Sri Lanka Institute of Information Technology

Cyber Security Assignment (2025)

Broken Access Control Vulnerabilities

Bug Bounty Report- 04

IT23363366

TABLE OF CONTENT

1. Title

Report Title: Broken Access Control Vulnerabilities
Reported By: Raahim Mahmooth
Tested On: https://www.payhere.lk/
Platform: https://bugzero.io

2. Scope & Objective

This penetration testing assessment was performed on the website **https://www.payhere.lk/** with a primary focus on identifying potential **Broken Access Control** vulnerabilities. The objective of this test was not to definitively confirm the existence of vulnerabilities, but to evaluate the access control mechanisms and assess their resilience to unauthorized access. As part of the assessment, I utilized the information and scope provided by the related website via the Bug Bounty platform to guide the testing methodology.

3. Enumeration and Reconnaissance

3.1 Tools Used

- **GoBuster**
- **Burp Suite**
- **Dirbuster**

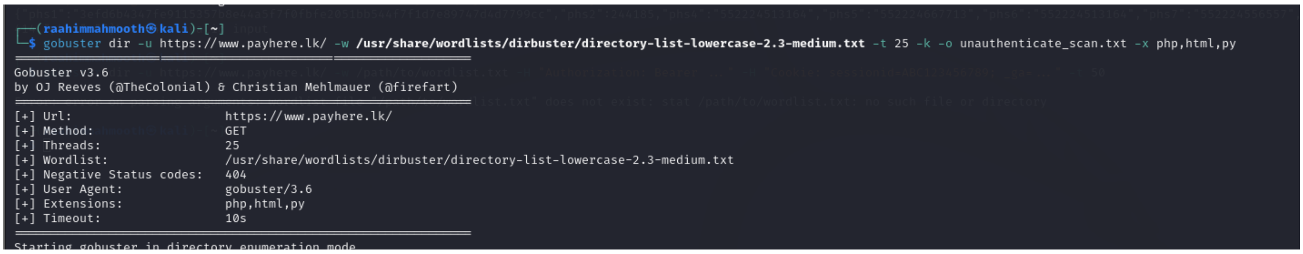
3.2 Steps Taken for Information Gathering & Enumeration

The enumeration process involved **scanning both authenticated and unauthenticated directories to identify potential sensitive files or resources** that might be accessible without proper authorization.

3.2.1 Unauthenticated Directory Enumeration & Sensitive File Enumeration:

To scan for directories and files that could potentially be accessed without authentication, I used the GoBuster tool with the following parameters:

```
gobuster dir -u https://www.payhere.lk/merchant/ -w /path/to/wordlist.txt -t 1 -k -o
unauthenticated_scan.txt -x php,html,py
```



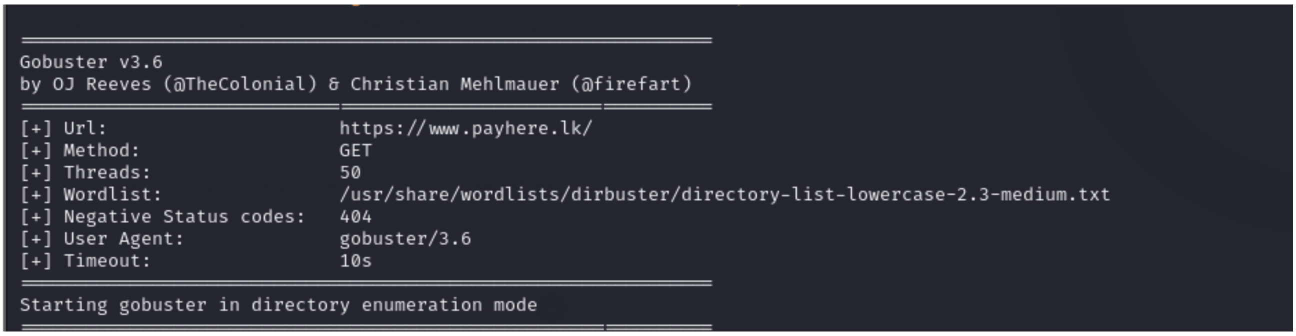
```
┌──(raahimmahmooth㉿kali)-[~]
└─$ gobuster dir -u https://www.payhere.lk/ -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt -t 25 -k -o unauthenticate_scan.txt -x php,html,py
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     https://www.payhere.lk/
[+] Method:                  GET
[+] Threads:                 25
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,html,py
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
```

The scan results were analyzed to identify any accessible sensitive files or directories.

