# SCENARIO

Recently, my company was hired to carry out a penetration test for a **medium sized medical clinic** containing **60 employees** with many branches across Sri Lanka. The clinic offers a variety of online services including checking online lab reports, scheduling appointments and checking availability of doctors.

The clinic **web application** allows patients to view their reports online and schedule appointments, so it handles **sensitive information like patient data, history and transactions**. Also, the staff attendance monitoring and employee data is also contained in the application due to payroll and HR management. It is critical that the clinic complies with mandatory healthcare rules and regulations like HIPAA due to the personal and sensitive information it contains.

The clinic contains **2 types of users: staff and patients**. Staff members use the clinic’s application to monitor appointments and publish patient/lab reports. Patients can access the web application to schedule appointments, view reports and check the availability of doctors.

# PART A – INFORMATION GATHERING

## A.1 OSINT Activities

### A.1.1 OSINT Activity 1 – WHOIS

The author utilized WHOIS which is a protocol for querying and retrieving information about domain names and IP addresses. Below illustrations shows the command used and the results obtained by using this utility.

Command Used – whois cwscenario.site

A computer screen shot of a computer program

Description automatically generated

Figure 1: WHOIS results

The results obtained from WHOIS for the cwscenario.site reveals private information such as domain status, registration expiry and other name servers. This information can be used to identify potential vulnerabilities in the infrastructure and plan attack strategies. Some of the sensitive information gathered are listed below:

* Registrar URL (https://ionos.com)
* Creation date – 2021-01-07, Registry expiry date – 2025-01-07
* Domain status
* Registrant Country – GB
* Registrant Province – GLS
* Name servers
* Registrar emails, phone numbers

They can provide valuable insights to attackers which can be used to plan and execute various types of malicious activities such as: phishing, social engineering, DNS manipulation and cybercrimes.

### A.1.2 OSINT Activity 2 – TheHarvester

The author utilized TheHarvester which is a tool for querying and retrieving passive information mainly about domain names, hosts and IPs. Below illustration shows the command used and the results obtained by using this utility.

Command Used - theHarvester -d cwscenario.site -b all

A screenshot of a computer program

Description automatically generated

Figure 2: TheHarvester results

The results obtained from TheHarvester for the cwscenario.site reveals private information such as ASNS, URLs, IPs and hosts. This information can be used for various harmful purposes. Some of the sensitive information gathered are listed below:

* ASNS
* 3 IP addresses
* 12 Hosts information

The information gathered like the ASNS can be used to identify the internet service provider and the IP addresses obtained could be specifically targeted for cybercrimes and host information could be used to identify entry points for attacks. Attackers can use this to plan and execute various types of malicious activities such as: network attacks, data breaches, and service disruptions.

### A.1.3 OSINT Activity 3 – dnsenum

The author utilized dnsenum to extract information about DNS records and locating DNS servers. Below illustration shows the command used and the results obtained by using this utility.

Command Used - dnsenum --enum cwscenario.site

A screenshot of a computer

Description automatically generated

Figure 3: The results for dnsenum (1/2)

A screenshot of a computer

Description automatically generated

Figure 4: The results for dnsenum (1/2)

The results obtained from dnsenum for the cwscenario.site reveals private information such as Host addresses, Name servers and IPs. This information can be used for various harmful purposes. Some of the sensitive information gathered are listed below:

* 4 name servers
* 2 mail servers
* Host information
* IPs

The information gathered via DNS Enumeration revealed sensitive information such as IPs, name servers, mail servers and host information. However, performing reverse lookup was denied which suggests the implementation of a security measure which denied a zone transfer.

### A.1.4 How OSINT Can be Effective and Why it is the First Activity Carried Out

OSINT stands for Open-Source Intelligence which is used to collect information from public sources. It involves acquiring data and gathering information from sources available publicly like websites, social media, public databases and etc. The data gathered can be used for multiple purposes such as market research, data analysis and investigating competition. On the other hand, it is a critical tool used for cybercrimes to plan and execute targeted attacks on organizations (Titterington, 2023).

