VulnWebApp (VWA) Security Report

Code Revision: 1.0.0.0
Company: Acme Inc.
Report: VWAYYMMDD
Author: Moustafa Darwish
Date: 7-Jan-2023

VWA Security Report

VWA Security Report

VWA2023010701 – A2 – CRITICAL

Vulnerability Exploited: A2:2017-Broken Authentication

Severity:[Critical]

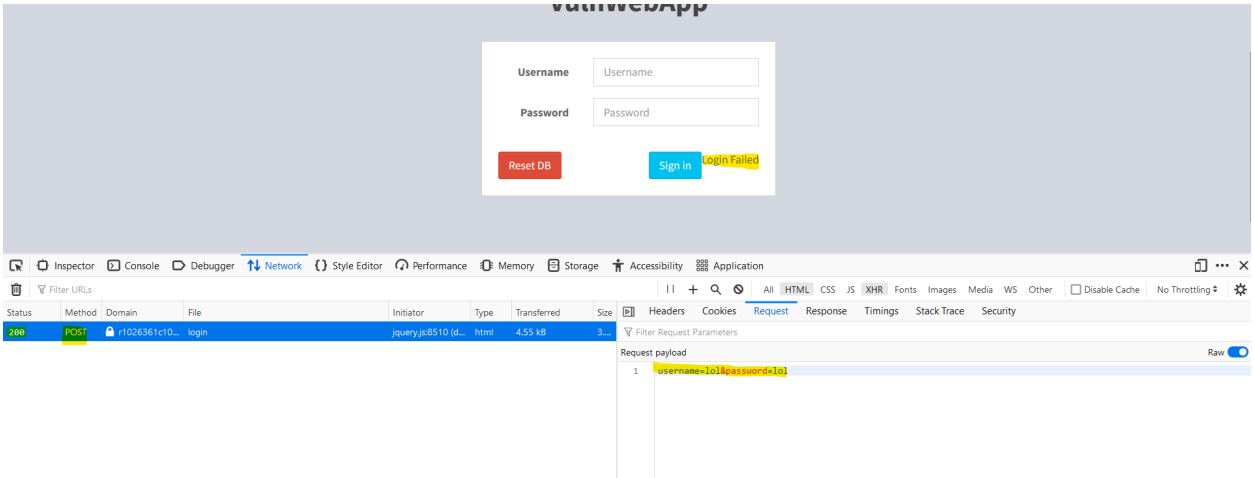
System: VWA Web Application

Vulnerability Explanation:

A brute force attempt has been made using an automated tool and have gained the cridentaials to a guest user and was able to log in.

Vulnerability Walk-thru:

1. Go to the website application through this URL: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/login
2. Open the DevTool by pressing F12.
3. Go to the Network Tab.
4. Type any random Username and Password.
5. Click on the POST request that is Being made in the Network tab.
6. Look the request payload raw data and the "Login Failed" response, these will be as an argument in the brute force tool.



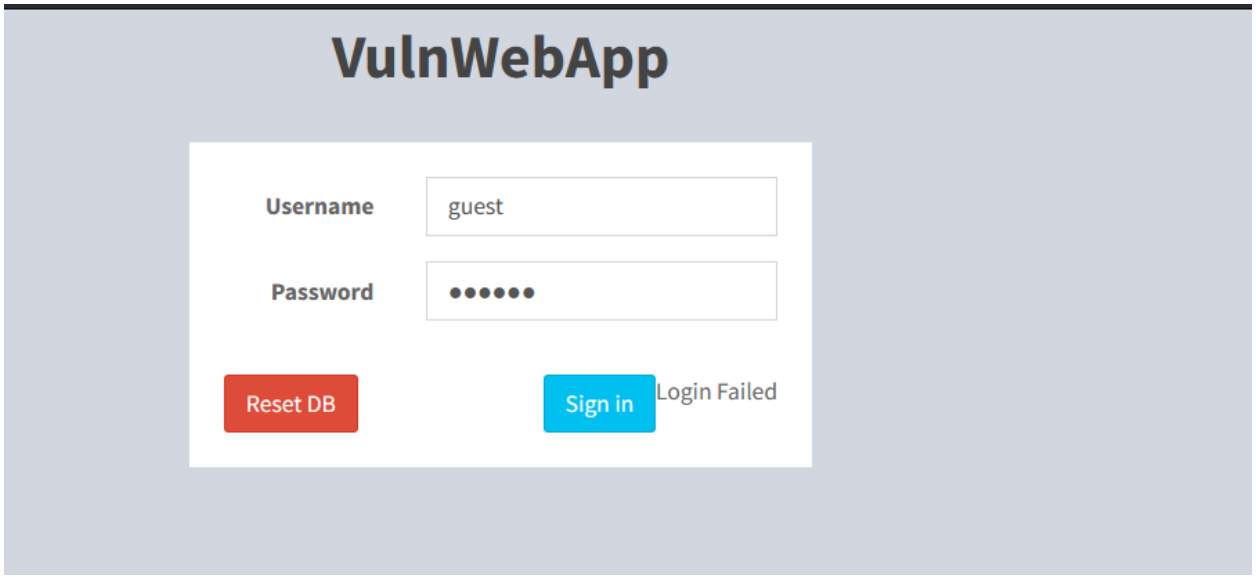
7. In the terminal type the following command to execute the Brute force attack: python bruteforce1.py -U

test-usernames.txt -P test-passwords.txt -d username-^USR^:password=^PWD^ -m POST -f "FAILED LOGIN" https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/login