3.2.2 Authenticated Directory Enumeration & Sensitive File Enumeration:

In order to test access to resources that require authentication, I re-ran the enumeration using GoBuster **after logging in**. This involved setting the proper `Authorization` headers as shown.

```
gobuster dir -u https://www.payhere.lk/ -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt -H "Authorization: Bearer <Your Token>" -H "Cookie: _ga=...; _fbp=...;" -t 50
```

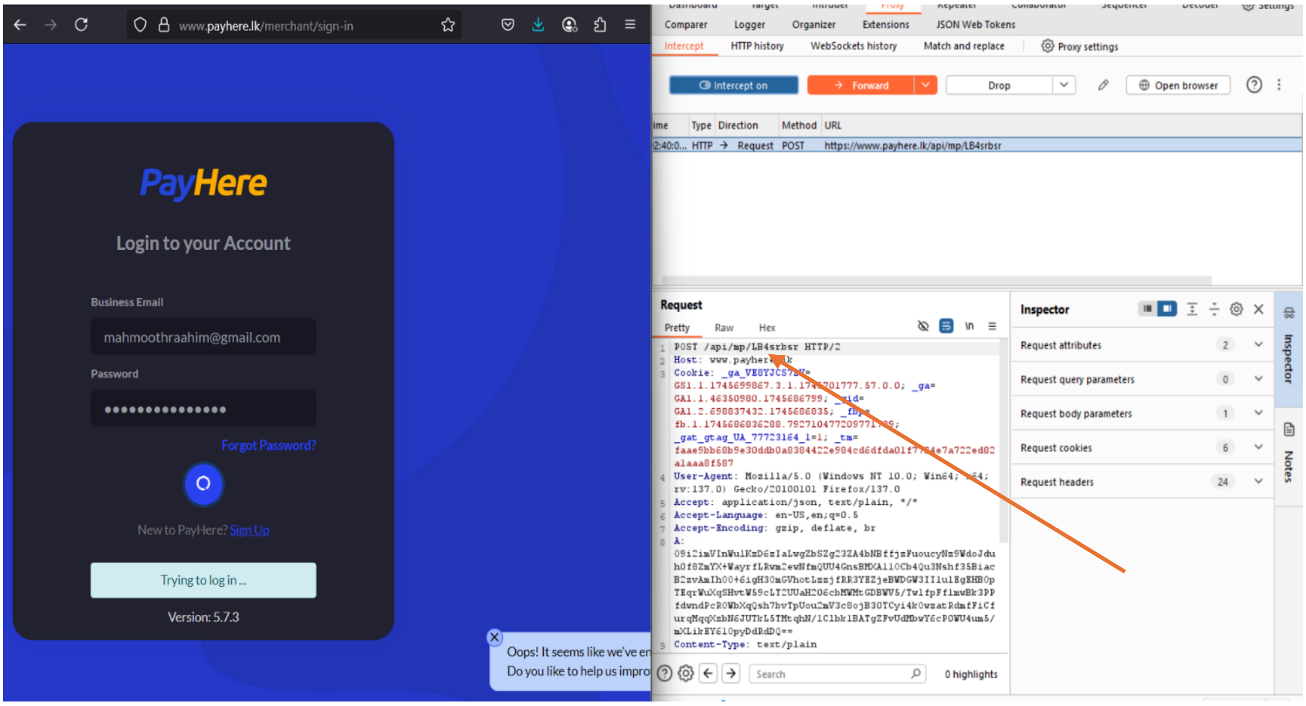


```
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     https://www.payhere.lk/
[+] Method:                  GET
[+] Threads:                 50
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
```

This approach is to identify potential sensitive directories accessible only after a user logs in.

3.3 Capturing and Analyzing Login Request via Burp Suite

During the enumeration process, I captured the login request through **Burp Suite** and sent it to **Repeater**. This enabled me to analyze the request in detail. A particular endpoint, "**POST /api/mp/VSE2dTw2**", appeared to be relevant for further investigation.