Some of the ways in which OSINT activities can be **effective** are listed below:

1. Target Identification
2. Attack Surface Enumeration
3. Vulnerability Assessment
4. Phishing

It is often the **first step carried out by penetration testers** because by the effectiveness mentioned above, it can be used to understand the target environment and gather information about the infrastructure of an organization. Also, helps identify potential attack entry points during the assessment such as exposed systems and network configurations. So these activities help penetration testers identify weaknesses and identify breaches in industry regulations and mitigate risks which will contribute to a comprehensive testing process (Pritchard, 2020).

### A.1.5 Scenario Assessment

|  |  |
| --- | --- |
| **Information Obtained** | **How Dangerous it can be** |
| Registrar URL | Knowing the registrar URL, an attacker could impersonate communications from the registrar and deceive clinic staff into providing sensitive information. |
| Creation date and registry expiry date | This could be used to plan attacks in the long-term by strategically planning based on the duration of the domain of the clinic |
| Domain status | Attackers could exploit the domain status by impersonating and hijacking the domain with the exposed domain information. |
| Registrant country and province | Attackers could use this for social engineering attacks and phishing purposes by exploiting regional vulnerabilities near the clinic area and location. |
| Registrar emails and phone numbers | An attacker could impersonate as an employee from the medical clinic and try to gain unauthorized access. |
| Name servers | Knowledge of the name servers could lead to targeting DNS infrastructure for hijacking or DDoS attacks which could disrupt clinic operations. |
| IPs and Host information | This exposure could be used to find potential entry points for attack for the clinic which could lead to the loss of personal patient information. |

Figure 5: Scenario assessment (OSINT)

## A.2 Reconnaissance

Reconnaissance is a crucial step in penetration testing which involves gathering and locating confidential information such as interacting with open ports, running services, etc (Khan, 2022). The reconnaissance methods utilized in this study are shown below.

**Machine and Corresponding IPs**

|  |  |  |
| --- | --- | --- |
| Kali Linux (Attacker) | OWASP (Vulnerable Server/ Victim) | Windows (Victim) |
| 192.168.56.100 | 192.168.56.101 | 192.168.56.102 |

Table 1: Virtual machine IPs

### A.2.1 Reconnaissance Activity 1 - DirBuster

The author utilized DirBuster to extract information by brute force about searching for a list of files and directory in a web server by giving a text file which contains the target list of words. Below illustration shows the commands used and the results obtained by using this utility.

Commands Used  
touch folderbuster.txtmousepad folderbuster.txt  
dirbuster

A screenshot of a computer

Description automatically generated

Figure 6: DirBuster setup (1/2)

A screenshot of a computer

Description automatically generated

Figure 7: DirBuster setup (2/2)

A screenshot of a computer

Description automatically generated

Figure 8: DirBuster results

OWASP bricks, jquery files, images and multiple JavaScript files were found using brute force on the server URL. Various directories were identified as well discovering hidden and different directories and file names. By identifying vulnerable and unprotected directories, attackers can use this to identify weak points allowing access to directories without authorization. Also, discovering hidden files could be used to gather intelligence and target attacks while stealing sensitive data. Then, configurations could be exploited from the misconfigured directories using the information gathered.

### A.2.2 Reconnaissance Activity 2- Active Reconnaissance

Sometimes by simply looking at the code of the page source, some information could be exposed. The author utilized active reconnaissance by ‘View page source’. Below illustration shows the results obtained by using this utility