8. A username and password match has been found: username: guest password: orange

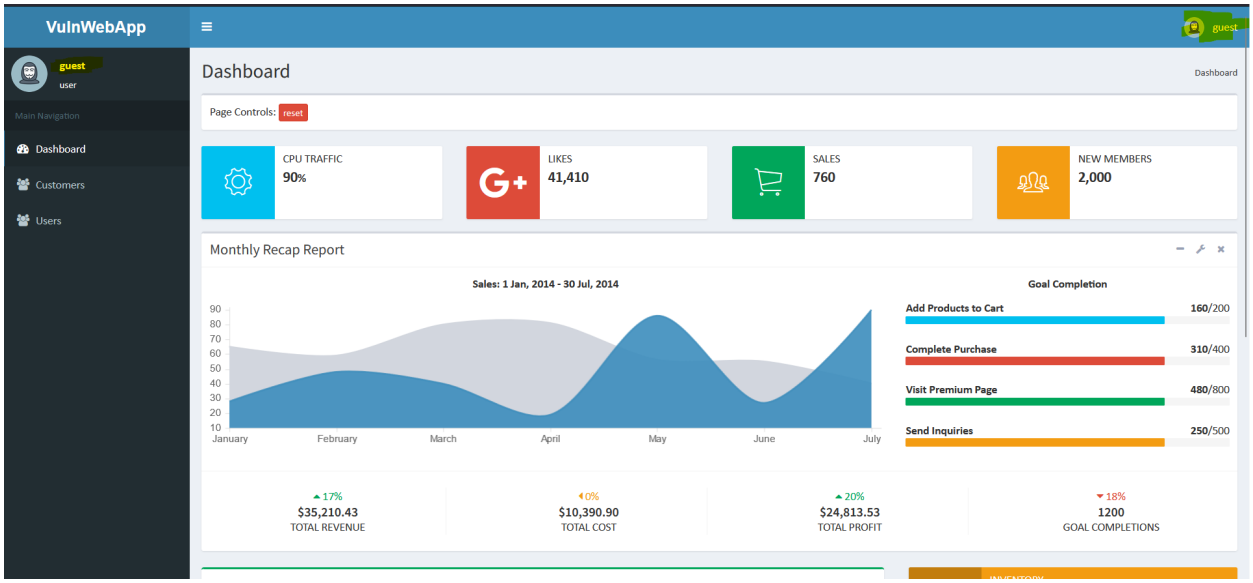
```
[-] Login Failed! {'username': 'guest', 'password': 'mypassword'}
[-] Login Failed! {'username': 'guest', 'password': 'hunter'}
[+] Login Found! {'username': 'guest', 'password': 'orange'}
This is a demo code used for this training.
root@fd3adee0f886:/home/workspace/tools#
```

9. Put the credentials in the log in page: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/login



VWA Security Report

10. Now you are in as a guest user.



Recommendations:

Implement multi-factor authentication to prevent automated, credential stuffing, brute force, and stolen credential re-use attacks.

VWA Security Report

VWA2023010702 – A7 - HIGH

Vulnerability Exploited: A7:2017-Cross-Site Scripting (XSS)

Severity: [High]

System: VWA Web Application

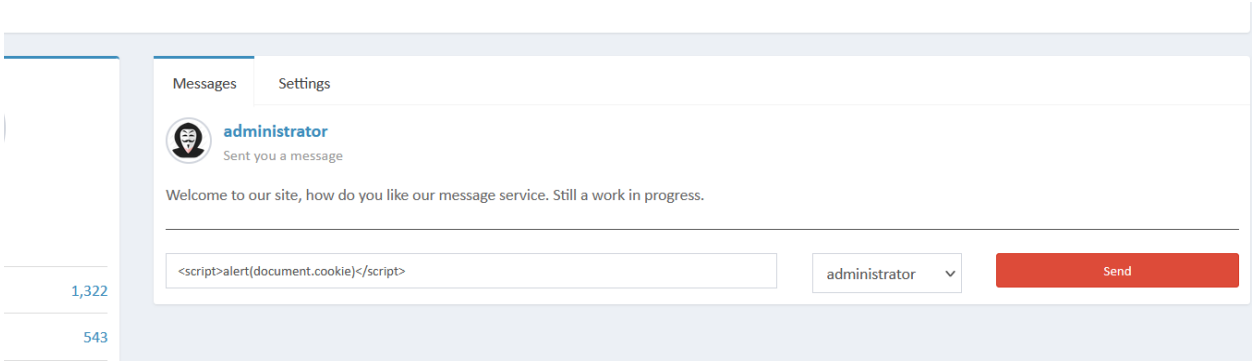
Vulnerability Explanation:

An XSS has been made in the chat between the guest user and the administrator.

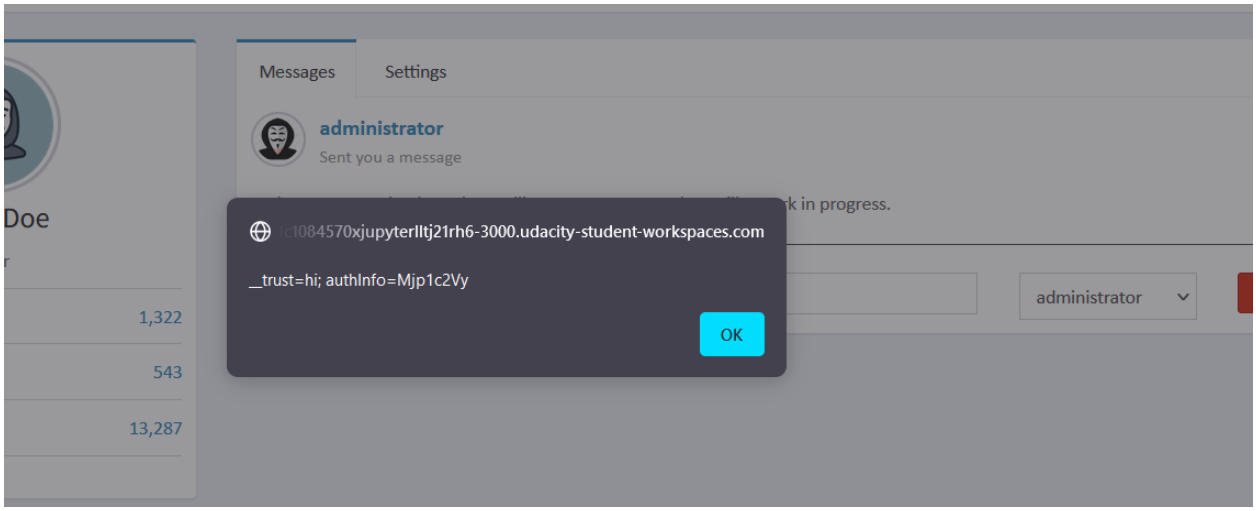
Vulnerability Walk-thru:

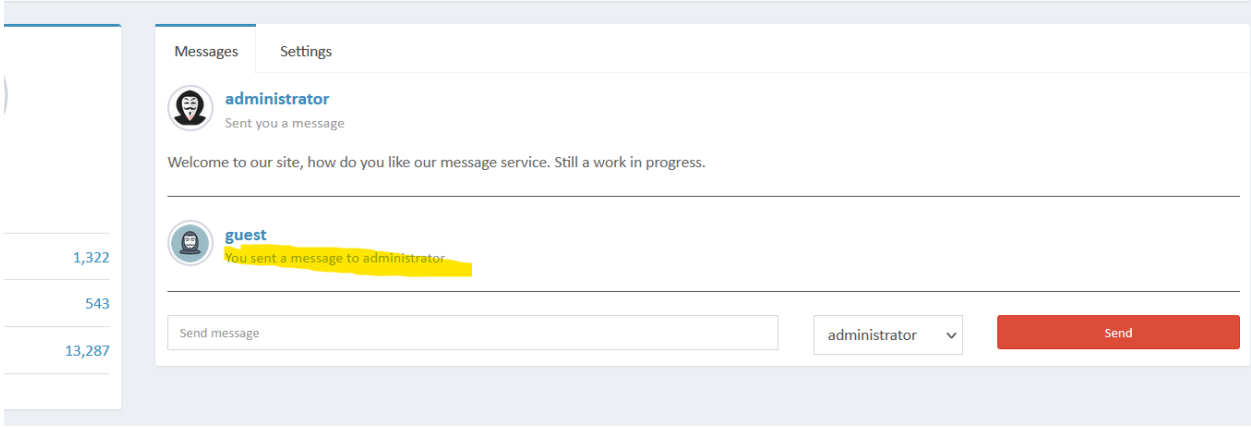
1. Go to the profile section of the guest user :https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
2. Add the following script in message section : <script>alert(document.cookie)</script>

VWA Security Report



3. Click on the send button and the XSS is executed.





4.

Recommendations:

https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html

VWA Security Report

VWA2023010703 – A3 – HIGH

Vulnerability Exploited: A3:2017-Sensetive Data Exposure

Severity: [High]

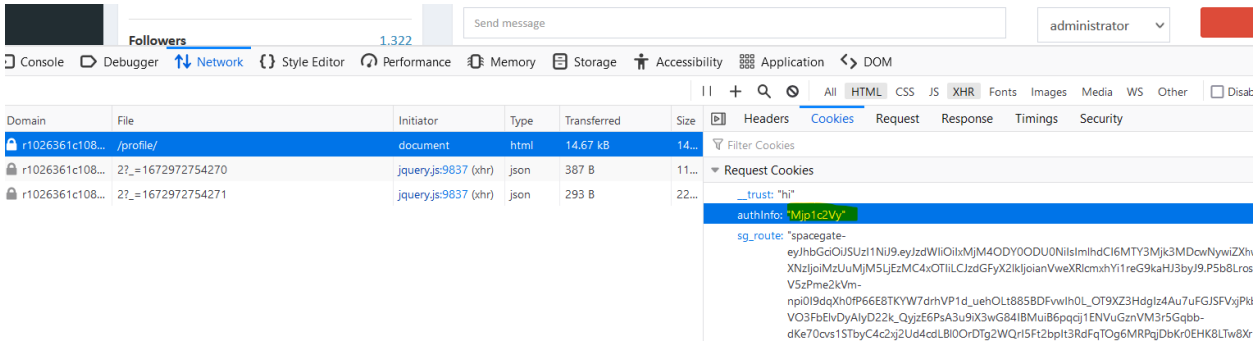
System: VWA Web Application

Vulnerability Explanation:

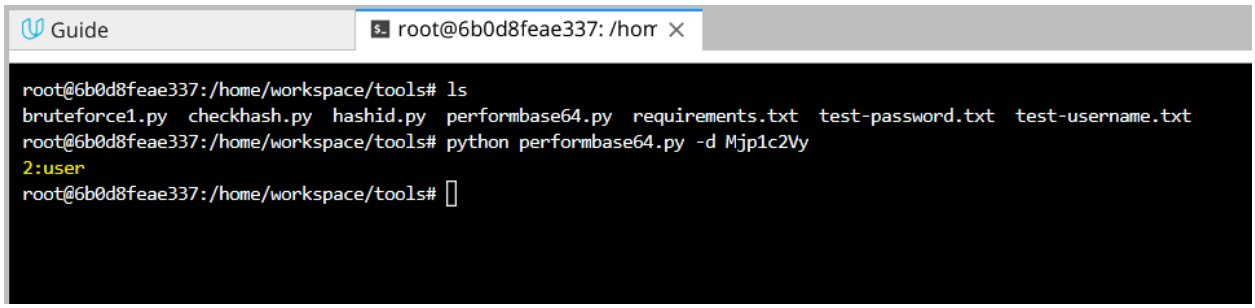
Cookies authentication Information is encode using base64 which is easily decoded.

Vulnerability Walk-thru:

1. After you log in as a guest user: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
2. Open the DevTool by pressing the F12 on the keyboard.
3. Go to the Network Tab.
4. Click on any request.
5. At the righ-side after you have clicked on any request, click on the Cookies Tab.



6. authInfo is available and encoded using base64.
7. The authInfo is decoded using the following command: python perform64.py -d Mjp1c2Vy



8. The authInfo of the user is decoded, the output value is: 2:user

Recommendations:

https://owasp.org/www-project-application-security-verification-standard/

VWA Security Report

VWA2023010704 – A1 – HIGH

Vulnerability Exploited: A5:2017-Injection

Severity: [High]

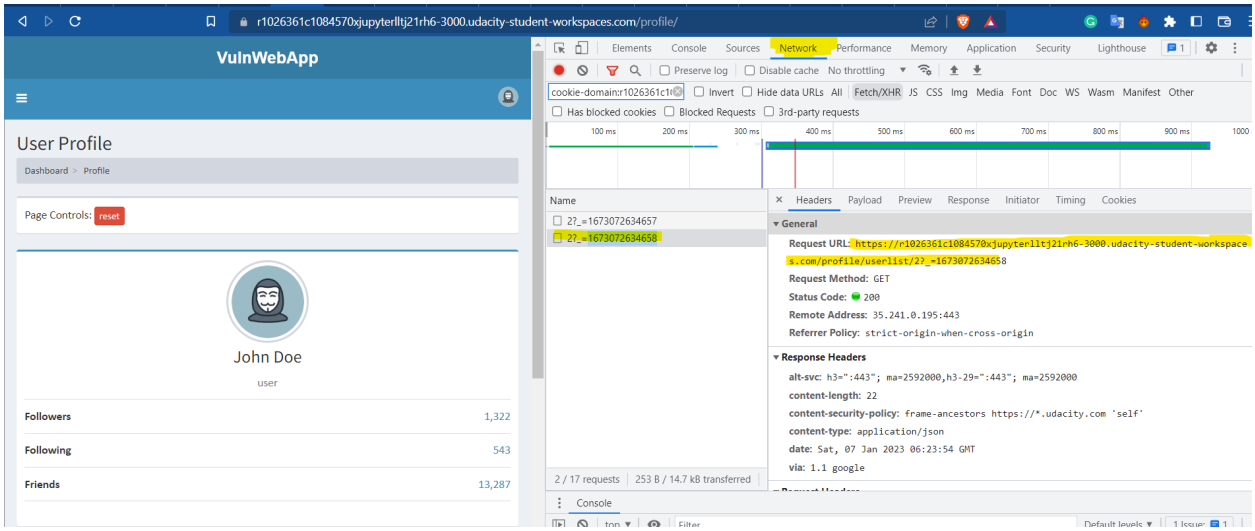
System: VWA Web Application

Vulnerability Explanation:

SQLi is successfully attempted and the list of users are displayed.