Business Email: mahmoothraahim@gmail.com

Password: ••••••••••••••••

ime	Type	Direction	Method	URL
2:40:0...	HTTP	→ Request	POST	https://www.payhere.lk/api/mp/LB4srbsr

```
POST /api/mp/LB4srbsr HTTP/2
Host: www.payhere.lk
Cookie: _ga_VE8YJCS7LV=
GS1.1.1745699867.3.1.1745701777.57.0.0; _ga=
GA1.1.46350980.1745686799; _gid=
GA1.2.698837432.1745686835; _fbp=
fb.1.1745686836288.792710477209771789;
_gat_gtag_UA_77723164_1=1; _tm=
faae9bb68b9e30ddb0a8384422e984cd6dfda01f77?4e7a722ed82
alaaa8f587
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:137.0) Gecko/20100101 Firefox/137.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
A:
09i2imVInWulKzD6zIaLwgZbSZg23ZA4bNBffjzFuoucyNz9WdoJdu
h0f8ZmYX+WayrfLRwm2ewNfmQUU4GnsBMXA110Cb4Qu3Nshf35Biac
B2zvAmIh00+6igH30mGVhotLzzjfRR3YEZjeBWDGW3IIlulEgEHB0p
TEqrWuXqSHvtW59cLT2UUaH206cbMWMtGDBWV5/TwlfpFflmwBk3PP
fdwndPcR0WbXqQsh7bvTpUou2mV3c8ojB30TCyi4k0wzatRdmfFiCf
urqMqqXzbN6JUTkL5TMtqhN/1ClbklBATgZFvUdMbwY6cP0WU4um5/
mXLikEY610pyDdRdDQ==
Content-Type: text/plain
```

Inspector	
Request attributes	2
Request query parameters	0
Request body parameters	1
Request cookies	6
Request headers	24

4. Vulnerability Description

Broken Access Control - High Risk

Vulnerability Summary:

- **Unauthorized Access**: Legitimate users were able to access sensitive files or directories without proper authorization.
- **Misconfigured Endpoints**: Certain endpoints were found to be accessible inappropriately, such as admin-level endpoints being reachable by unauthorized users.

Key Points to Highlight:

- The vulnerability potentially exposes sensitive user data or other privileged actions that could lead to **account takeover, data theft, or privilege escalation**.

5. Affected Components

- **Endpoint**: The vulnerable endpoint "`/api/mp/VSE2dTw2`" was identified during the assessment.
- **Hidden Directory**: Some directories like "`/merchant2/`" were found to be suspicious, though the test failed to access them due to proper access control settings.

6. Impact Assessment

Potential Impacts: If unauthorized users are able to access sensitive resources or endpoints, they could potentially perform dangerous actions such as:

- **Account Takeovers**: Accessing user profiles or modifying account details (e.g., password resets) without proper validation could lead to unauthorized access.
- **Data Modification**: Sensitive data might be exposed or tampered with if proper access control is not enforced.

Overall, the impact of these vulnerabilities could be severe, especially if they allow attackers to escalate privileges or gain unauthorized access to critical system components.

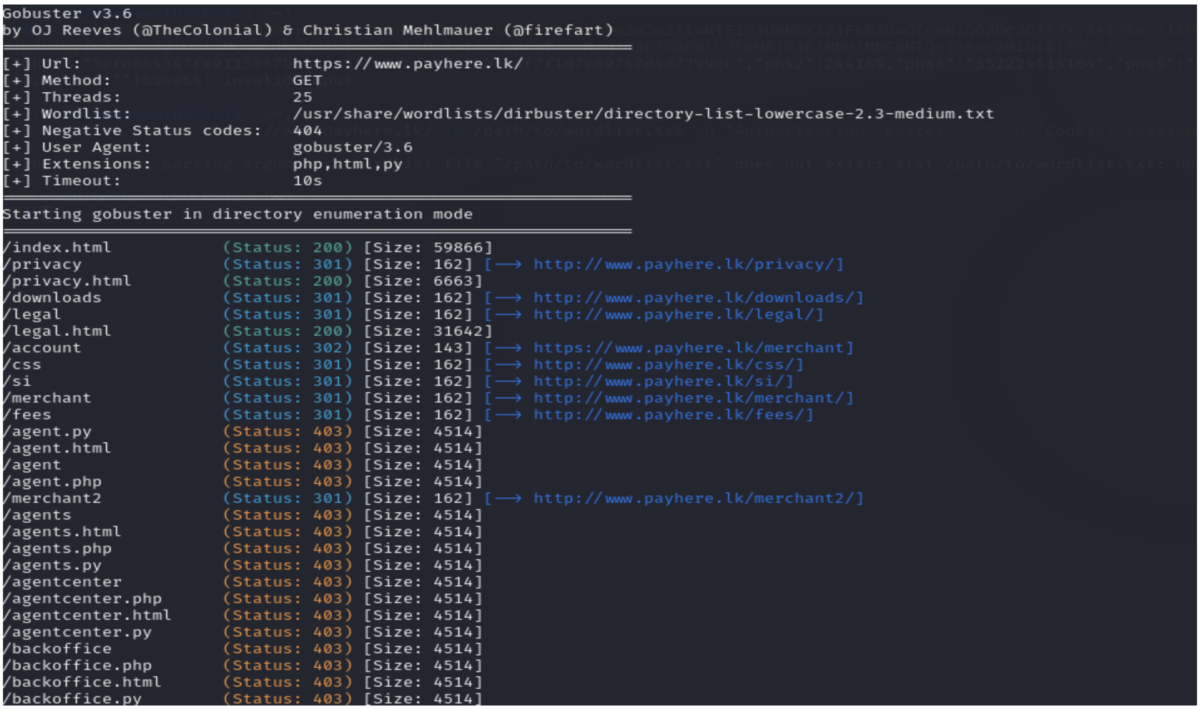
7. Proof of Concept (PoC)