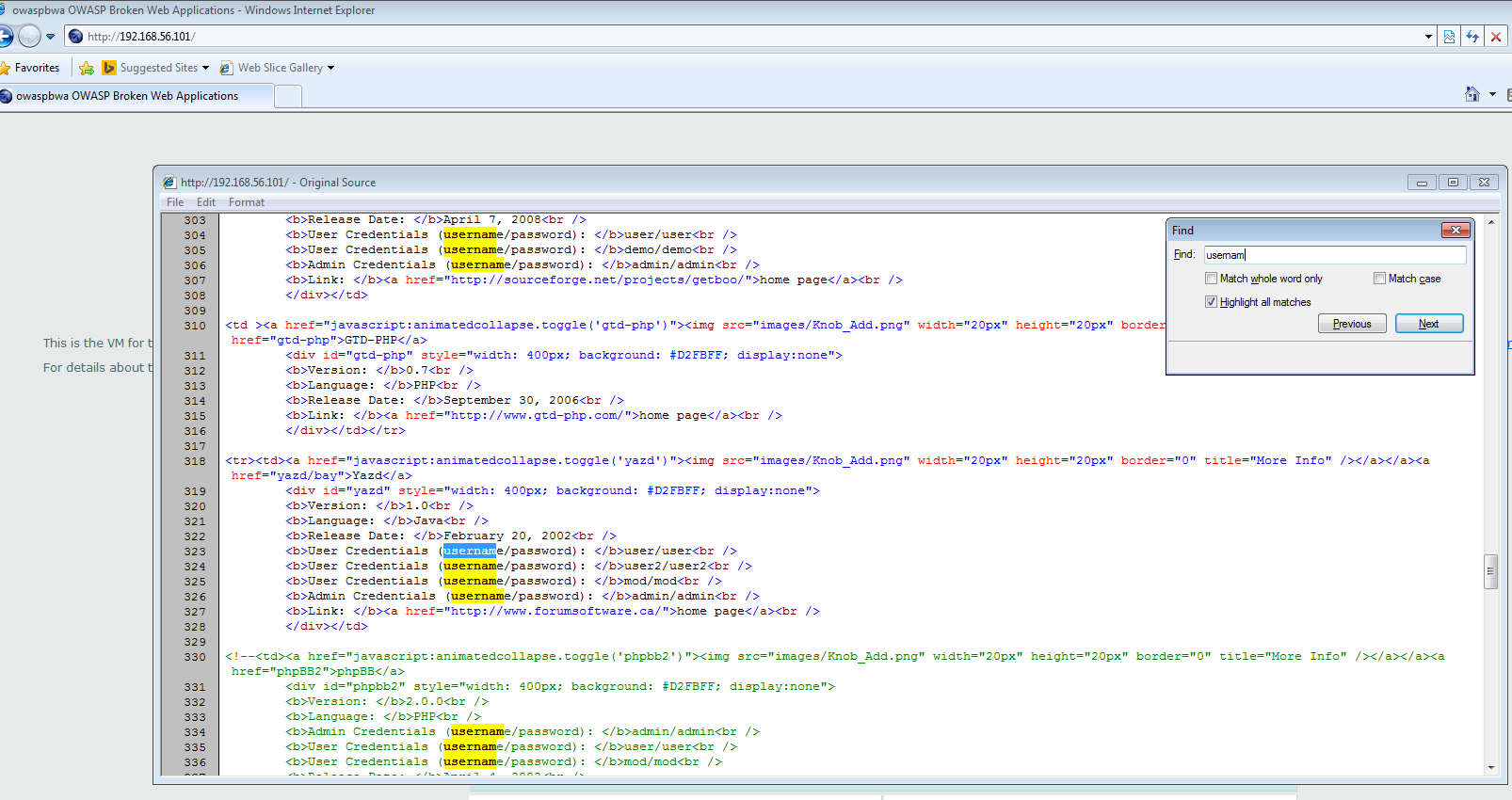


Figure 9: View page source code

Here you can see two crucial information visible by just inspecting the page source:

* Usernames: user, demo, admin
* Corresponding passwords: user, demo admin

This information could be easily used by attackers to gain unauthorized access to any system and word lists and workaround can be used to update these usernames and passwords to whichever credential the attacker wants.

### A.2.3 Reconnaissance Activity 3 -robots.txt

The author utilized vicnum which contains web apps based on games. The author utilized this method by trying to access to robots.txt file of this site to find files and directory that may not be directly linked to the main application.

