

From the Outside In: How I Uncovered the Cybersecurity Failures of Thousands of Companies

Josephine Pfeiffer, 03/2023



Some background on me

- Coding, tinkering around with Linux for ~9 years
- Most interested in (hybrid) cloud, SRE, cybersecurity
- Previously TPM, SRE at Sygnum
- Currently Cloud Native Consultant at Red Hat

<https://josie.lol>

Disclaimer

The opinions expressed in this presentation are solely those of the presenter and do not necessarily reflect the views or policies of the presenter's past, current or future employers.

The information presented is for educational and informational purposes only and should not be construed as professional advice.

The presenter takes no responsibility for any actions taken based on the information provided during this presentation.

Once upon a time...

I was bored on PTO waiting to start my new job.

Researching data breaches & started to wonder what is floating around in public...

...what about S3 buckets and `.tfstate` files?



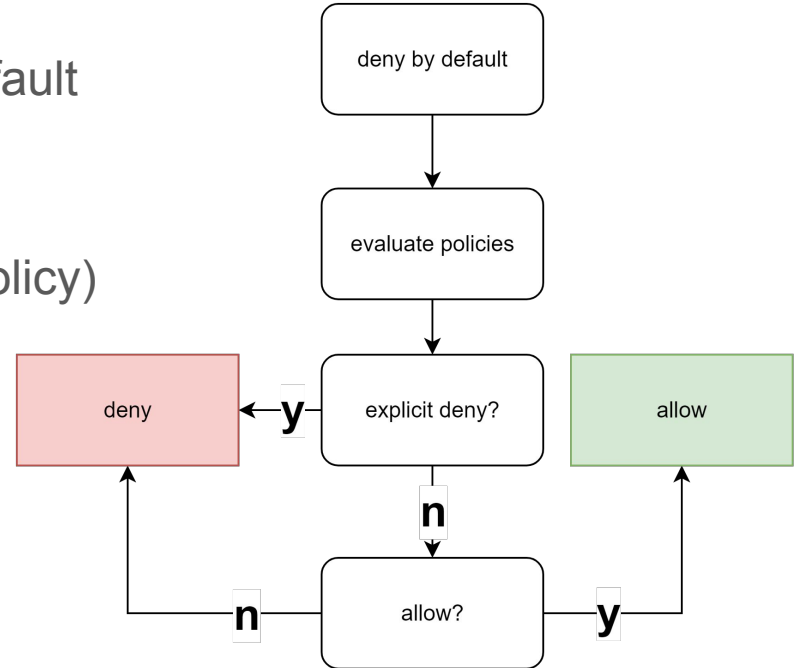
Some facts on S3 endpoints to begin with

- Be unique across all of Amazon S3
- Be between 3 and 63 characters long
- Not contain uppercase characters
- Start with a lowercase letter or number

```
https://\/[a-z0-9]([-a-z0-9]{2,61}[a-z0-9])?.s3.amazonaws.com\/
```

S3 bucket access controls

- The bucket policy is always private by default
- One explicit “deny” policy trumps “allow” policies at other levels
(IAM, S3 object/bucket ACL, S3 bucket policy)



Scraping endpoints (brief for legal reasons)

- Public? (y/N)
- Paths/prefixes make things more complicated
 - Traversing paths, recursively scan for filenames, extensions (`.tfstate`, `production`, `.env`, `secret.yaml`, etc.)



What I found

- In total, I scanned through ~308k AWS S3 buckets.
- Within only a few minutes, I could freely look through production secrets for thousands of large, international companies.