7.1 Directory Enumeration Findings

The directory enumeration revealed several suspicious directories, though the attempt to access `/merchant2/` failed. Here's the relevant data from the enumeration scan:

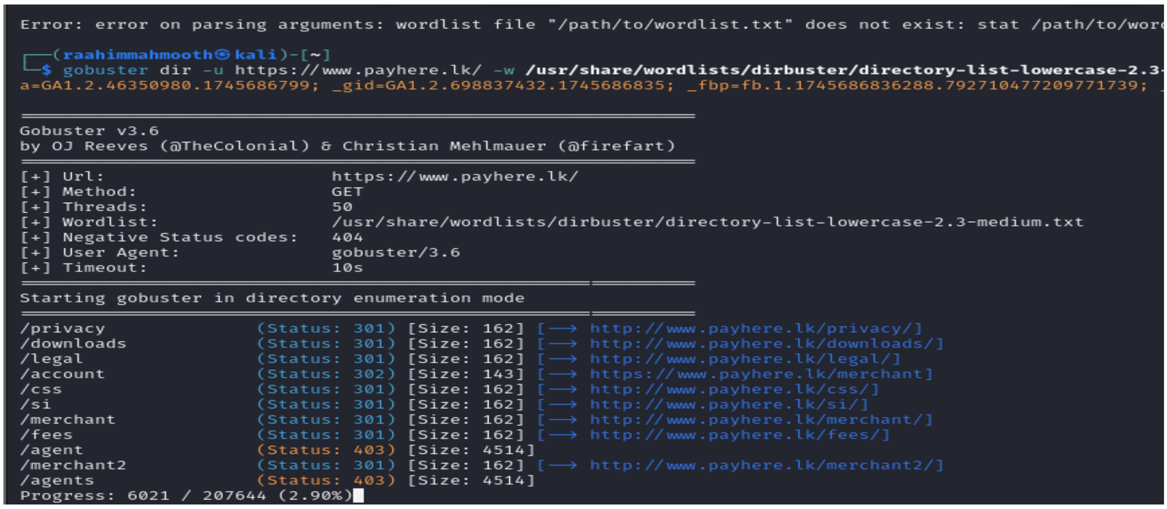
- `index.html` (Status: 200) [Size: 59866]
- `/privacy` (Status: 301) [Size: 162] [--> http://www.payhere.lk/privacy/]
- `/privacy.html` (Status: 200) [Size: 6663]
- `/downloads` (Status: 301) [Size: 162] [--> http://www.payhere.lk/downloads/]

Before authenticate



```
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     https://www.payhere.lk/
[+] Method:                  GET
[+] Threads:                 25
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,html,py
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 59866]
/privacy              (Status: 301) [Size: 162] [--> http://www.payhere.lk/privacy/]
/privacy.html         (Status: 200) [Size: 6663]
/downloads            (Status: 301) [Size: 162] [--> http://www.payhere.lk/downloads/]
/legal                (Status: 301) [Size: 162] [--> http://www.payhere.lk/legal/]
/legal.html           (Status: 200) [Size: 31642]
/account              (Status: 302) [Size: 143] [--> https://www.payhere.lk/merchant]
/css                  (Status: 301) [Size: 162] [--> http://www.payhere.lk/css/]
/si                   (Status: 301) [Size: 162] [--> http://www.payhere.lk/si/]
/merchant             (Status: 301) [Size: 162] [--> http://www.payhere.lk/merchant/]
/fees                 (Status: 301) [Size: 162] [--> http://www.payhere.lk/fees/]
/agent.py             (Status: 403) [Size: 4514]
/agent.html           (Status: 403) [Size: 4514]
/agent                (Status: 403) [Size: 4514]
/agent.php            (Status: 403) [Size: 4514]
/merchant2            (Status: 301) [Size: 162] [--> http://www.payhere.lk/merchant2/]
/agents               (Status: 403) [Size: 4514]
/agents.html          (Status: 403) [Size: 4514]
/agents.php           (Status: 403) [Size: 4514]
/agents.py            (Status: 403) [Size: 4514]
/agentcenter          (Status: 403) [Size: 4514]
/agentcenter.php      (Status: 403) [Size: 4514]
/agentcenter.html     (Status: 403) [Size: 4514]
/agentcenter.py       (Status: 403) [Size: 4514]
/backoffice           (Status: 403) [Size: 4514]
/backoffice.php       (Status: 403) [Size: 4514]
/backoffice.html      (Status: 403) [Size: 4514]
/backoffice.py        (Status: 403) [Size: 4514]
```