A screenshot of a computer

Description automatically generated

Figure 10: Access robots.txt

This shows that the /jotto/ and /cgi-bin/ is disallowed for all user agents. However, may be vulnerable to exploitation and can be checked by trying to access that directory.

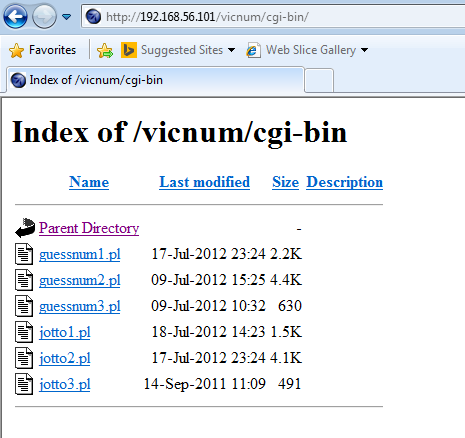


Figure 11: Accessing cgi-bin

A screenshot of a computer

Description automatically generated

Figure 12: Accessing jotto

A screenshot of a computer

Description automatically generated

Figure 13: Exploited jotto answers

By seeing the robots.txt we can see the important files for the web application. By indexing and accessing jotto and cgi-bin directories, unauthorized access was gained and the answers were obtained using this approach.

### A.2.4 Scenario Assessment

The DirBuster revealed information such as OWASP bricks, image directories, multiple JS and jQuery files. For a clinic it could lead to identifying the infrastructure of directories and could be used to exploit potential vulnerabilities. Also, if image directories are exposed that would compromise sensitive patient data and lead to reputation damage as it violates compliance regulations. Discovery of OWASP bricks could also lead to security flaws and attacks of the clinic’s web services.

Looking at the code directly and having exposed credentials like the examples could lead to an attacker stealing those credentials and using it for unauthorized access. So, the medical services of a clinic could be disrupted due to a system breach and affect the overall reputation and trust of the clinic. Leaked credentials could also lead to: identity theft, financial losses and reputational damage.

Accessing the robots.txt file revealed a way to access directories which cannot normally be browsed through a web application directly. So, if the medical clinic has an exposed or unprotected robots.txt file it could be used by attackers to exploit the web services of the clinic in several ways such as: directory enumeration and access, discovering vulnerabilities, gathering information and identifying entry points visible through the robots.txt like the /cgi-bin/ and /jotto/ directories. So, accessing directories in an unauthorized way will compromise patient data and leak sensitive directories and files of the clinic.

## A.3 Port Scanning and Enumeration

### A.3.1 Ports Identified

A screenshot of a computer

Description automatically generated

Figure 14: Investigating open ports and OS

A computer screen shot of a computer

Description automatically generated

Figure 15: Investigating remote access port

A computer screen shot of a computer

Description automatically generated

Figure 16: Identifying all machines on the network

Nmap or ‘Network Mapper’ is a tool used to provide insights about the network architecture and was utilized by the author. It helps in port scanning, service enumeration, OS detection and network mapping among many. The ports identified through this tool which are used by the server are listed below.

1. **Port 22/tcp (SSH)** – If an attacker finds this port to be open, they will try to gain unauthorized access through weak credentials and steal SSH keys and try to breach the vulnerability.
2. **Port 80/tcp (HTTP)** – An open HTTP port could lead to brute force attacks and DDoS attacks for running services on this open port.
3. **Port 143/tcp (IMAP) –** An attacker may intercept emails and try to gain unauthorized access from weak authentication through open services on this port.

Some of the other open ports identified include 139/tcp (NetBIOS-SSN), 445/tcp (Microsoft-DS), 8080/tcp (HTTP-proxy), 443/tcp (HTTPS), 5001 and 8081 ports. So having such open ports increase risk of unauthorized access and its best practice for any business or company with an online presence to minimize their open port to reduce the risks associated with open ports.

