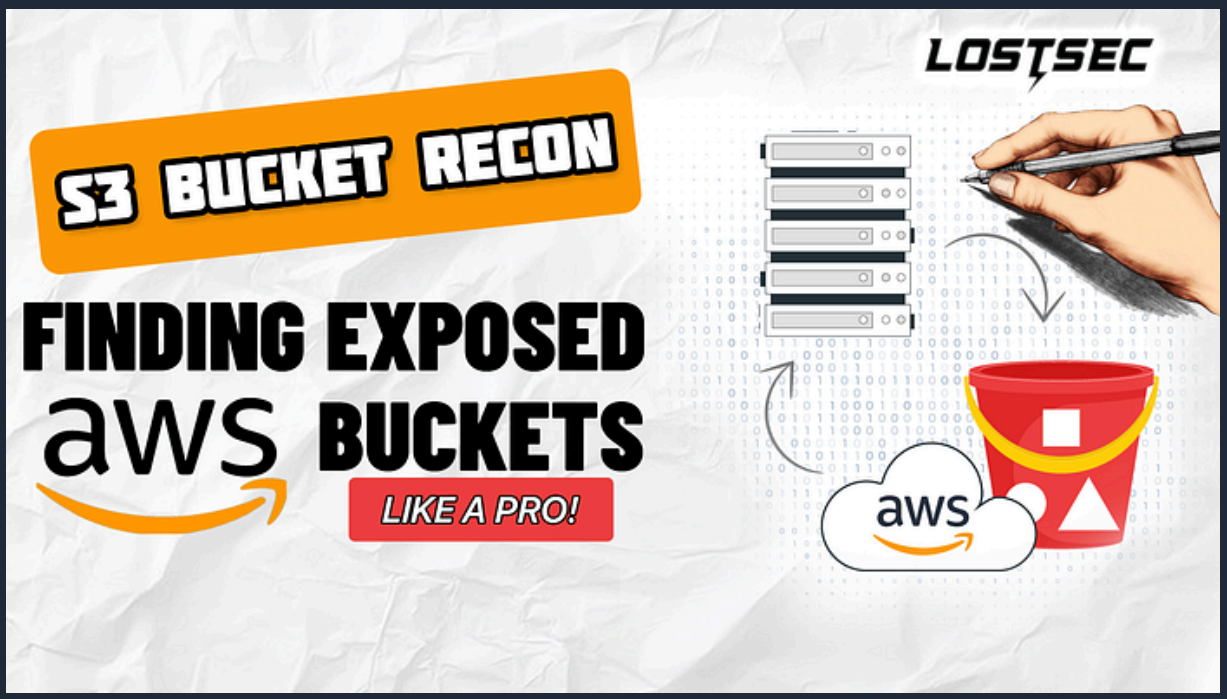


S3 Bucket Recon: Finding Exposed AWS Buckets Like a Pro!

From Discovery to Exploitation: A Complete Guide to S3 Bucket Recon

coffinxp

Follow

InfoSec Write-ups androidstudio ~7 min read ·
February 26, 2025 (Updated: February 26, 2025) · Free: No

A Step-by-Step Guide to Identifying and Exploiting Misconfigured AWS Buckets

Introduction

to serious security vulnerabilities. In this guide we'll explore how to audit S3 environments, uncover exposed buckets, analyze permissions and mitigate security risks. Using AWS tools and open-source scanners you'll learn to identify vulnerabilities before they become threats.

What is S3 Bucket Reconnaissance?

S3 bucket reconnaissance refers to the process of identifying and investigating publicly accessible or misconfigured AWS S3 buckets that may expose sensitive data. Developer or Security professional can use these techniques to help organizations to secure their cloud storage.

Table of Contents

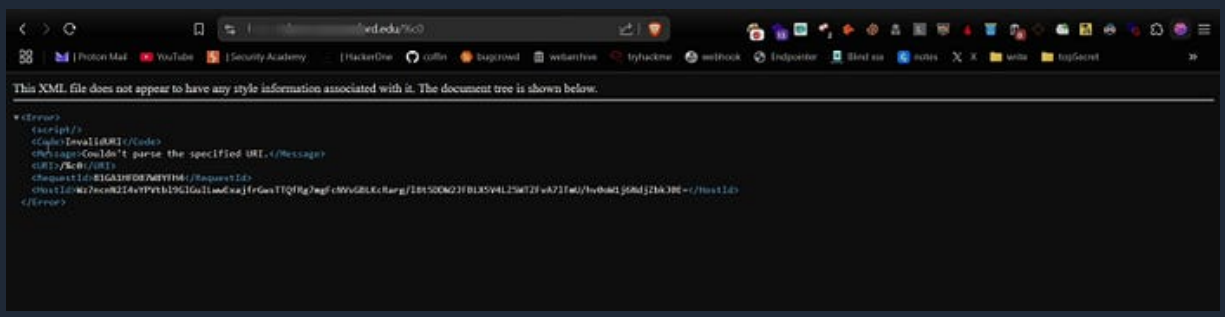
Copy

Manual Methods for Identifying S3 Buckets

Using Browser URL Inspection:

One of the simplest ways to check if a website is hosted on AWS is by entering the following in the browser URL bar:

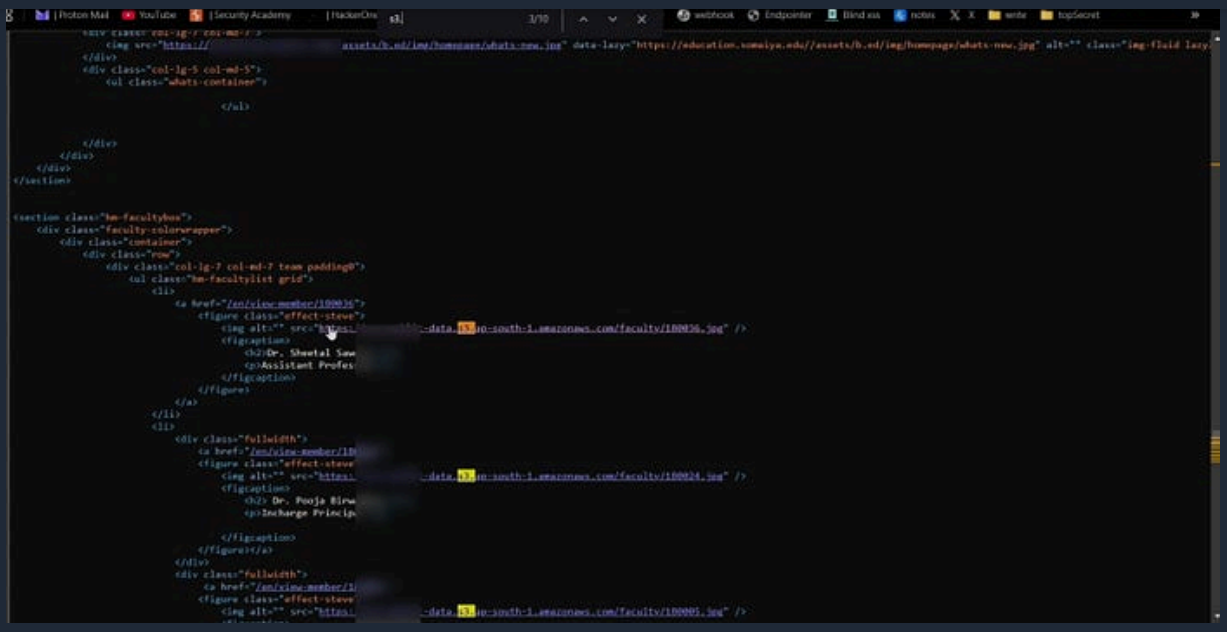
```
Copy

%c0
```

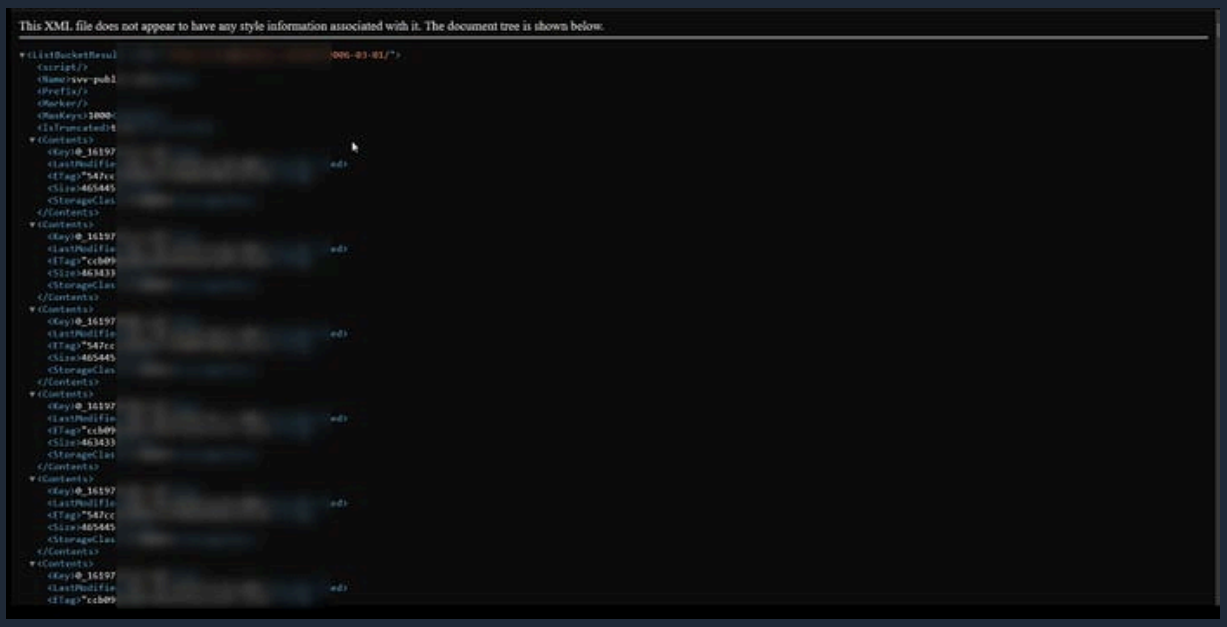


If you encounter an XML error the website is likely hosted on Amazon AWS. To verify further use browser extensions like Wappalyzer to check for AWS-related technologies.

Checking the Source Code

Inspect the website source code and search for "s3" to find any hidden S3 bucket URLs. if you found any just open and check if there bucket listing is enabled.





Google Dorking for AWS S3 Buckets

Google dorking helps uncover exposed S3 buckets. you can use the following dork to find open s3 buckets:

Copy

```
site:s3-external-1.amazonaws.com "target.com"
site:s3.dualstack.us-east-1.amazonaws.com "target.com"
site:amazonaws.com inurl:s3.amazonaws.com
site:s3.amazonaws.com intitle:"index of"
site:s3.amazonaws.com inurl:".s3.amazonaws.com/"
site:s3.amazonaws.com intitle:"index of" "bucket"

(site:*.s3.amazonaws.com OR site:*.s3-external-1.amazonaws.com OR site:*
```

If bucket listing is enabled you'll be able to view the entire directory and its files. If you see an "Access Denied" message it means the bucket is private.

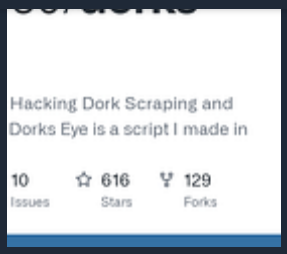
Automating Google Dorking with DorkEye

DorkEye automates Google dorking making reconnaissance faster by quickly extracting multiple AWS URLs for analysis.

Dork Scraping and Searching Script. Dorks...

