

VulnWebApp (VWA)

Security Report

Code Revision: 1.0.0.0

Company: Acme Inc.

Report: VWA210907

Author: Matteo S.

Date: 2021, Sep. 7th

VWA Security Report

VWA210907a - login-system - Critical	3
VWA210907b - session-role-cookie - Critical	4
VWA210907c - messaging-section-xss - High	5
VWA210907d - listable-sysadmins - Medium	6
VWA210907e - customers-pws-md5 - Medium	7
VWA210907f - cookies-settings - Low	8

VWA Security Report

VWA210907a - login-system - Critical

Vulnerability Exploited: login-system

Severity: Critical

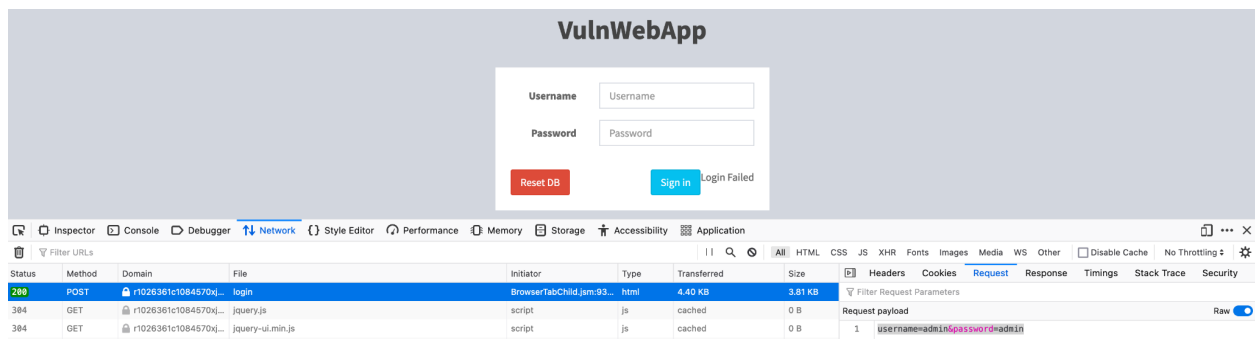
System: VWA Web Application

Vulnerability Explanation:

Pw was too simple to guess thus easily discoverable with automated tools, and no rate limit was detected for login trials. OWASP A2.

Vulnerability Walk-thru:

Found how the login page sends username and pw to the server.



Ran a script to try pws against a wordlist.

```
[+] Login Failed! {'username': 'guest', 'password': 'cowboy'}
[-] Login Failed! {'username': 'guest', 'password': 'silver'}
[-] Login Failed! {'username': 'guest', 'password': 'richard'}
[+] Login Found! {'username': 'guest', 'password': 'orange'}
This is a demo code used for this training.
root@a7586931f9f5:/home/workspace/tools# python bruteforce.py -U top-usernames-shortlist.txt -P top-passwords-shortlist.txt -d username=^USR^:password=^PWD^ -m post -f Failed http://0.0.0.0:3000/login
```

Recommendations:

Implement a pw strength policy and rate limit login trials.

VWA Security Report

VWA210907b - session-role-cookie - Critical

Vulnerability Exploited: session-role-cookie

Severity: Critical

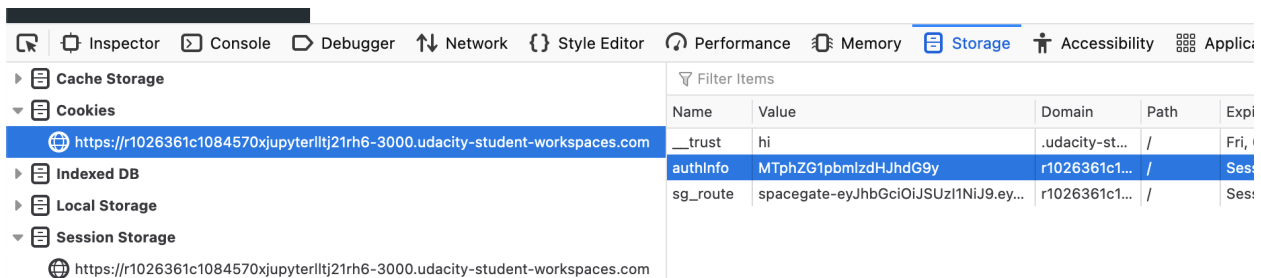
System: VWA Web Application

Vulnerability Explanation:

Permissions to retrieve data from server are based on roles and roles are assigned and persisted with a session cookie found vulnerable to deserialization attempts. OWASP A8.