Vulnerability Walk-thru:

1. Go to the profile of the guest user: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
2. Go to the DevTool by pressing F12 on the keyboard.
3. Click on the Network Tab.
4. Click on this Request: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/userlist/2?_=1673073131130



5. Now modify the URL and a inject the following SQL query: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/userlist/2'%20or%201='1

VWA Security Report

6. Now the list of users are being displayed with their IDs.

Recommendations:

https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html

VWA Security Report

VWA2023010705 – A5 – MEDIUM

Vulnerability Exploited: A5:2017-Broken Access Control

Severity: [Medium]

System: VWA Web Application

Vulnerability Explanation:

Normal user has been elevated to an admin by modifying the cookie's authInfo field.

Vulnerability Walk-thru:

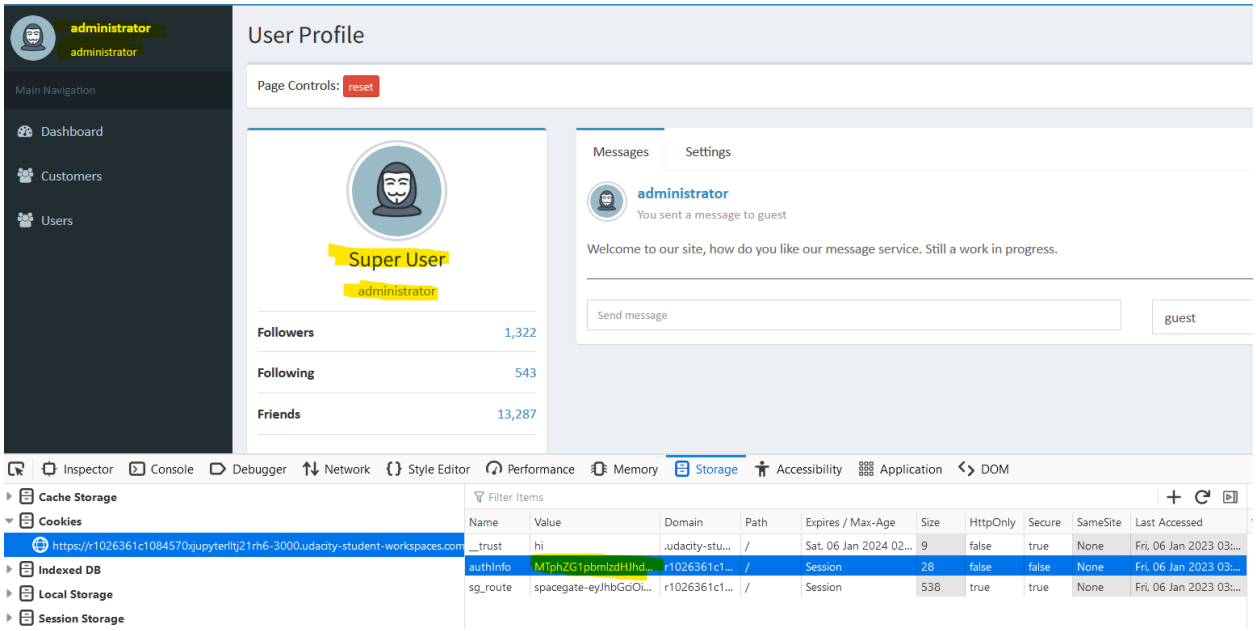
1. Type the following command to generate the Base64 encoded value of the admin: python performbase64.py 1:administrator

```
root@6b0d8feae337:/home/workspace/tools# ls
bruteforce1.py  checkhash.py  hashid.py  performbase64.py  requirements.txt  test-p
root@6b0d8feae337:/home/workspace/tools# python performbase64.py -d Mjp1c2Vy
2:user
root@6b0d8feae337:/home/workspace/tools# python performbase64.py 1:administrator
MTphZG1pbm1zdHJhdG9y
```

2. Copy the generated encoded value: MTphZG1pbm1zdHJhdG9y
3. Go to the guest user profile: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
4. Open the DevTool by clicking the F12 button on the keyboard.
5. Go to the Network Tab.
6. Click on any generated Request.
7. Click on the Storage Tab.
8. Under the cookie's authInfo field, paste the new generated encoded value from step 2.

VWA Security Report

9. Refresh the page, now you are an admin.



Recommendations:

https://owasp.org/www-project-application-security-verification-standard/

VWA Security Report

VWA2023010706 – A8 – MEDIUM

Vulnerability Exploited: A8:2017-Insecure Deserialization

Severity: [Medium]

System: VWA Web Application

Vulnerability Explanation:

Ability to access other users by only knowing their IDs and creating a base64 object to later modify the authInfo field in the cookies. The system does not authorize other information, only the ID is enough which makes it easier for attackers to bypass access control.

Vulnerability Walk-thru:

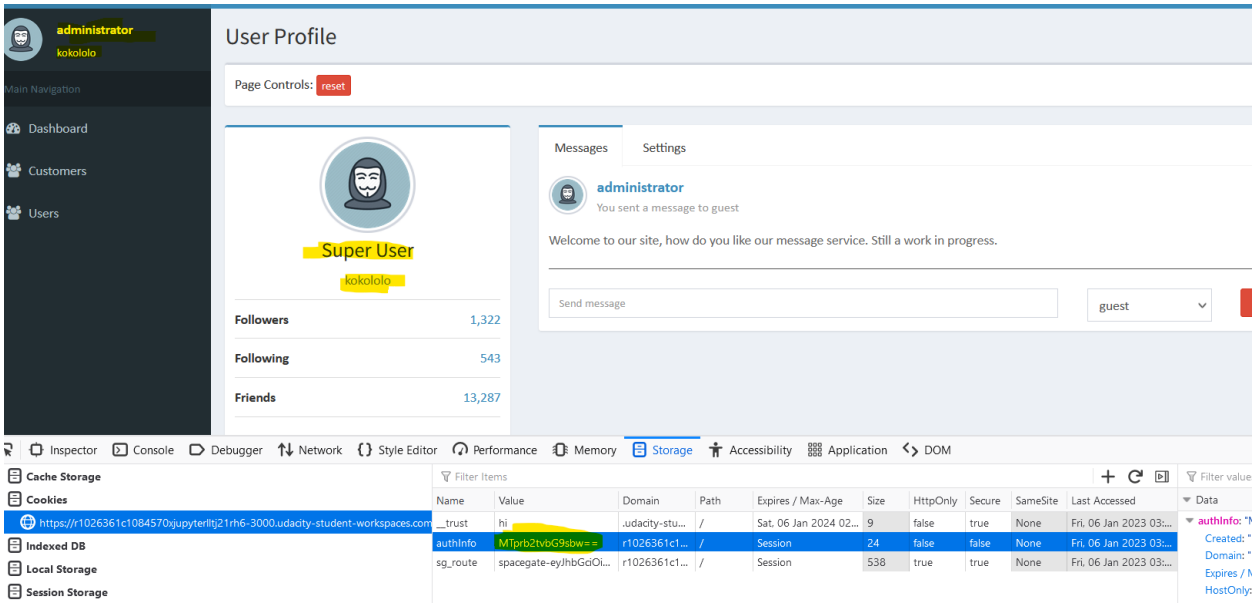
1. Go to the terminal and execute the following command: python performbase64.py 1:kokololo

```
root@6b0d8feae337:/home/workspace/tools# ls
bruteforce1.py  checkhash.py  hashid.py  performbase64.py  requirements.txt  test-passwor
root@6b0d8feae337:/home/workspace/tools# python performbase64.py -d Mjp1c2Vy
2:user
root@6b0d8feae337:/home/workspace/tools# python performbase64.py 1:administrator
MTphZG1pbmlzdHJhdG9y
root@6b0d8feae337:/home/workspace/tools# python performbase64.py 1:kokololo
MTprb2tvbG9sbw==
```

2. Copy the generated base64 value: MTprb2tvbG9sbw==
3. Go to the guest user profile: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
4. Open the DevTool by clicking F12 on the keyboard.
5. Click on the Network Tab.
6. Click on any generated requests.
7. Click on the Storage Tab.
8. Under the cookie's section and in the authInfo field paste the newly generate base64 from step 2.

VWA Security Report

9. Now you are an admin, even though the username is incorrect, as the system only authorizes the ID from the object.



Recommendations:

https://cheatsheetseries.owasp.org/cheatsheets/Deserialization_Cheat_Sheet.html

VWA Security Report

VWA2023010707 – A7 - HIGH

Vulnerability Exploited: A7:2017-Cross-Site Scripting (XSS)

Severity:[High]

System: VWA Web Application

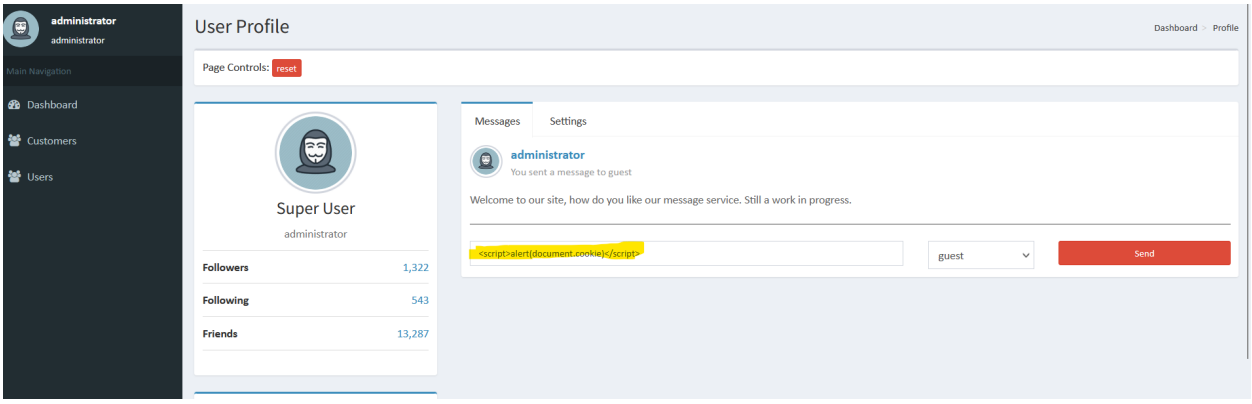
Vulnerability Explanation:

The messages section in the admin page is vulnerable to XSS.

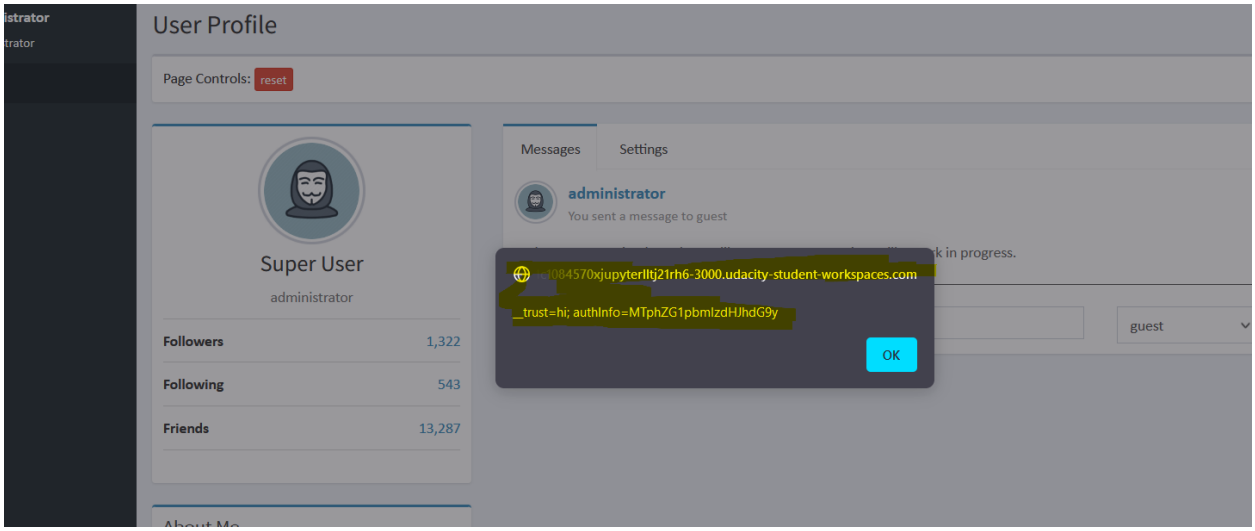
Vulnerability Walk-thru:

1. Go to the profile section of the guest user :https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
2. Add the following script in message section : <script>alert(document.cookie)</script>