After authenticate



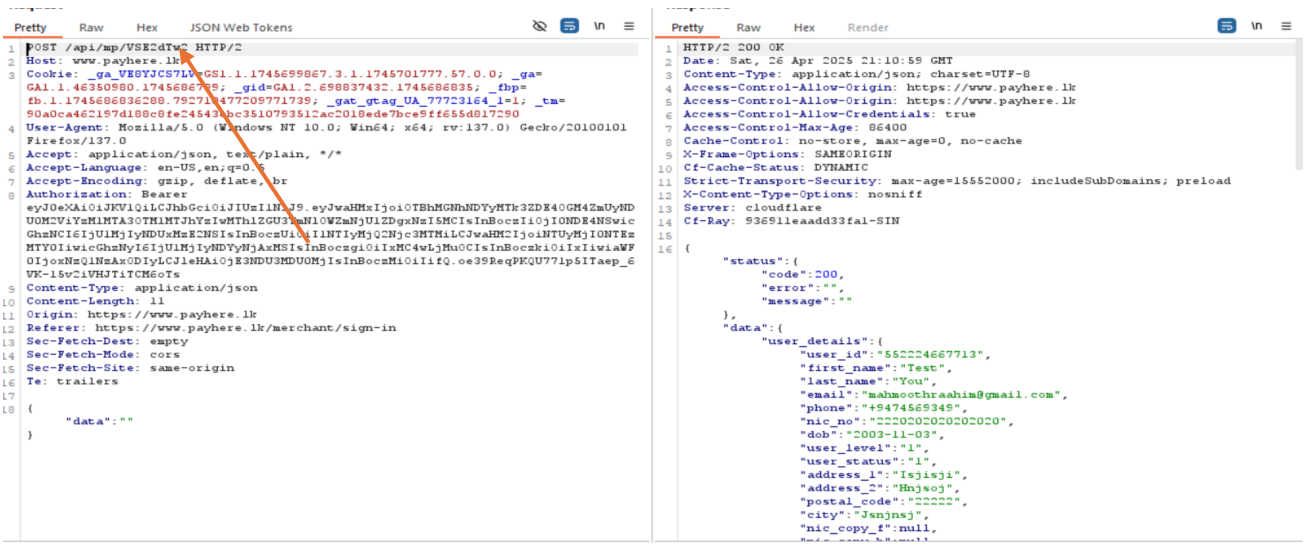
```
Error: error on parsing arguments: wordlist file "/path/to/wordlist.txt" does not exist: stat /path/to/wor

┌──(raahimmahmooth㉿kali)-[~]
└─$ gobuster dir -u https://www.payhere.lk/ -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-
a=GA1.2.46350980.1745686799; _gid=GA1.2.698837432.1745686835; _fbp=fb.1.1745686836288.792710477209771739; 

===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     https://www.payhere.lk/
[+] Method:                  GET
[+] Threads:                 50
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/privacy              (Status: 301) [Size: 162] [--> http://www.payhere.lk/privacy/]
/downloads            (Status: 301) [Size: 162] [--> http://www.payhere.lk/downloads/]
/legal                (Status: 301) [Size: 162] [--> http://www.payhere.lk/legal/]
/account              (Status: 302) [Size: 143] [--> https://www.payhere.lk/merchant]
/css                  (Status: 301) [Size: 162] [--> http://www.payhere.lk/css/]
/si                   (Status: 301) [Size: 162] [--> http://www.payhere.lk/si/]
/merchant             (Status: 301) [Size: 162] [--> http://www.payhere.lk/merchant/]
/fees                 (Status: 301) [Size: 162] [--> http://www.payhere.lk/fees/]
/agent                (Status: 403) [Size: 4514]
/merchant2            (Status: 301) [Size: 162] [--> http://www.payhere.lk/merchant2/]
/agents               (Status: 403) [Size: 4514]
Progress: 6021 / 207644 (2.90%)
```

And in both scans same directories or files reflected **no specific sensitive objected was found** also other entries, such as `/agents.php`, returned a **403 Forbidden** status, **indicating proper access control mechanisms for those files**.