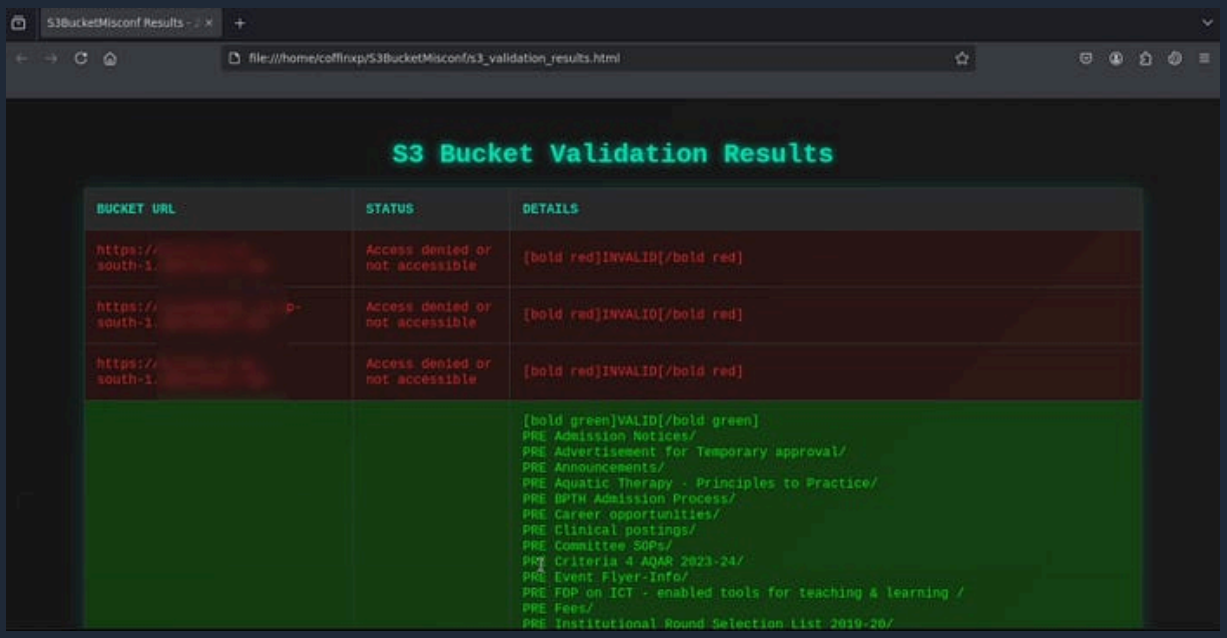
Dorks Eye Google Hacking Dork Scraping and Searching Script. Dorks Eye is a script I made in python 3. With this tool...

github.com



Using S3Misconfig for Fast Bucket Enumeration

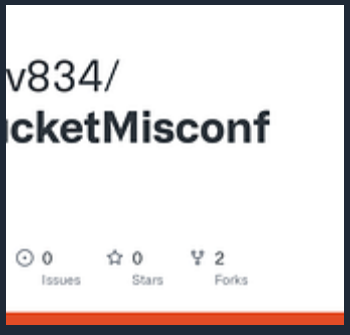
S3Misconfig scans a list of URLs for open S3 buckets with listing enabled and saves the results in a user friendly HTML format for easy review.



GitHub - Atharv834/S3BucketMisconf

Contribute to Atharv834/S3BucketMisconf development by creating an account on GitHub.

