#BugBounty — AWS S3 added to my "Bucket" list!

Avinash Jain (@logicbomb_1) Follow

Jan 16, 2018 · 2 min read

Hi Guys,

You might remember the Million Dollar Instagram Bug that allowed security researcher Wes Wineberg to access every single image and account on Instagram. This was only possible because he had gained access to *Instagram's S3 bucket*, where the company stored everything from source code to images. In this particular blog, I would be explaining you "How misconfigured AWS storage bucket can be a huge security risk".

Recently during my bug hunting, I came across a misconfigured AWS S3 bucket of an Indian E-commerce Company which gave me full access to their S3 bucket, allowing them to download, upload and overwrite files. Let's dig deeper into this and see how I was able to do so—So , while in search for some security vulnerabilities in the website , I came across a career page from where users can apply for the relevant jobs and upload their resume. I started testing it out and found an endpoint (let's name the company out as xyz)-

"*https://xyz.s3.amazonaws.com/career%2Ftest.pdf*" and there was no ACL restriction set , any non-authenticated user could simply access any file and below was the curl request —

curl -XGET 'https://xyz.s3.amazonaws.com/career/test.pdf'

and the response was the content of test.pdf. Similarly , I discovered that "PUT" method was enabled on the S3 bucket and I could simply write any file onto the S3 bucket , it was publicly writable.

curl -XPUT -d 'HACKED' 'https://xyz.s3.amazonaws.com/career/test.pdf'

I ran some bruteforcing over filename and I was able to read the resume content of other users. :) , Now, I have to take this one level up , my next target was to list down and read all the files that were available onto S3 bucket. I connected to s3 command line and run the following command

```
“root@logicbomb~# aws s3 ls s3://xyz.s3.amazonaws.com”
```

and to no surprise, I was able to access the complete s3 bucket , all the CVs/Resume of users (also there were more sensitive data and directories) were publicly accessible and readable :) I tried some more commands -

```
root@logicbomb:~# aws s3 rm s3://xyz.s3.amazonaws.com/career/test.pdf

delete: s3://xyz.s3.amazonaws.com/career/test.pdf
```

and I was also able to delete the files also.

As a conclusion resides that Misconfigured S3 bucket may take your organisation to expose sensitive data.

Mitigation—*Advise you to promptly review your S3 buckets and their contents to ensure that you are not inadvertently making objects available to users that you don’t intend. For reference, you can read the below link —*

Amazon S3 Security: master S3 bucket polices and ACLs

Welcome to part 8 of my AWS Security Series. This week I shall be looking at some of the security features around the…

cloudacademy.com

Report details-

08-Dec-2017—Bug reported to the concerned company.

29-Dec-2017—Bug was marked fixed.

01-Jan-2018—Re-tested and confirmed the fix.

07-Jan-2018—Awarded by company.

Thanks for reading!

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

AWS Bug Bounty Penetration Testing Vulnerability Hacking

165 claps

Avinash Jain (@logicbomb_1)

Follow

Lead Infrastructure Security Engineer @groferseng | DevSecops | Part time BugBounty Hunter | Acknowledged by Google, NASA, Yahoo, United Nations, BBC etc.

InfoSec Write-ups

Follow

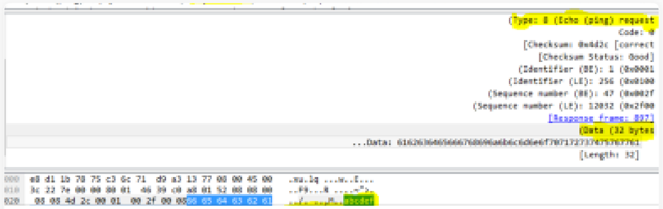
A collection of write-ups from the best hackers in the world on topics ranging from bug bounties and CTFs to vulnhub machines, hardware challenges and real life encounters. In a nutshell, we are the largest InfoSec publication on Medium. #sharingiscaring

More from InfoSec Write-ups

Writing a Password Protected Bind Shell (Linux/x64)

0x0FFB347
Mar 8 · 5 min read

246



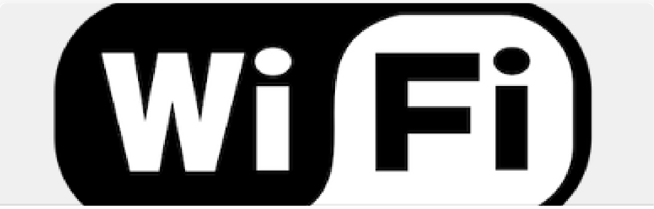
```
(Type: 8 (Echo (ping) request
Code: 0
[Checksum: 0x4d2c [correct
[Checksum Status: Good]
(Identifier (BE): 1 (0x0001
(Identifier (LE): 256 (0x0100
(Sequence number (BE): 47 (0x002f
(Sequence number (LE): 12032 (0x2f00
[Response frame: 897]
(Data (32 bytes
...Data: 6162636465666768696a6b6c6d6e6f7071727374757677616263646566676869
[Length: 32]

000  a8 d1 1b 78 75 c3 6c 71  d9 a3 13 77 00 00 45 00   ..xu.lq ...w..E...
010  3c 22 7e 00 00 00 01  46 39 c0 a8 01 52 00 00 00   ..F9...R ......
020  08 00 4d 2c 00 01  00 2f 00 00 66 65 64 63 62 61   ../.....abcdef
```

More from InfoSec Write-ups

Ping Power — ICMP Tunnel

Nir Chako
Dec 17, 2018 · 8 min read

488



More from InfoSec Write-ups

How to Make a Captive Portal of Death

Trevor Phillips
Dec 18, 2018 · 6 min read

280

Responses

Write a response...