules, the firewall examines and blocks incoming and outgoing traffic. It is a cyber security tool that filters network traffic and helps users block malicious software from accessing the Internet in infected computers.

Fencing your property protects your house and keeps trespassers at bay; similarly, firewalls are used to secure a computer network. Firewalls are network security systems that prevent unauthorized access to a network. It can be a hardware or software unit that filters the incoming and outgoing traffic within a private network, according to a set of rules to spot and prevent cyber attacks.

Firewalls are used in enterprise and personal settings. They are a vital component of network security. Most operating systems have a basic built-in firewall. However, using a third-party firewall application provides better protection.

5.4.2 Types of Firewall

Depending on their structure and functionality, there are different types of firewalls. The following is a list of some common types of firewalls:

(a) Proxy Firewall

(b) Packet-filtering firewalls

(c) Stateful Multi-layer Inspection (SMLI) Firewall

(d) Unified threat management (UTM) firewall

(e) Next-generation firewall (NGFW)

(f) Network address translation (NAT) firewalls

(a) Proxy Firewall:

It filter network traffic at the application level. It acts an intermediary between two end systems. The client must send a request to the firewall, where it is then evaluated against a set of security rules and then permitted or blocked. Most notably, proxy firewalls monitor traffic for layer 7 protocols such as HTTP and FTP, and use both stateful and deep packet inspection to detect malicious traffic.

(b) Packet-filtering firewalls:

It is the most common type of firewall. It examines packets and prohibit them from passing through if they don't match an established security rule set. This type of firewall checks the packet's source and destination IP addresses. If packets match those of an "allowed" rule on the firewall, then it is trusted to enter the network. Packet-filtering firewalls are divided into two categories: stateful and stateless.

(c) Stateful Multi-layer Inspection (SMLI) Firewall:

It filter packets at the network, transport, and application layers, comparing them against known trusted packets. SMLI also examine the entire packet and only allow them to pass if they pass each layer individually. These firewalls examine packets to determine the state of the communication (thus the name) to ensure all initiated communication is only taking place with trusted sources.

(d) Unified threat management (UTM) firewall:

A UTM device generally integrates the capabilities of a stateful inspection firewall, intrusion prevention, and antivirus in a loosely linked manner. It may include additional services and, in many cases, cloud management. UTMs are designed to be simple and easy to use.

(e) Next-generation firewall (NGFW):

It combines traditional firewall technology with additional functionality, such as encrypted traffic inspection, intrusion prevention systems, anti-virus, and more. Most notably, it includes deep packet inspection (DPI). While basic firewalls only look at packet headers, deep packet inspection examines the data within the packet itself, enabling users to more effectively identify, categorize, or stop packets with malicious data.

(f) Network address translation (NAT) firewalls:

It allows multiple devices with independent network addresses to connect to the internet using a single IP address, keeping individual IP addresses hidden. As a result, attackers scanning a network for IP addresses can't capture specific details, providing greater security against attacks. NAT firewalls are similar to proxy firewalls in that they act as an intermediary between a group of computers and outside traffic.

5.4.3 How does a firewall work?

A firewall system analyzes network traffic based on pre-defined rules. It then filters the traffic and prevents any such traffic coming from unreliable or suspicious sources. It only allows incoming traffic that is configured to accept.

Typically, firewalls intercept network traffic at a computer's entry point, known as a port. Firewalls perform this task by allowing or blocking specific data packets (units of communication transferred over a digital network) based on pre-defined security rules. Incoming traffic is allowed only through trusted IP addresses, or sources.

Advantages of Firewall:

- It provides enhanced security and privacy from vulnerable services. It promotes privacy.
- It prevents unauthorized users from accessing a private network that is connected to the internet.
- Firewalls are designed to protect the computer from viruses, malware, and other harmful codes.

- Firewalls provide faster response time and can handle more traffic loads. It monitors network traffic.
- A firewall allows you to easily handle and update the security protocols from a single authorized device.
- It safeguards your network from phishing attacks.

Disadvantages of Firewall:

1. **Cost Oriented**

   It can be costly for organizations as they need to pay for them. If they look for hardware firewalls, then it will cost more for them. As there are installation charges, maintenance charges and also they need to hire the IT technician for this. The cost also varies on the type of firewall the company chooses.

2. **It can restrict some organizational activities.**

   The firewall prevents access to several sites that have malware or any virus. This thing can be good for its users, but large companies often face problems because of it. As the firewalls use strict security guidelines and that can affect the employees' work efforts. And hence it will also impact the productivity of the company by which it can face the loss.

3. It can decrease performance level.

   Firewalls are the security tools that keep running in the background of the computer. And as there are multiple tabs open and the firewall is also running, in that case, the performance of the computer will be slow.

4. **Still, Some Hacking Attacks Can Happen**

   The firewalls can be effective on the basic trojan and their types. Hence, another type of malware can enter the computer device. If you have a firewall installed on your computer, then you should also install the anti-virus applications. So that you can run the malware detection test and remove all these malware and viruses.

5. **Need A Careful Maintenance**

   For many large businesses, it needs to have a dedicated team of IT experts who can maintain all the maintenance work of firewalls. Hence, it will increase the company's capital expenditure, and they will need to accommodate all these employees. And they need to work as per the latest policies provided by a company that can be challenging to execute. Hence, the maintenance issue is also one of the significant disadvantages of firewalls.