### A.3.2 Open Ports and Threats

By definition, an open port refers to TCP or UDP port number that is configured to accept packets. Meanwhile, a port that declines connections or processes all packets yet drops them is a closed one (Tunggal, 2024). Having a port open is similar to leaving an open door to enter and exit a system that allows network communication flow to come in and out. This raises the possibility of potential dangers that comes with a port being open as it could be used as an entry point by an attacker (Schrader, 2022).

Some threats that can be caused by having an open **Port 22/tcp (SSH)** include gaining unauthorized access from weak credentials or attempts to steal SSH keys that may compromise the system and lose control (Einorytė, 2024).

Additionally, open **HTTP ports such as 80/tcp** could lead to brute force attacks and DDoS attacks by using the port as a launching point. This is where attackers can overwhelm the application listening on open ports with large amounts of traffic and disrupt the system (Chappell, 2022).

So, attackers scan for open ports to find potential vulnerabilities and exploits. By accessing the services and versions of a machine, attackers carry out mapping and find vulnerabilities in a system. So for that purpose, attackers rely on open ports which are publicly accessible to analyze the infrastructure of the services on a machine (Tunggal, 2024).

### A.3.3 Scenario Assessment

|  |  |
| --- | --- |
| **Port Identified** | **Applicability to Scenario** |
| Port 22/tcp (SSH) | Gaining unauthorized access via this port or through stolen SSH keys could lead to the attacker gaining unauthorized access. This would compromise the system security and lead to a breach of patient and employee information which is personal. |
| Port 80/tcp (HTTP) | A brute force or DDoS attack carried out on this open port could lead to overwhelming traffic for the medical clinic application. This could lead to patients or potential customers not being able to access the application which threatens the availability. |
| Port 143/tcp (IMAP) | An attacker may intercept emails sent to the medical clinic via this port. This would compromise sensitive information shared between the clinic and its patients. It could also disrupt the communication service if exploited through weak authentication and credentials. |

Table 2: Ports applicability to scenario

Overall, it is crucial for medical clinics to minimize the number of open ports it has in their web application and integrate strong security measures in place. Also constantly, updating the software and web application with patches and recent security updates would help mitigate risks associated with compromising patient data and operations within a clinic.

# PART B – SERVER-SIDE EXPLOTS

B.1 Data Tampering

A screenshot of a computer

Description automatically generated

Figure 17: Application landing page

The login page of the site which data tampering is attempted is shown above. A tool called OWASP mantra will be used for this purpose to see if the data can be intercepted and exploited if possible.

A screenshot of a computer

Description automatically generated

Figure 18: Intercepted request

Above illustration shows the intercepted request before submission where the data can be manipulated such as the username and password making it vulnerable. Therefore this could be used to bypass invalid credentials with valid ones.