Vulnerability Walk-thru:

Identified session cookie.



Assigned new role to session at cookie level.

```
Traceback (most recent call last):
  File "performbase64.py", line 22, in <module>
    decode(args.string)
  File "performbase64.py", line 11, in decode
    message_bytes = base64.b64decode(base64_bytes)
  File "/opt/conda/lib/python3.6/base64.py", line 87, in b64decode
    return binascii.a2b_base64(s)
binascii.Error: Incorrect padding
root@a7586931f9f5:/home/workspace/tools# python3 performbase64.py "1:admin"
"
MTphZG1pbG==
root@a7586931f9f5:/home/workspace/tools#
```

Recommendations:

Check for insecure deserialization and implement better access control to mitigate privilege escalation risks.

VWA Security Report

VWA210907c - messaging-section-xss - High

Vulnerability Exploited: messaging-section-xss

Severity: High

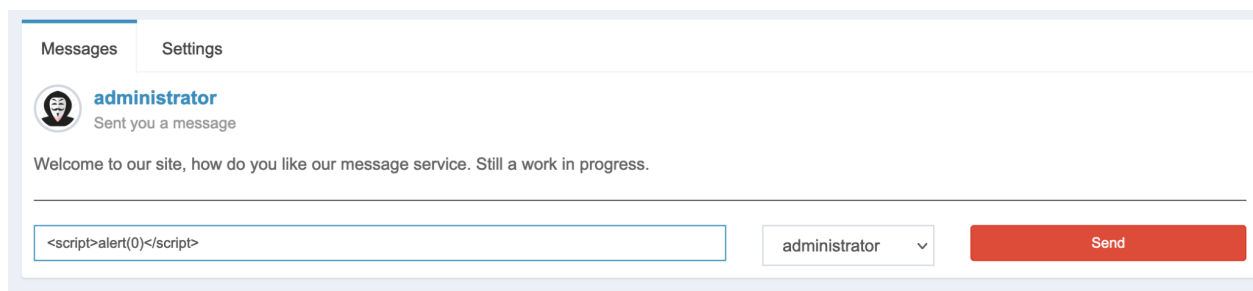
System: VWA Web Application

Vulnerability Explanation:

Messaging section includes a form vulnerable to xss as shown below, thus sensitive payloads or session data is exfiltrable and the attack would be difficult to log. OWASP A7.

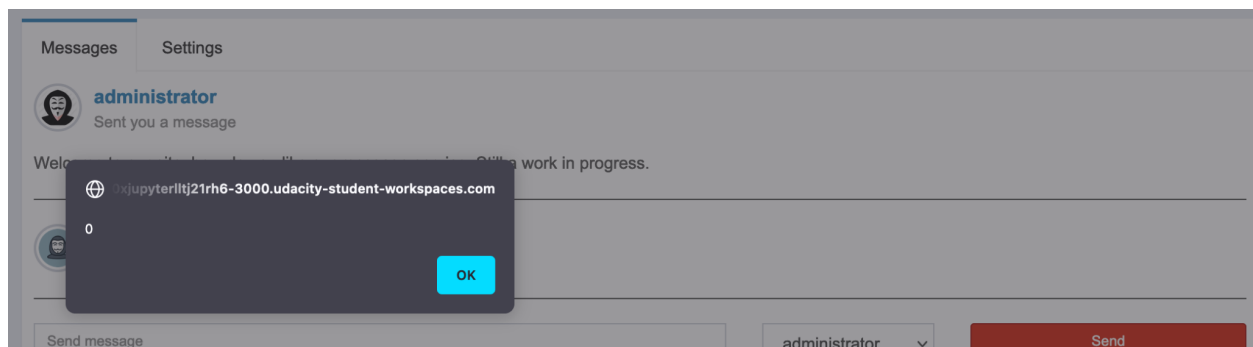
Vulnerability Walk-thru:

Tried to inject a js script.



The screenshot shows the 'Messages' tab of the VWA interface. At the top, there's a header with 'Messages' and 'Settings' tabs. Below this, a user profile for 'administrator' is shown with a message 'Sent you a message'. A welcome message reads: 'Welcome to our site, how do you like our message service. Still a work in progress.' Below the message, there is a text input field containing the payload '<script>alert(0)</script>'. To the right of the input field is a dropdown menu set to 'administrator' and a red 'Send' button.