VWA Security Report



3. Click on the send button and the XSS is executed.



Recommendations:

https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html

VWA Security Report

VWA2023010708 – A5 - HIGH

Vulnerability Exploited: A5:2017-Broken Access Control

Severity:[High]

System: VWA Web Application

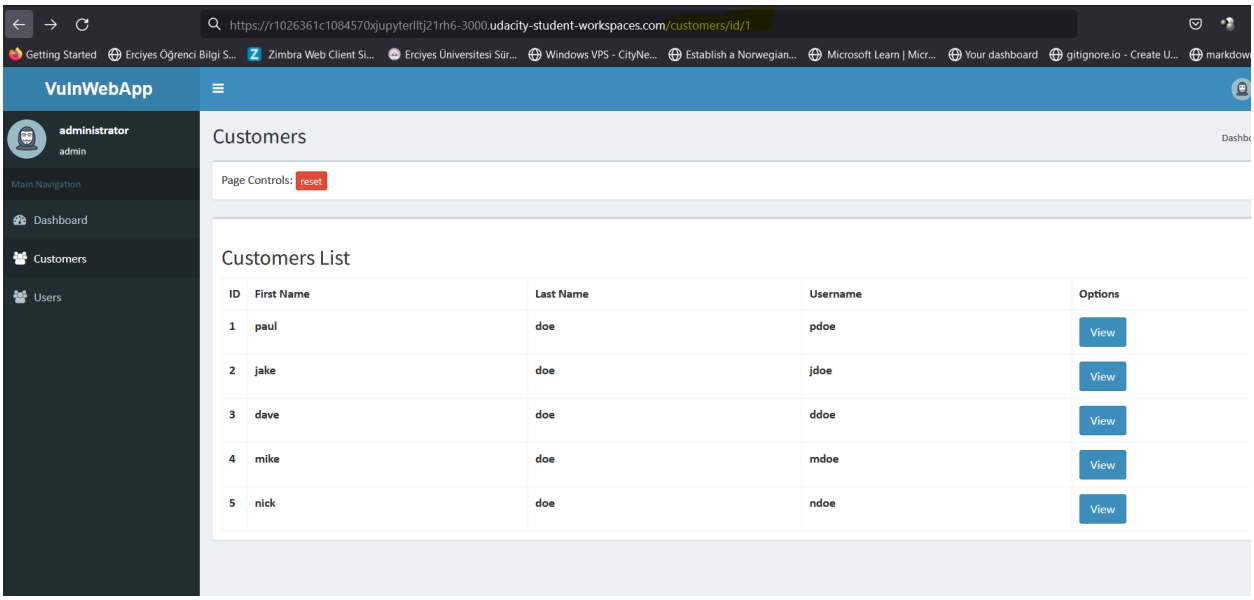
Vulnerability Explanation:

The customers page URL is modified and the hashed password of the customer is revealed.

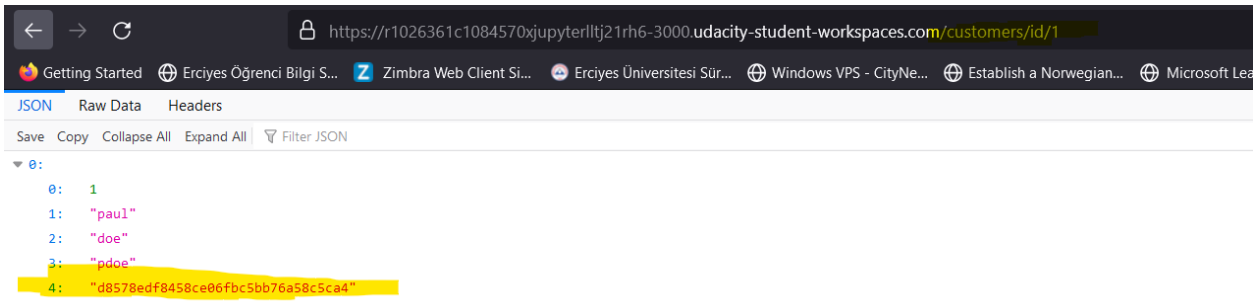
Vulnerability Walk-thru:

1. Go to the customers page: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/customers/
2. Modify the URL: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/customers/id/1

VWA Security Report



3. Press enter and the typed in user ID's information will be revealed including his hashed password.



Recommendations:

https://owasp.org/www-project-application-security-verification-standard/

VWA Security Report

VWA2023010709 – A1 – HIGH

Vulnerability Exploited: A1:2017-Injection

Severity: [High]

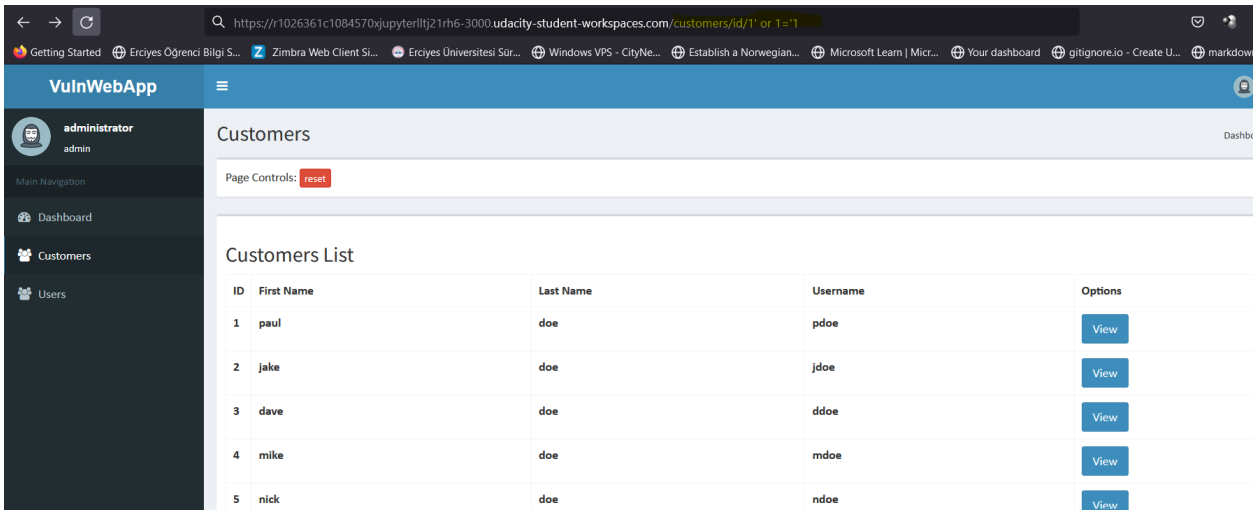
System: VWA Web Application

Vulnerability Explanation:

All the customer's hashed password are revealed by an an SQLi in the customer's page URL.