A screenshot of a computer

Description automatically generated

Figure 19: Manipulated request

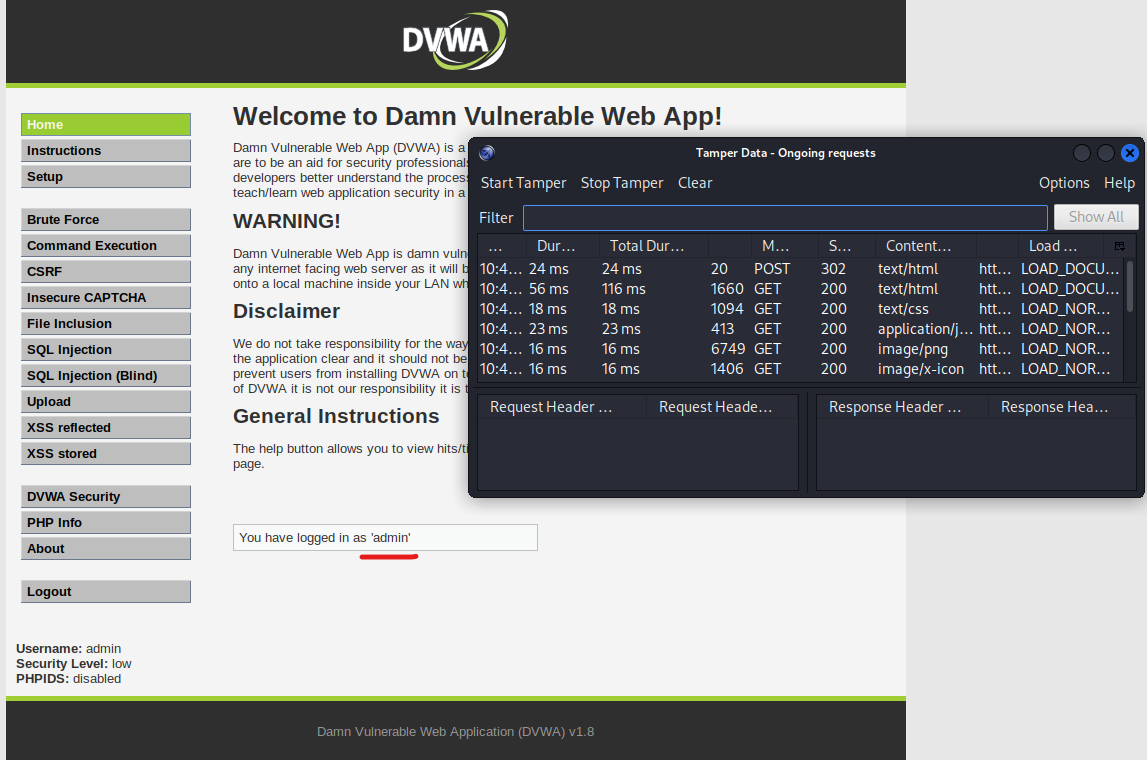


Figure 20: Successful login via manipulation

This proves that there is a major vulnerability in the sites login where unauthorized access can be gained by manipulating the POST request sent for a user to login to the site. By intercepting and changing the POST request values, data tampering and exploitation can be done.

B.1.2 Research

Data tampering vulnerability is a major weakness which allows the modification of data through unauthorized access (Kapsamer, 2022). Some of the key impacts it leads to include:

1. Data compromise and financial loss
2. Leakage of sensitive information
3. Unidentified attacker or hard to trace

Therefore, these key impacts could lead to overall reputational damage of an organization and should be minimized by applying strict security measures (Awati, 2022).

This breaches the cyber security tenet of **integrity** of the CIA triad because it modifies the accuracy of data initially set or sent through unauthorized entities.

B.1.3 Scenario Assessment

In context of our medical clinic, tampered data could be used to gain unauthorized access to the clinic’s web application. This breach could lead an attacker to access personal patient data through tampered credentials.

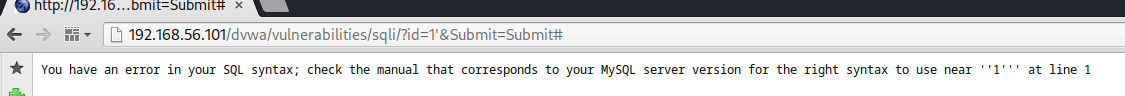
Attackers also may modify the details of appointments which were scheduled. This could lead to a risk of the patient’s health as an appointment may be urgent for a patient. Also, by gaining unauthorized access, the hospital data and lab reports will be compromised affecting the overall reputation of the clinic and violating the compliance regulations.

Since the medical clinic relies on the web application to schedule appointments and publish lab reports, data tampering could introduce incorrect doctor details and reports violating the integrity of the available data through the application. This could mislead patients and impact their appointment scheduling and payment.