github.com

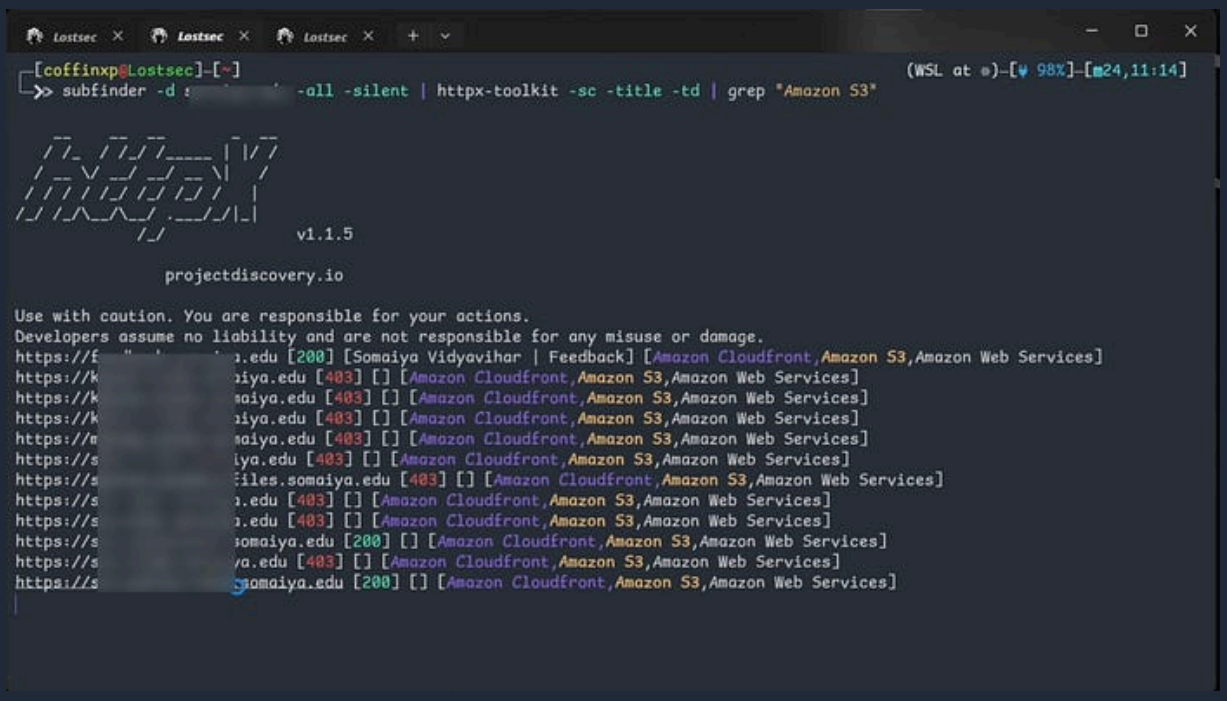


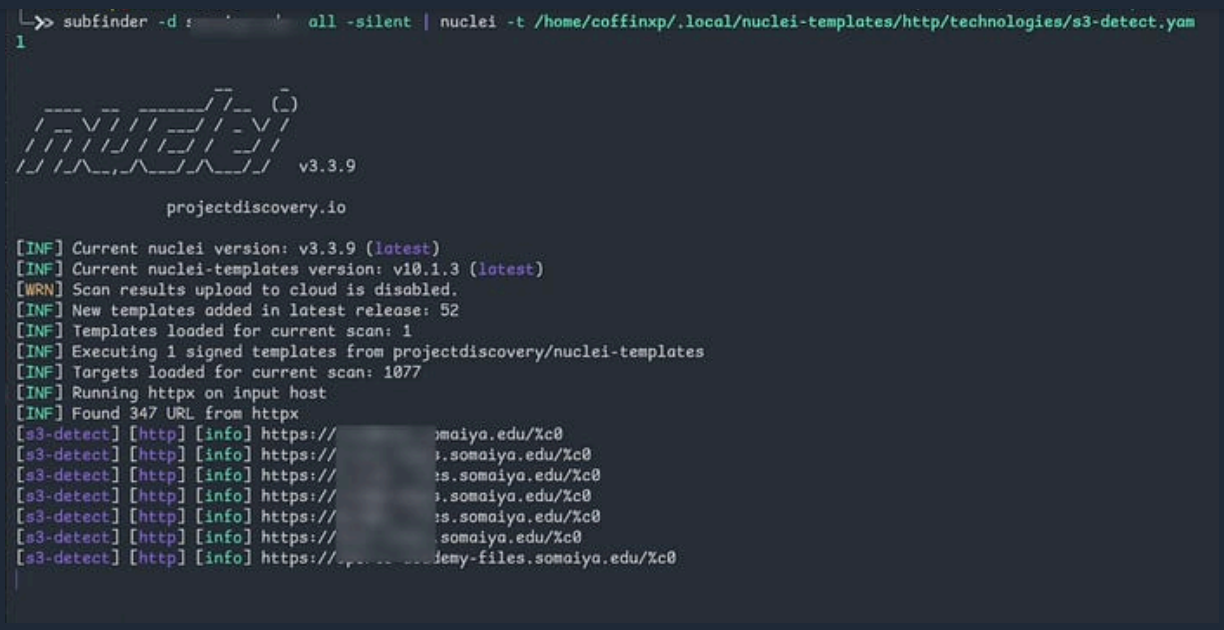
Finding S3 Buckets with HTTPX and Nuclei

You can use the HTTPX command along with the Nuclei tool to quickly identify all S3 buckets across subdomains saving you significant time in recon.

Copy

```
using subfinder+HTTPX
subfinder -d target.com -all -silent | httpx-toolkit -sc -title -td | gr

Nuclei template:
subfinder -d target.com -all -silent | nuclei -t /home/coffinxp/.local/n
```





Extracting S3 URLs from JavaScript Files

Next we'll use the Katana tool to download JavaScript files from target subdomains and extract S3 URLs using the following grep command:

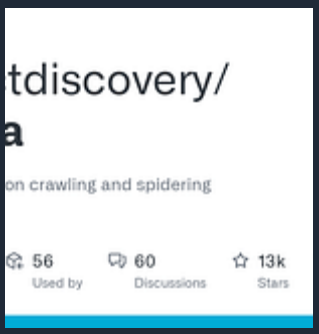
Copy

```
katana -u https://site.com/ -d 5 -jc | grep '\.js$' | tee alljs.txt
cat alljs.txt | xargs -I {} curl -s {} | grep -oE 'http[s]?://[^"]*\.s3\
```

GitHub - projectdiscovery/katana: A next-generation crawling and spidering framework.