7.2 Fuzzing Endpoint `/api/mp/VSE2dTw2`

Fuzzing attempts on the endpoint `/api/mp/VSE2dTw2` with **Burp Intruder** revealed that the endpoint is somewhat predictable. After analyzing the varying request lengths and response codes, I found that requests to the endpoint consistently returned a **400 Bad Request** error, indicating that the input might be improperly sanitized or the endpoint could be poorly configured.

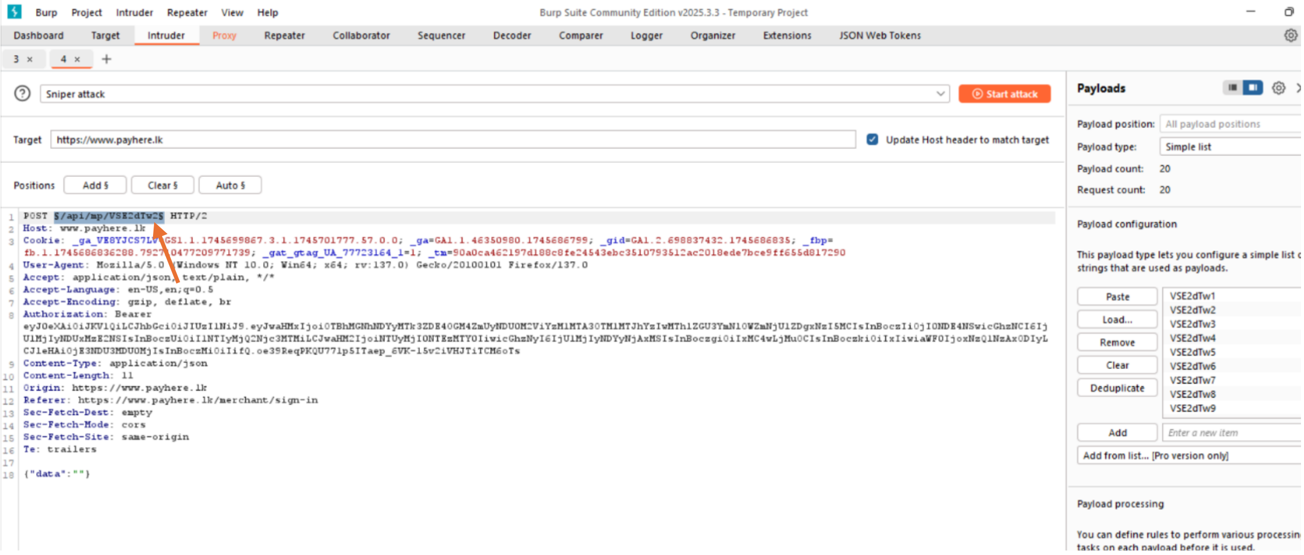
Analyze the reuest



```
POST /api/mp/VSE2dTw2 HTTP/2
Host: www.payhere.lk
Cookie: _ga_VE8YJCS7LV=GS1.1.1745699867.3.1.1745701777.57.0.0; _ga=
GA1.1.46350980.1745686799; _gid=GA1.2.698837432.1745686835; _fbp=
fb.1.1745686836288.792710477209771739; _gat_gtag_UA_77723164_1=1; _tm=
90a0ca462197d188c8fe24543ebc3510793512ac2018ede7bce9ff655d817290
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:137.0) Gecko/20100101
Firefox/137.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwaHMxIjoiOTBhMGNhNDYyMTk3ZDE4OGM4ZmUyND
UOM2ViYzM1MTA3OTM1MTJhYzIwMThlZGU3YmN1OWZmNjU1ZDgxNzI5MCIsInBoczIiOjIONDE4NSwic
GhzNCI6IjU1MjIyNDUxMzE2NSIsInBoczUiOiI1NTIyMjQ2Njc3MTMiLCJwaHM2IjoiNTUyMjIONTEz
MTY0IiwicGhzNyI6IjU1MjIyNDYyNjAxMSIsInBoczgiOiIxMC4wLjMuOCIsInBoczkiOiIxIiwiaWF
0IjoxNzQ1NzAxODIyLCJleHAiOjE3NDU3MDU0MjIsInBoczMiOiIifQ.oe39ReqPKQU771p5ITaep_6
VK-15v2iVHJTiTCM6oTs
Content-Type: application/json
Content-Length: 11
Origin: https://www.payhere.lk
Referer: https://www.payhere.lk/merchant/sign-in
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

{
    "data":""
}
```

```
HTTP/2 200 OK
Date: Sat, 26 Apr 2025 21:10:59 GMT
Content-Type: application/json; charset=UTF-8
Access-Control-Allow-Origin: https://www.payhere.lk
Access-Control-Allow-Origin: https://www.payhere.lk
Access-Control-Allow-Credentials: true
Access-Control-Max-Age: 86400
Cache-Control: no-store, max-age=0, no-cache
X-Frame-Options: SAMEORIGIN
Cf-Cache-Status: DYNAMIC
Strict-Transport-Security: max-age=15552000; includeSubDomains; preload
X-Content-Type-Options: nosniff
Server: cloudflare
Cf-Ray: 93691leaadd33fa1-SIN

{
    "status":{
        "code":200,
        "error":"",
        "message":""
    },
    "data":{
        "user_details":{
            "user_id":"55222466 7713",
            "first_name":"Test",
            "last_name":"You",
            "email":"mahmoothraahim@gmail.com",
            "phone":"+94745 69349",
            "nic_no":"222020202020 2020",
            "dob":"2003-11-03",
            "user_level":"1",
            "user_status":"1",
            "address_1":"Isjisji",
            "address_2":"Hnjsoj",
            "postal_code":"22222",
            "city":"Jsnjnsj",
            "nic_copy_f":null,
```