Confirmed it worked as expected.



Recommendations:

Sanitize inputs from users and arginate xss risks at code level.

VWA Security Report

VWA210907d - listable-sysadmins - Medium

Vulnerability Exploited: listable-sysadmins

Severity: Medium

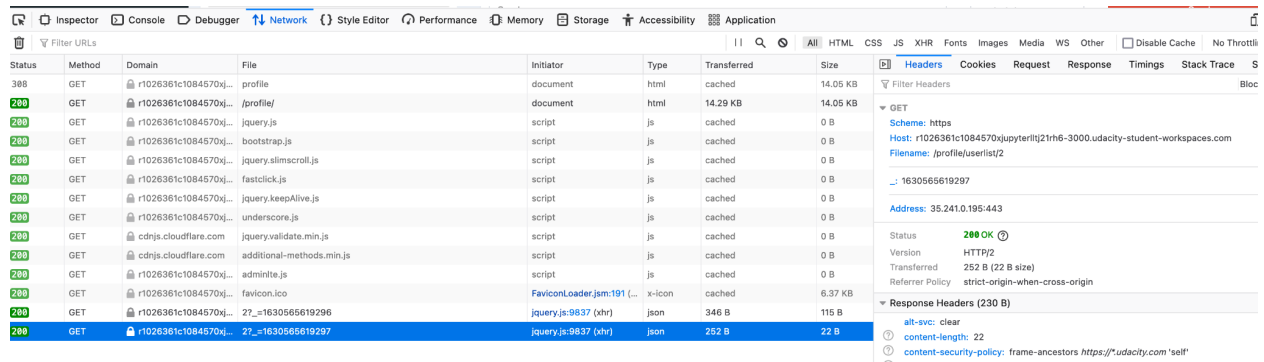
System: VWA Web Application

Vulnerability Explanation:

Access control for listing users (and admins) is broken and a full list of admins' usernames is obtainable. OWASP A5.

Vulnerability Walk-thru:

Found how client requests users list.



Status	Method	Domain	File	Initiator	Type	Transferred	Size
308	GET	r1026361c1084570xj...	/profile	document	html	cached	14.05 KB
200	GET	r1026361c1084570xj...	/profile/	document	html	14.29 KB	14.05 KB
200	GET	r1026361c1084570xj...	/jquery.js	script	js	cached	0 B
200	GET	r1026361c1084570xj...	/bootstrap.js	script	js	cached	0 B
200	GET	r1026361c1084570xj...	/jquery.slimscroll.js	script	js	cached	0 B
200	GET	r1026361c1084570xj...	/fastclick.js	script	js	cached	0 B
200	GET	r1026361c1084570xj...	/jquery.keepAlive.js	script	js	cached	0 B
200	GET	r1026361c1084570xj...	/underscore.js	script	js	cached	0 B
200	GET	cdnjs.cloudflare.com	/jquery.validate.min.js	script	js	cached	0 B
200	GET	cdnjs.cloudflare.com	/additional-methods.min.js	script	js	cached	0 B
200	GET	r1026361c1084570xj...	/adminlte.js	script	js	cached	0 B
200	GET	r1026361c1084570xj...	/favicon.ico	FaviconLoader.jam:191 (...)	x-icon	cached	6.37 KB
200	GET	r1026361c1084570xj...	27_1630565619296	jquery.js:9837 (xhr)	json	346 B	115 B
200	GET	r1026361c1084570xj...	27_1630565619297	jquery.js:9837 (xhr)	json	252 B	22 B

Response Headers (230 B)

- alt-svc: clear
- content-length: 22
- content-security-policy: frame-ancestors https://*.udacity.com/*self

Tried to change a parameter and circumvented access control.



```
0: 1
1: "administrator"
1: 2
1: "guest"
```

Recommendations:

Better enforce access control when serving sensitive information.

VWA Security Report

VWA210907e - customers-pws-md5 - Medium

Vulnerability Exploited: customers-pws-md5

Severity: Medium

System: VWA Web Application

Vulnerability Explanation:

Found sensitive data (weakly hashed customers' pws) was served to client for no reason. OWASP A3.