A next-generation crawling and spidering framework. - projectdiscovery/katana

github.com

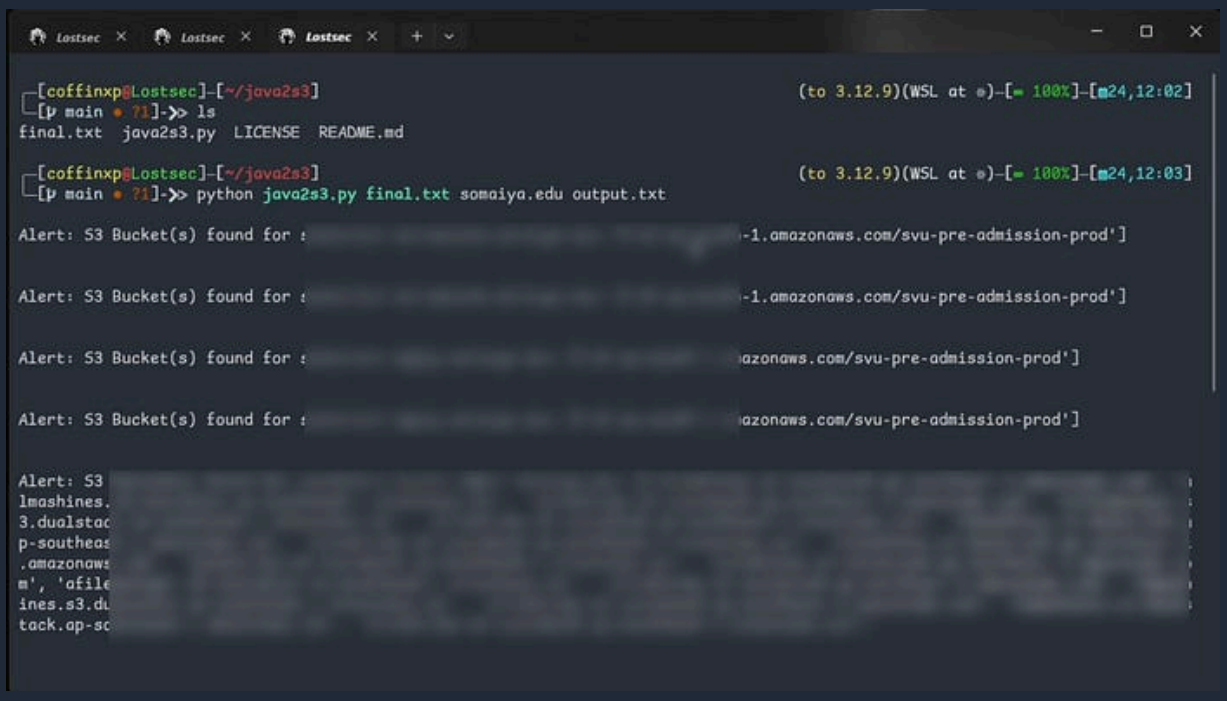


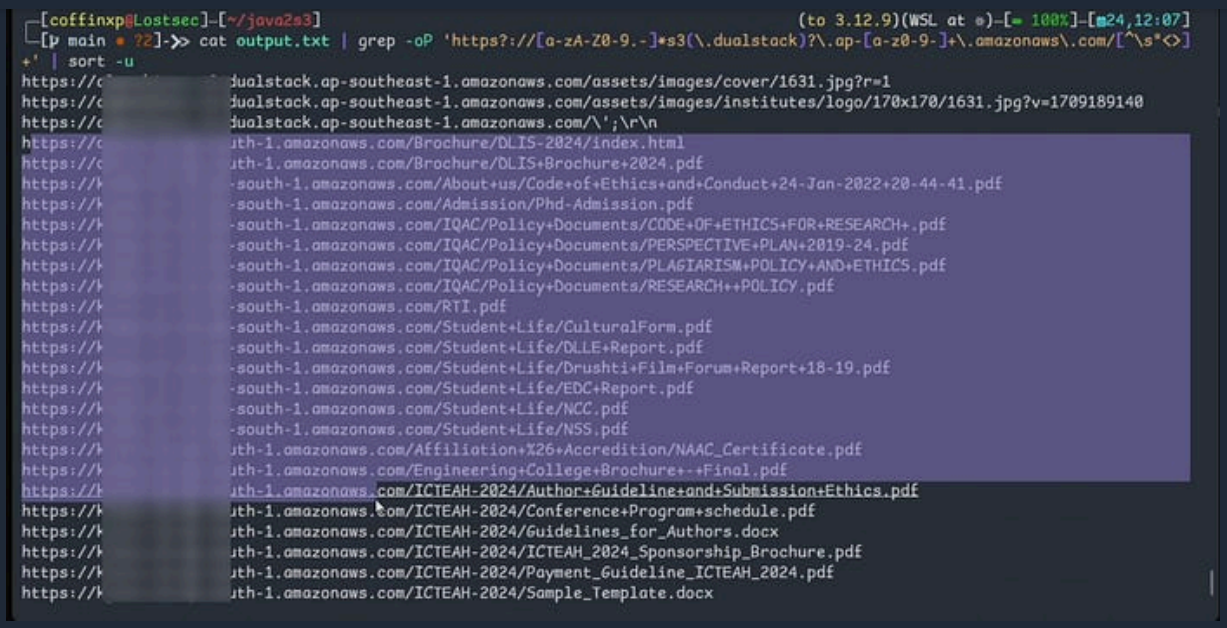
Using java2s3 tool to find s3 urls in js files

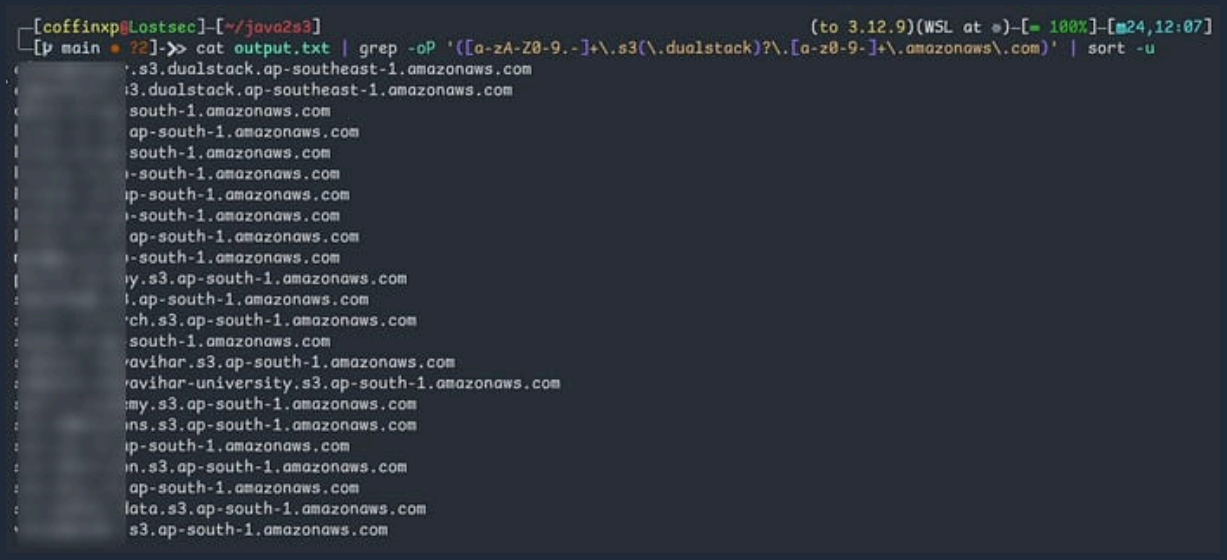
Alternatively you can use this powerful approach to extract all S3 URLs from JavaScript files of subdomains. First combine Subfinder and HTTPX to generate the final list of subdomains then run the java2s3 tool for extraction.

Copy

```
subfinder -d target.com -all -silent | httpx-toolkit -o file.txt
cat file.txt | grep -oP '(?<=https?:\/\/).*'
python java2s3.py input.txt target.com output.txt
cat output3.txt | grep -E "S3 Buckets: \['[^]]+"
cat output.txt | grep -oP 'https?://[a-zA-Z0-9.-]*s3(\.dualstack)?\.ap-[
cat output3.txt | grep -oP '([a-zA-Z0-9.-]+\.s3(\.dualstack)?\.[a-z0-9-]
```



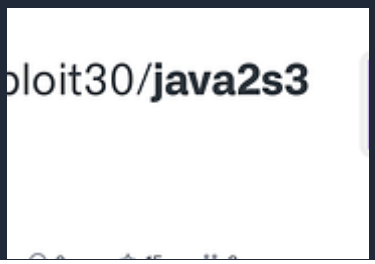




after this you can use use the S3Misconfig tool to identify publicly accessible S3 buckets with listing enabled by sending all these s3 urls to the tool

GitHub - mexploit30/java2s3

Contribute to mexploit30/java2s3 development by creating an account on GitHub.