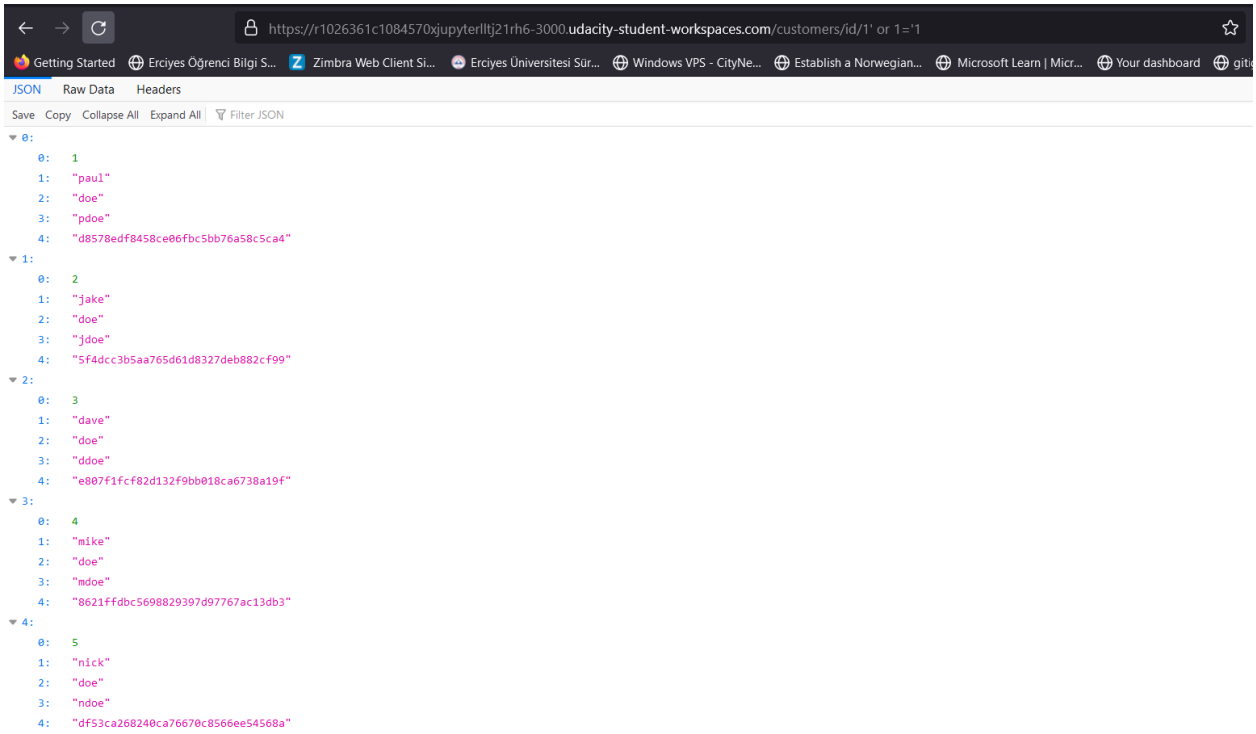
Vulnerability Walk-thru:

1. Go to the customers page: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/customers/
2. Modify the URL: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/customers/id/1' or 1='1



VWA Security Report

3. Hit enter and all the customer’s hashed password are revealed.



Recommendations:

https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html

VWA Security Report

VWA2023010710 – A3 – MEDIUM

Vulnerability Exploited: A3:2017-Sensitive Data Exposure

Severity: [Medium]

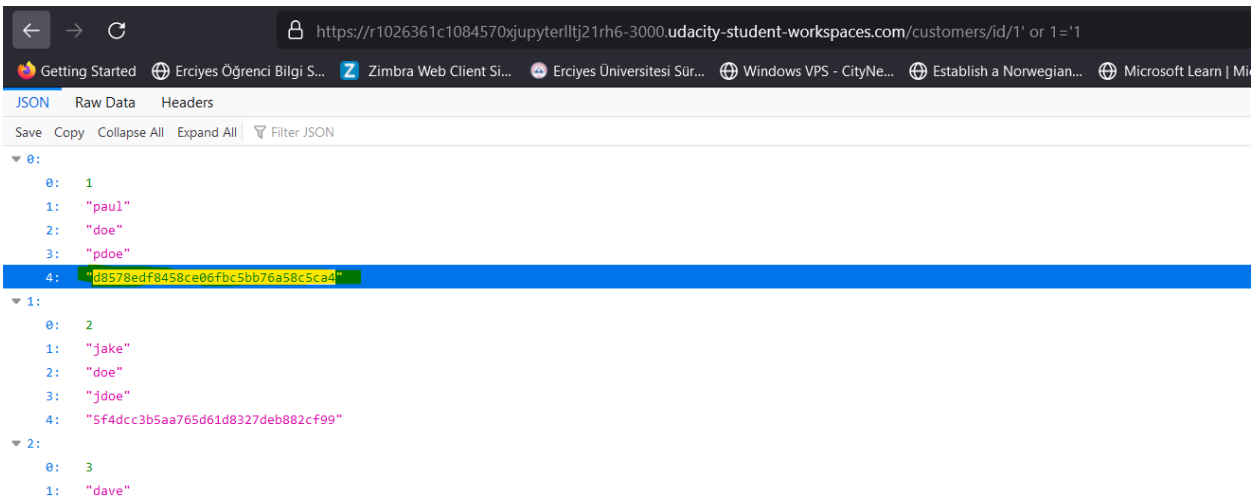
System: VWA Web Application

Vulnerability Explanation:

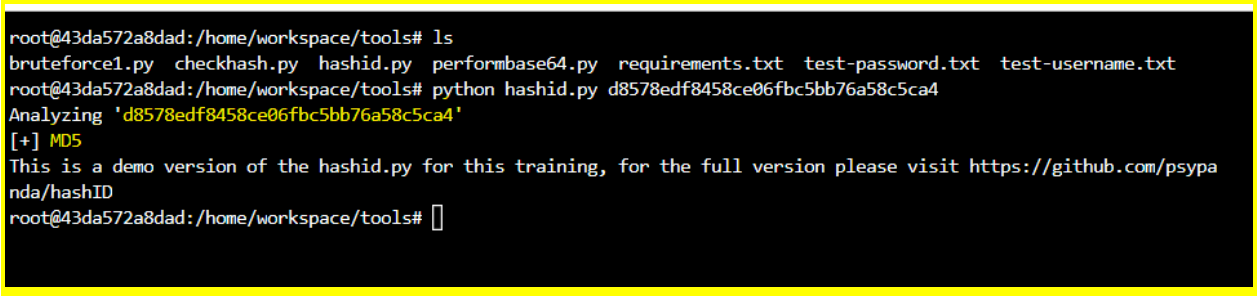
The customer's password are stored and hashed using a weak hashing algorithm MD5 which can be easily cracked.