B.2 SQL Injection

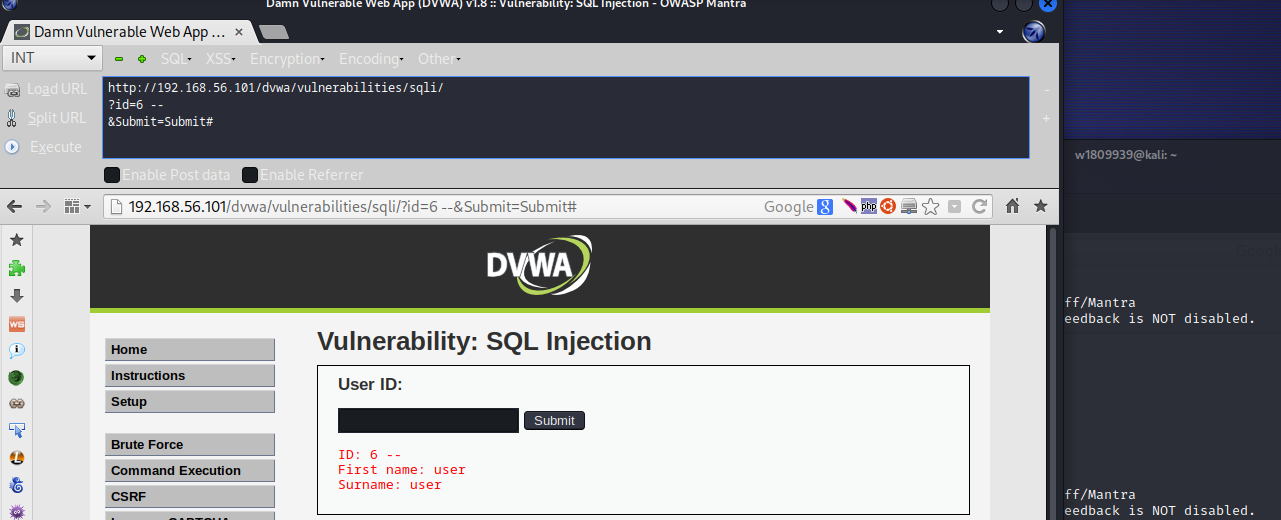
A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generated

B.3 XSS Scripting

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

B.4 Other Vulnerabilities

B.5 Cryptanalysis Attack

We can demonstrate a cryptanalysis attack from this challenge using the security shepherd tool.

A screenshot of a computer

Description automatically generated

Figure 21: Cryptanalysis challenge

From the given ciphered text, we can use external tools such as: <https://raw.org/tool/caesar-cipher/> to decrypt the ciphered text as shown below.

A screenshot of a computer

Description automatically generated

Figure 22: External tool for decryption

By selecting different keys, we can arrive with the decrypted message. In this by choosing the key as 21, it said that t he result key is given in the next line so then the next line could be attempted to decrypt as shown below.

A screenshot of a computer

Description automatically generated

By selecting ‘guess’, it seemed to give a meaningful string which could be used in the next step to see if it is the result key.

A screenshot of a computer

Description automatically generated

Figure 23: Result key

The result key was successfully figured through using an external tool such as shown in the above illustrations. Therefore, if we apply the same concept to a real-world scenario where encrypted usernames and passwords are stored and retrieved through SQL injections and data gathering phases then we could decrypt the password using this external tool. Such an example is shown below.

* Encrypted password retrieved – ‘cnffjbeq’’

A screenshot of a computer

Description automatically generated

Figure 24: Decrypting a password

This shows a means by which cryptanalysis attacks can be conducted and passwords can be retrieved via deciphering text retrieved.

B.5.1 Scenario Assessment

Cryptanalysis attacks could lead to the exposure of encrypted data in the medical clinic where an attacker could compromise confidentiality and lead to data breach of patients using the decrypted credentials.

This violates the confidentiality tenet of the CIA triad as the attacker could gain unauthorized access via the decrypted data gathered.

PART C – CLIENT-SIDE EXPLOITS

C.1 Man in the Middle Attack (MiTM)

C.1.1 Scenario Assessment

C.2 Social Engineering Attack

C.2.1 Scenario Assessment

PART D – DENIAL OF SERVICE ATTACKS

PART E – RECOMMENDATIONS TO PROTECT THE SCENARIO COMPANY SERVER