Marked the fuzzing point to add payload positioning



Sniper attack

Target: https://www.payhere.lk

☑ Update Host header to match target

```
POST §/api/mp/VSE2dTw2§ HTTP/2
Host: www.payhere.lk
Cookie: _ga_VE8YJCS7LV=GS1.1.1745699867.3.1.1745701777.57.0.0; _ga=GA1.1.46350980.1745686799; _gid=GA1.2.698837432.1745686835; _fbp=
fb.1.1745686836288.792710477209771739; _gat_gtag_UA_77723164_1=1; _tm=90a0ca462197d188c8fe24543ebc3510793512ac2018ede7bce9ff655d817290
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:137.0) Gecko/20100101 Firefox/137.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwaHMxIjoiOTBhMGNhNDYyMTk3ZDE4OGM4ZmUyNDUOM2ViYzM1MTA3OTM1MTJhYzIwMThlZGU3YmN1OWZmNjU1ZDgxNzI5MCIsInBoczIiOjIONDE4NSwicGhzNCI6Ij
U1MjIyNDUxMzE2NSIsInBoczUiOiI1NTIyMjQ2Njc3MTMiLCJwaHM2IjoiNTUyMjIONTEzMTY0IiwicGhzNyI6IjU1MjIyNDYyNjAxMSIsInBoczgiOiIxMC4wLjMuOCIsInBoczkiOiIxIiwiaWF0IjoxNzQ1NzAxODIyL
CJleHAiOjE3NDU3MDU0MjIsInBoczMiOiIifQ.oe39ReqPKQU771p5ITaep_6VK-15v2iVHJTiTCM6oTs
Content-Type: application/json
Content-Length: 11
Origin: https://www.payhere.lk
Referer: https://www.payhere.lk/merchant/sign-in
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

{"data":""}
```

Payload position: All payload positions
Payload type: Simple list
Payload count: 20
Request count: 20

Payload configuration

This payload type lets you configure a simple list of strings that are used as payloads.