What I found

A US lottery company

Directly stored customer and transaction data as `.csv` files in a publicly accessible S3 bucket.

```
Retailer, ██████████
Program, ██████████
Start, 05/01/2021
End, 05/05/2021
Total Card Sales, 8280.00
Total Card Reloads, 2200.00
Total Card Reversal, 0.00
Total Sales, 10480.00
Total Redemptions, 5943.50

Id, Date, Time, Transaction Type, Amount, Card UID, Store, Terminal
215439, 05/01/2021, 07:03:34, Card Sale, 20.00, 00000000000b51, 248, 2481
214972, 05/01/2021, 07:10:01, Card Sale, 20.00, 00000000000b43, 1081, 10815
226520, 05/01/2021, 07:14:17, Card Sale, 20.00, 00000000000b06, 327, 3275
220040, 05/01/2021, 07:19:19, Card Sale, 20.00, 00000000000ac6, 332, 3321
214906, 05/01/2021, 07:20:14, Card Sale, 20.00, 00000000000b42, 245, 2455
214989, 05/01/2021, 07:20:14, Card Sale, 20.00, 00000000000b44, 1081, 10812
231192, 05/01/2021, 07:26:44, Card Sale, 20.00, 00000000000b2f, 1211, 12115
220160, 05/01/2021, 07:35:34, Card Sale, 20.00, 00000000000ac7, 332, 3321
215006, 05/01/2021, 07:38:44, Card Reload, 20.00, 00000000000b44, 1081, 10812
231245, 05/01/2021, 07:44:28, Card Reload, 20.00, 00000000000b2f, 1211, 12113
```

What I found

A US lottery company

The same bucket contained
.tfstate files for all environments
containing database credentials, TLS
certificates, encryption keys, and
sensitive networking configuration.


```
1 1
2 "version": 4,
3 "terraform_version": "1.0.11",
4 "serial": 22,
5 "lineage": "e4f[redacted]a3a",
6 "outputs": {
7   "password": {
8     "value": "wl[redacted]zP5",
9     "type": "string",
10    "sensitive": true
11  },
12  "rds_address": {
13    "value": "[redacted]-production.ca[redacted]16.us-west-2.rds.amazonaws.com",
14    "type": "string"
15  },
16  "rds_name": {
17    "value": "[redacted]-production",
18    "type": "string"
19  },
20  "password": {
21    "value": "ZA[redacted]H9",
22    "type": "string",
23    "sensitive": true
24  },
25  "rds_address": {
26    "value": "[redacted]-production.ca[redacted]16.us-west-2.rds.amazonaws.com",
27    "type": "string"
28  },
29  "rds_name": {
30    "value": "[redacted]-production",
31    "type": "string"
32  }
33 }
```

Blog post

- Reached out to companies (none responded)
- Published blog post
- ~5k reads on medium
- Discussed at length on r/cybersecurity, and hackernews

From the Outside In: How I Uncovered the Cybersecurity Failures of Thousands of Companies (and you can too)

The author has discovered the ignored negligence involved in the 2016, 2017 and 2018 breaches on how to better protect sensitive information and how to prevent future breaches. Here's how the company failed.



In both digital and physical, companies are increasingly relying on technology to store sensitive information such as customer data, financial records, medical records, legal records, and personal communication. Unfortunately, many businesses are failing to properly secure this information, leaving it vulnerable to a wide range of data breaches.

Leaving the Door Wide Open

If you have a website, mobile application, or cloud application, you are more likely to be hacked than you think. Some examples include AWS S3, Google Cloud Storage, etc.

However, to avoid this, you need to take a proactive approach by implementing a comprehensive security strategy. This includes regular security audits, patch management, and secure configuration of all services.

Real World Examples of Sensitive Data Exposed to the Public

Over the last few years, there have been hundreds of thousands of public data breaches, which can contain the sensitive and critical information of individuals and organizations. These include, but are not limited to, credit card numbers, Social Security numbers, and more. Here are some other interesting examples of data breaches:


- Healthcare:** In 2015, a major healthcare provider exposed 800,000 patient records, including names, addresses, and Social Security numbers.
- Government:** In 2016, a major government agency exposed 100,000 records, including names, addresses, and phone numbers.
- Retail:** In 2017, a major retail company exposed 100,000 customer records, including names, addresses, and phone numbers.

These examples could be used to illustrate the consequences of large data breaches and the importance of proactive security measures.

To learn more about this topic, I'll share some specific examples.


An International digital healthcare company from MENA

Since a public record file is publicly accessible in Saudi Arabia, the file contains API keys and admin credentials for a CRM, patient database, and other internal systems.