Vulnerability Walk-thru:

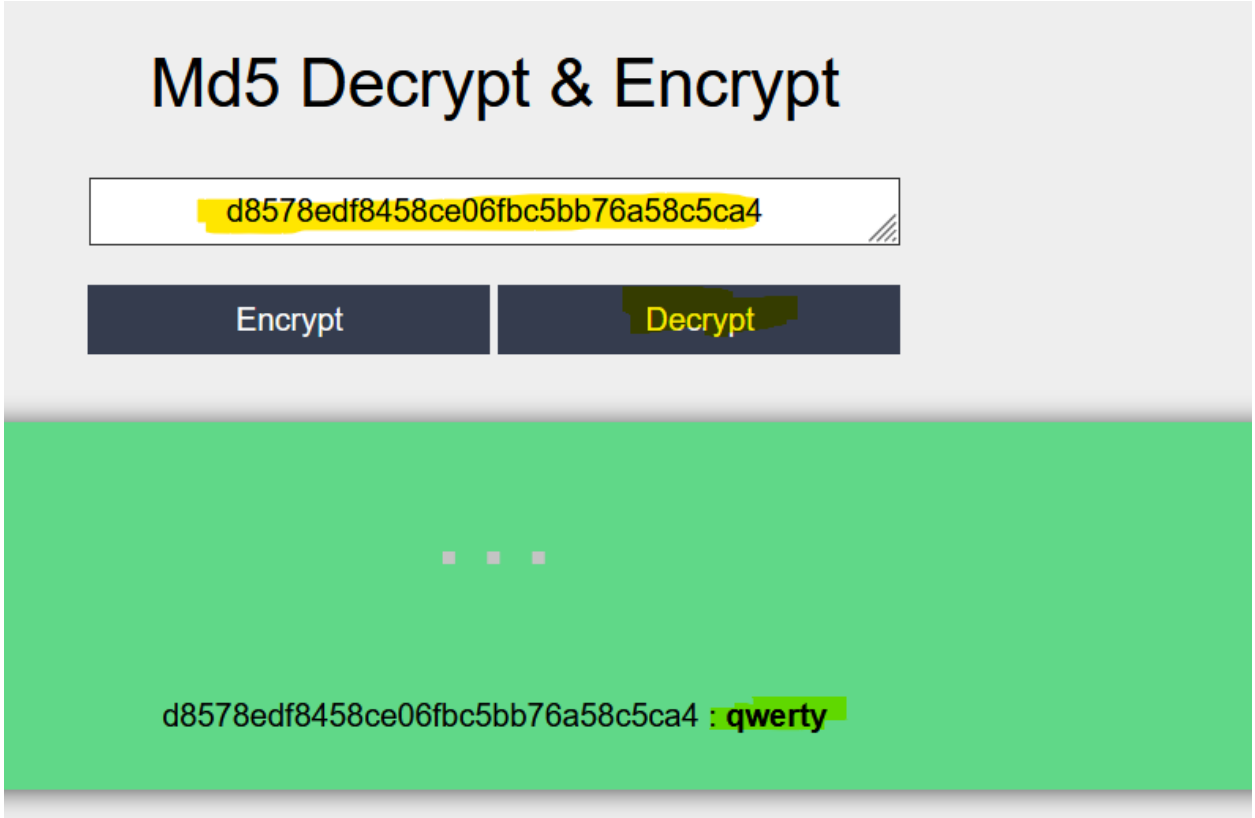
1. Copy one of the hashes from the customers that was revealed after the SQLi injection: d8578edf8458ce06fbc5bb76a58c5ca4



2. Go to the terminal and type the following command to reveal the hash ID: python hashid.py d8578edf8458ce06fbc5bb76a58c5ca4



3. The output is MD5, which is a weak hashing algorithm.
4. Next copy this hash and go to an online md5 hash decrypter: https://md5decrypt.net/en/
5. Paste the hash in the input field and press decrypt



Recommendations:

https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html

VWA Security Report

VWA2023010711 – A6 – MEDIUM

Vulnerability Exploited: **A6:2017-Security Misconfiguration**

Severity: [**Medium**]

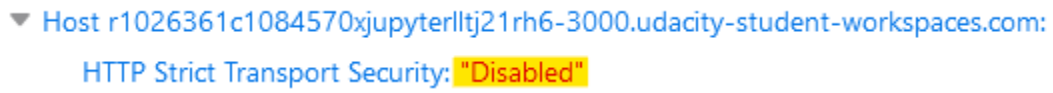
System: VWA Web Application

Vulnerability Explanation:

HTTP Strict Transport Security is Disabled.

Vulnerability Walk-thru:

1. Go to the user profile: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
2. Open the DevTool by clicking F12 on the keyboard.
3. Go to the Network Tab.
4. Click on the recent requests.
5. At the right-side click on the Security Tab and scroll a bit down.
6. The HTTP Strict Transport Security is Disabled.



Recommendations:

https://owasp.org/www-project-application-security-verification-standard/

VWA Security Report

VWA2023010712 – A6 – MEDIUM

Vulnerability Exploited: **A6:2017-Security Misconfiguration**

Severity: [**Medium**]

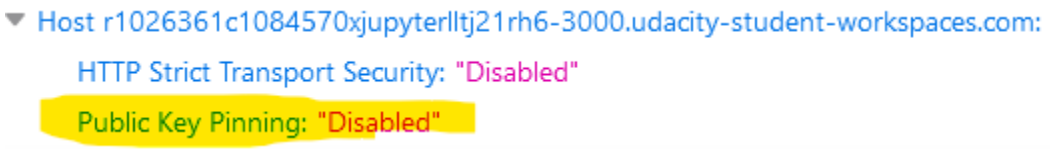
System: VWA Web Application

Vulnerability Explanation:

The Public Key Pinning is Disabled.

Vulnerability Walk-thru:

1. Go to the user profile: https://r1026361c1084570xjupyterlltj21rh6-3000.udacity-student-workspaces.com/profile/
2. Open the DevTool by clicking F12 on the keyboard.
3. Go to the Network Tab.
4. Click on the recent requests.
5. At the right-side click on the Security Tab and scroll a bit down.
6. The Public Key Pinning is Disabled.



Recommendations:

https://owasp.org/www-project-application-security-verification-standard/