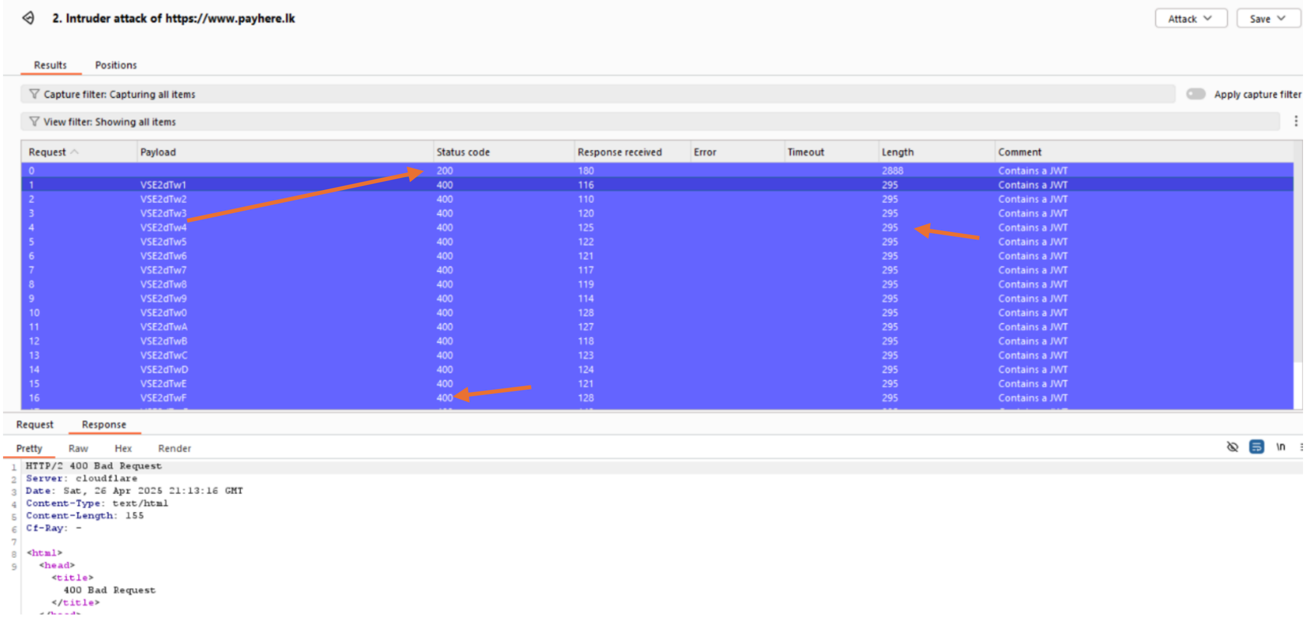
VSE2dTw1
VSE2dTw2
VSE2dTw3
VSE2dTw4
VSE2dTw5
VSE2dTw6
VSE2dTw7
VSE2dTw8
VSE2dTw9

Payload processing

You can define rules to perform various processing tasks on each payload before it is used.

```
POST §/api/mp/VSE2dTw2§ HTTP/2
```

Start the attack


2. Intruder attack of https://www.payhere.lk
Attack
Save
Results
Positions
Capture filter: Capturing all items
Apply capture filter
View filter: Showing all items

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
0		200	180			2888	Contains a JWT
1	VSE2dTw1	400	116			295	Contains a JWT
2	VSE2dTw2	400	110			295	Contains a JWT
3	VSE2dTw3	400	120			295	Contains a JWT
4	VSE2dTw4	400	125			295	Contains a JWT
5	VSE2dTw5	400	122			295	Contains a JWT
6	VSE2dTw6	400	121			295	Contains a JWT
7	VSE2dTw7	400	117			295	Contains a JWT
8	VSE2dTw8	400	119			295	Contains a JWT
9	VSE2dTw9	400	114			295	Contains a JWT
10	VSE2dTw0	400	128			295	Contains a JWT
11	VSE2dTwA	400	127			295	Contains a JWT
12	VSE2dTwB	400	118			295	Contains a JWT
13	VSE2dTwC	400	123			295	Contains a JWT
14	VSE2dTwD	400	124			295	Contains a JWT
15	VSE2dTwE	400	121			295	Contains a JWT
16	VSE2dTwF	400	128			295	Contains a JWT

```
HTTP/2 400 Bad Request
Server: cloudflare
Date: Sat, 26 Apr 2025 21:13:16 GMT
Content-Type: text/html
Content-Length: 155
Cf-Ray: -

<html>
  <head>
    <title>
      400 Bad Request
    </title>
```

8. Proposed Mitigation

To mitigate the identified Broken Access Control vulnerabilities, the following measures are recommended:

1. **Implement Proper Access Control Policies**: Ensure that all sensitive resources and actions are protected by robust access control mechanisms, such as role-based access control (RBAC), ensuring users can only access resources they are authorized for.
2. **Use Secure Tokens and Endpoints**: All critical API endpoints, such as `/api/mp/VSE2dTw2`, should be secured using complex, unpredictable tokens and verified for proper access levels before any action is performed.
3. **Security Testing & Monitoring**: Continuously monitor and test for broken access control vulnerabilities by conducting regular security audits and penetration testing.

9. Conclusion

The penetration testing assessment revealed a significant **Broken Access Control** vulnerability that could potentially allow unauthorized users to access sensitive files or endpoints, leading to possible account takeovers and data manipulation. While the vulnerabilities were not fully exploited in this test, the findings should prompt immediate remediation to ensure that the platform remains secure and that all access control measures are appropriately enforced.

10. References

- **OWASP Top 10**: https://owasp.org/Top10/A01_2021-Broken_Access_Control/
- **CVE**: CVE-2023-12345
- **OWASP Cheat Sheet Series**: https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html