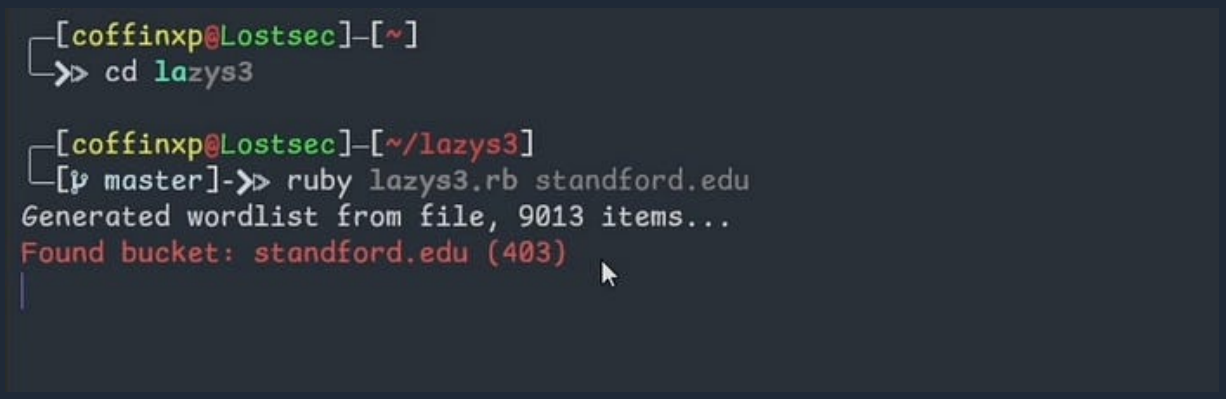


Brute-Forcing S3 Bucket with LazyS3

You can also use this LazyS3 tool — it's basically a brute force tool for AWS S3 buckets using different permutations. you can run the following command by specifying the target domain

Copy

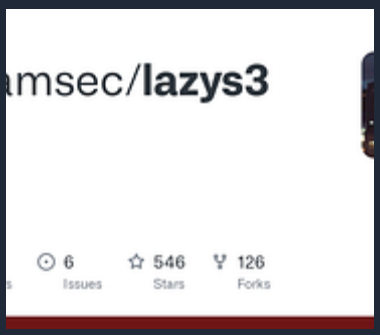
```
ruby lazys3.rb <COMPANY>
```



GitHub - nahamsec/lazys3

Contribute to nahamsec/lazys3 development by creating an account on GitHub.

github.com

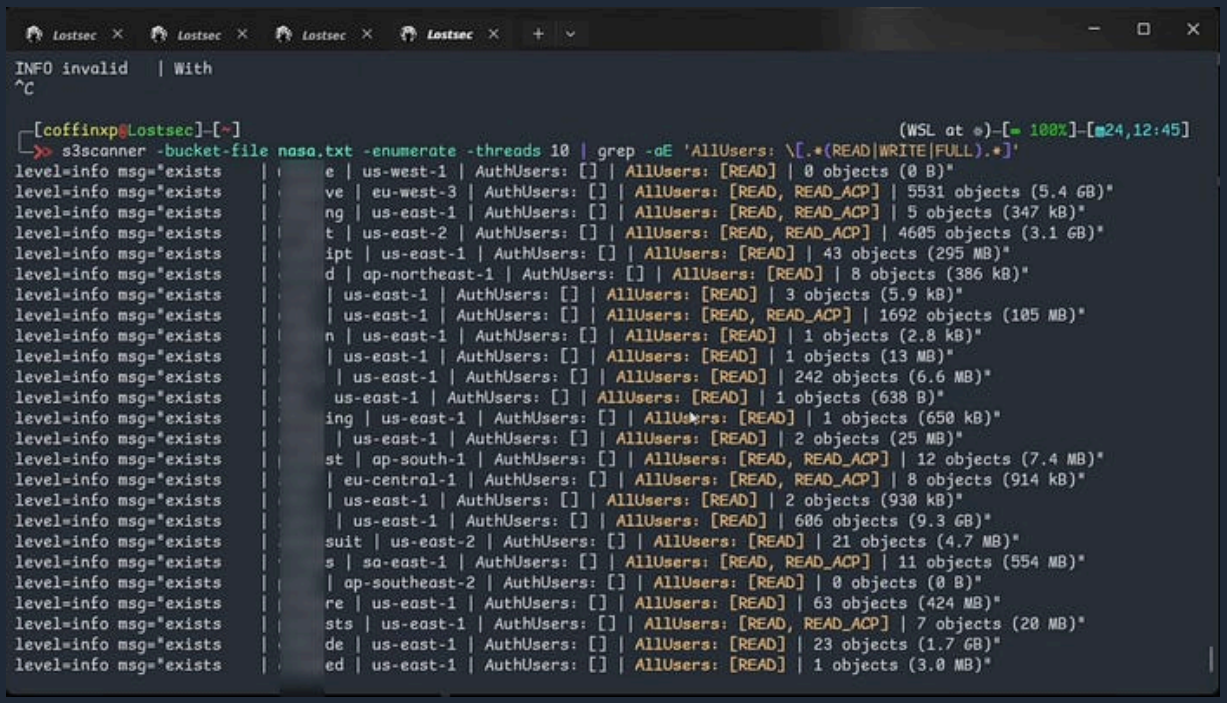


Using Cewl + S3Scanner to find open buckets

Next use the Cewl tool to generate a custom wordlist from the target domain. Then run S3Scanner with the list to identify valid and

using the AWS CLI.

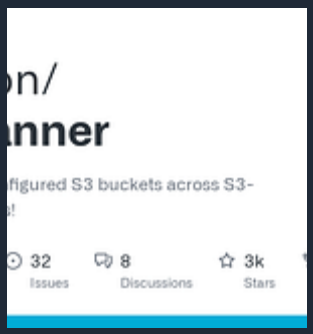
```
Copy

cewl https://site.com/ -d 3 -w file.txt

s3scanner -bucket-file file.txt -enumerate -threads 10 | grep -aE 'AllUs
```



GitHub - sa7mon/S3Scanner: Scan for misconfigured S3 buckets across S3-compatible APIs!