Vulnerability Walk-thru:

Analized server response and found hashed object, cracked.

The screenshot displays a web application interface with a 'Customers List' table and a 'Customer Info' modal. The table lists customers: paul, jake, dave, mike, and nick. The modal shows details for Customer ID 1: Username paul, First Name doe, Last Name pdoe. Below the interface is a network traffic log showing a GET request to /customers/id?_id=1630571304754, which returns a JSON response containing a hashed password.

ID	First Name
1	paul
2	jake
3	dave
4	mike
5	nick

Customer Info

Customer ID: 1

Username: paul

First Name: doe

Last Name: pdoe

Options: View, View, View, View, View

Network Traffic Log:

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	r1026361c1084570x...	/customers/	document	html	11.99 KB	11.76 KB
200	GET	r1026361c1084570x...	jquery.js	script	js	cached	273.79 KB
200	GET	r1026361c1084570x...	bootstrap.js	script	js	cached	73.71 KB
200	GET	r1026361c1084570x...	jquery.slimscroll.js	script	js	cached	13.51 KB
200	GET	r1026361c1084570x...	fastclick.js	script	js	cached	25.36 KB
200	GET	r1026361c1084570x...	jquery.KeepAlive.js	script	js	cached	1.73 KB
200	GET	r1026361c1084570x...	underscore.js	script	js	cached	66.95 KB
200	GET	cdnjs.cloudflare.com	jquery.validate.min.js	script	js	cached	22.72 KB
200	GET	cdnjs.cloudflare.com	additional-methods.min.js	script	js	cached	18.03 KB
200	GET	r1026361c1084570x...	adminlte.js	script	js	cached	29.05 KB
200	GET	r1026361c1084570x...	favicon.ico	FaviconLoader.jsm:191 (...)	x-icon	cached	6.37 KB
200	GET	r1026361c1084570x...	/customers/id?_id=1630571304754	jquery.js:9837 (xhr)	json	353 B	122 B
200	GET	r1026361c1084570x...	?_id=1630571304754	jquery.js:9837 (xhr)	json	291 B	61 B

Response:

```
{
  "0": 1,
  "1": "paul",
  "2": "doe",
  "3": "pdoe",
  "4": "d8578edf8458ce06fbc5bb76a58c5ca4"
}
```

Recommendations:

Serve back info to client on a need-to-know bases and store sensitive pws hashed with stronger and more up to date algorithms (ie sha256 w/ salting).

VWA Security Report

VWA210907f - cookies-settings - Low

Vulnerability Exploited: cookies-settings

Severity: Low

System: VWA Web Application

Vulnerability Explanation:

Session cookies not flagged as 'secure' and their usage not restricted to parent site. OWASP A6.

Vulnerability Walk-thru:

Checked for session cookies' properties.

VulnWebApp administrator admin

Dashboard

Page Controls: **Reset**

CPU TRAFFIC 90%

LIKES 41,410

SALES 760

NEW MEMBERS 2,000

Monthly Recap Report

Sales: 1 Jan, 2014 - 30 Jul, 2014

Goal Completion

- Add Products to Cart: 160/200
- Complete Purchase: 310/400
- Visit Premium Page: 480/800
- Send Inquiries: 250/500

Inspector **Cache Storage** **Cookies** **Indexed DB** **Local Storage** **Session Storage**

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last
__trust	hi	.udacity-st...	/	Fri, 02 Sep 2022 0...	9	false	true	None	Thu, 02 Sep 2021 08:22:03 GMT
authinfo	MTphZG1pbG==	1026361c1...	/	Session	20	false	false	None	Thu, 02 Sep 2021 08:22:03 GMT
sg_route	spacegate-ey...	1026361c1...	/	Session	573	true	true	None	Thu, 02 Sep 2021 08:22:03 GMT

Application Panel

authinfo: "MTphZG1pbG=="

Created: "Thu, 02 Sep 2021 08:22:03 GMT"

Domain: "https://1026361c1084570xjupyterllj21h6-3000.udacity-student-workspaces.com"

Expires / Max-Age: "Session"

HostOnly: true

HttpOnly: false

Last Accessed: "Thu, 02 Sep 2021 08:25:12 GMT"

Path: "/"

SameSite: "None"

Secure: false

Size: 20

Recommendations:

Help reducing the attack surface by preventing cross-site usage of session cookies and allow them to be sent over https only.