Scan for misconfigured S3 buckets across S3-compatible APIs! - sa7mon/S3Scanner

github.com

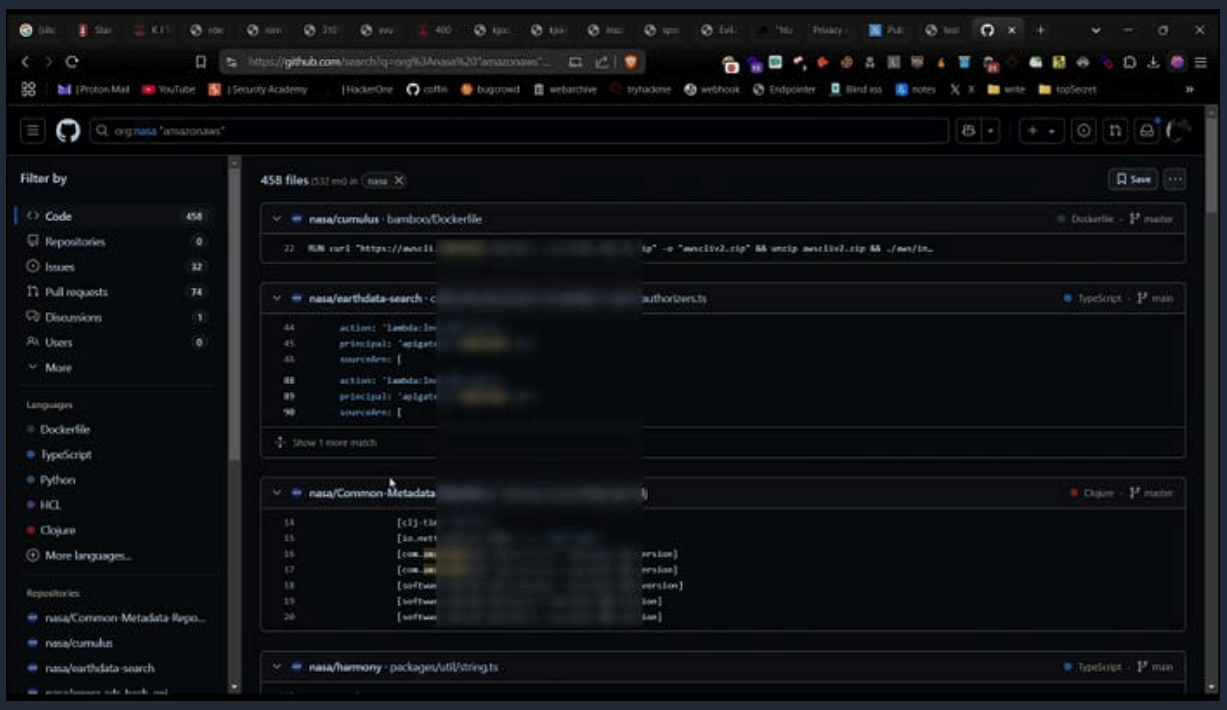


Extracting S3 Buckets from GitHub Repositories

you discover sensitive data report it to bug bounty programs.

Copy

```
org:target "amazonaws"
org:target "bucket_name"
org:target "aws_access_key"
org:target "aws_access_key_id"
org:target "aws_key"
org:target "aws_secret"
org:target "aws_secret_key"
org:target "S3_BUCKET"
```



Websites for Public S3 Bucket Discovery

Use these websites to search for files in public AWS buckets by keyword. Download and inspect the contents and if you find any sensitive files report them responsibly:

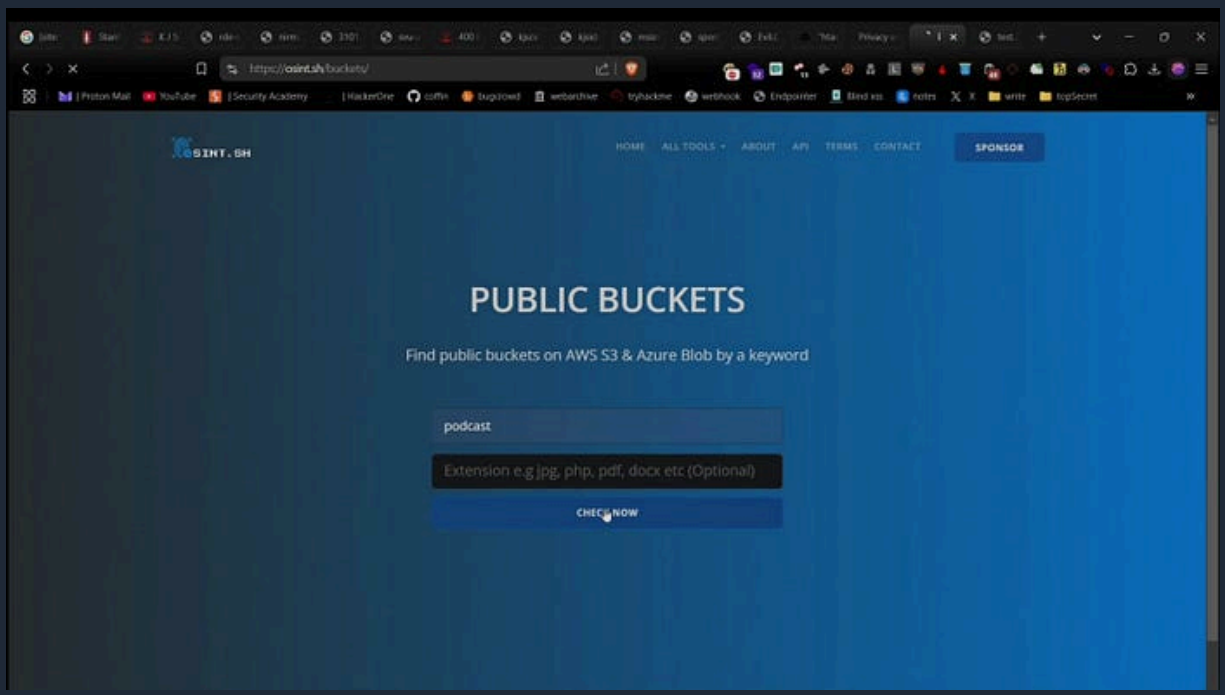
Public Buckets by GrayhatWarfare

Edit description

grayhatwarfare.com

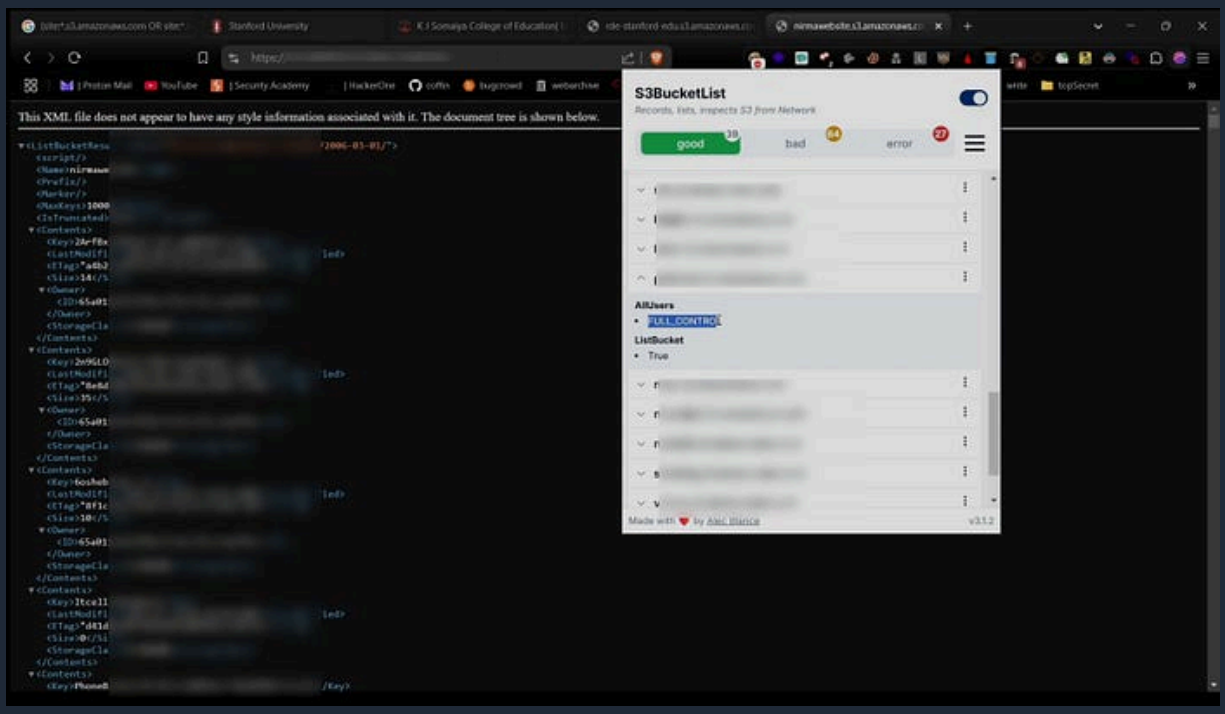
> osint.sh

https://osint.sh/buckets/



Finding Hidden S3 URLs with Extensions

The S3BucketList Chrome extension scans web pages for exposed S3 URLs helping researchers quickly identify misconfigured buckets



S3BucketList - Chrome Web Store

Search, lists, and checks S3 Buckets found in network requests

google.com

AWS S3 Bucket Listing & File Management

Easily manage AWS S3 buckets with these AWS CLI commands. These commands help security researchers, penetration testers and cloud administrators list, copy, delete and download files for efficient storage management and security assessments.

Reading Files:

```
aws s3 ls s3://[bucketname] --no-sign-request
```

Copying Files:

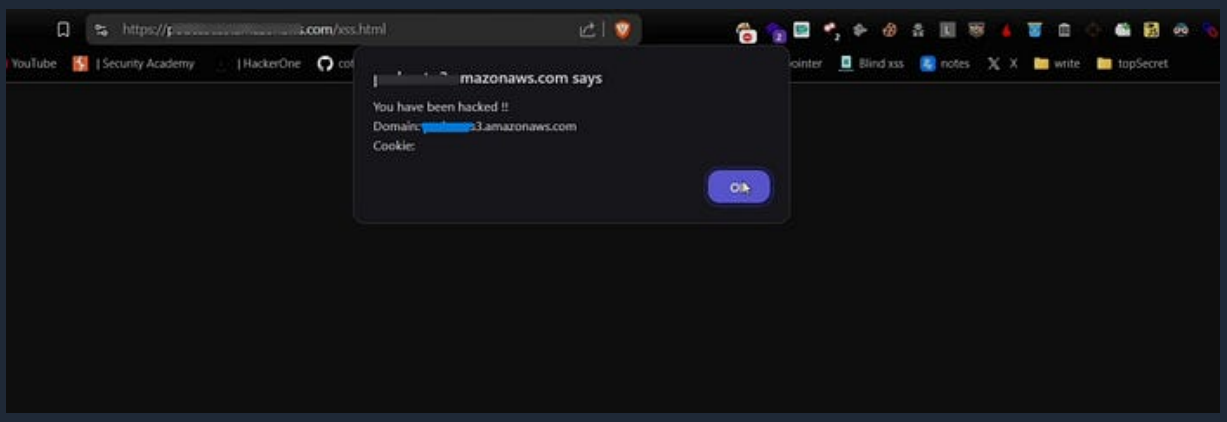
```
Copy
aws s3 cp file.txt s3://[bucketname] --no-sign-request
```

Deleting Files:

```
Copy
aws s3 rm s3://[bucketname]/file.txt --no-sign-request
```

Downloading All Files:

```
Copy
aws s3 cp s3://[bucketname]/ ./ --recursive --no-sign-request
```



> Buckets with "Full Control" permission allow file uploads and deletions which could lead to security risks. Always follow

Securing S3 Buckets

Organizations should follow best practices to prevent unauthorized access:

- Enable bucket policies and restrict access.
- Disable public ACLs unless necessary.
- Monitor logs using AWS CloudTrail.
- Implement encryption for sensitive data.

> *You can also watch this video where I showed the complete practicle of this method:*

loading soon..

Conclusion

S3 bucket reconnaissance is essential for ethical hackers and security professionals. Identifying and securing misconfigured buckets helps organizations strengthen their cloud security and prevent data leaks

> The content provided in this article is for educational and informational purposes only. Always ensure you have proper authorization before conducting security assessments. Use this information responsibly

#bug-bounty #bug-bounty-tips #aws #technology #amazon-web-services