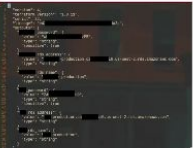


A US battery company

Energy storage companies and consumer devices use the same protocols. Available in public.




The source code is available for all to see, including the source code. This is a public record of the source code, and it is available for anyone to see.




A real-time market data API provider

This is a real-time market data API provider, and it is available for anyone to see.



A Web3 software development agency

This is a Web3 software development agency, and it is available for anyone to see.



This is a real-time market data API provider, and it is available for anyone to see.

The Consequences of Poor Cybersecurity for Businesses and Customers

When businesses neglect cybersecurity, they are putting their customers' information at risk. This can lead to financial loss, reputational damage, and legal consequences.

During the "data breach" phase, you may be exposed to a wide range of consequences, including:

- Financial loss: Data breaches can result in significant financial loss, including the cost of investigation, legal fees, and customer compensation.
- Reputational damage: Data breaches can result in significant reputational damage, leading to a loss of customer trust and a decline in sales.
- Legal consequences: Data breaches can result in significant legal consequences, including fines and lawsuits.

The most common consequence is the loss of sensitive information for a business or individual. If this information has been hacked, it can be used to steal money, identity, and other sensitive information. A data breach can also result in a loss of customer trust, which can be difficult to rebuild.

[REDACTED] · 1st

Security Intelligence Engineer @ AWS.

TUESDAY

[REDACTED] · 9:55 PM







Hi Josephine,


I work in AWS's threat intel team, and I just read your medium post regarding unsecured sensitive data in S3 buckets--would you be open to discussing your findings in more detail? I'd like to surface this with folks in AWS Security.


Thanks!

-- [REDACTED]

Results

From Amazon Web Services, Inc. <no-reply-aws@amazon.com>      More 

To Me  2/11/23, 17:18

Subject Amazon S3 to automatically apply bucket security best practices for all new buckets [AWS Account: 

Hello,

We are reaching out to inform you that starting in April 2023 Amazon S3 will change the default security configuration for all new S3 buckets. For new buckets created after this date, S3 Block Public Access will be enabled, and S3 access control lists (ACLs) will be disabled.

The majority of S3 use cases do not need public access or ACLs. For most customers, no action is required. If you have use cases for public bucket access or the use of ACLs, you can disable Block Public Access or enable ACLs after you create an S3 bucket. In these cases, you may need to update automation scripts, CloudFormation templates, or other infrastructure configuration tools to configure these settings. To learn more, read the AWS News blog [1] and What's New announcement [2] on this change or visit our user guide for S3 Block Public Access [3] and S3 Object Ownership to disable ACLs [4]. Also, see our user guide for AWS CloudFormation on these settings [5][6].

If you have any questions or concerns, please reach out to AWS Support [7].

[1] <https://aws.amazon.com/blogs/aws/heads-up-amazon-s3-security-changes-are-coming-in-april-of-2023/>
[2] <https://aws.amazon.com/about-aws/whats-new/2022/12/amazon-s3-automatically-enable-block-public-access-disable-access-control-lists-buckets-april-2023/>
[3] <https://docs.aws.amazon.com/AmazonS3/latest/userguide/access-control-block-public-access.html>
[4] <https://docs.aws.amazon.com/AmazonS3/latest/userguide/about-object-ownership.html>
[5] <https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-s3-bucket-publicaccessblockconfiguration.html>
[6] <https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-s3-bucket-ownershipcontrols.html>
[7] <https://aws.amazon.com/support>

Sincerely,
Amazon Web Services

Amazon Web Services, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com is a registered trademark of Amazon.com, Inc. This message was produced and distributed by Amazon Web Services Inc., 410 Terry Ave. North, Seattle, WA 98109-5210

Reference: https://phd.aws.amazon.com/phd/home?region=us-east-1#/event-log?eventID=arn:aws:health:global::event/S3/AWS_S3_OPERATIONAL_NOTIFICATION/AWS_S3_OPERATIONAL_NOTIFICATION_ddf06bb412a9a056822e3f69eaa80749adc3fc4e95a17eb48d5c124903ea5dd&eventTab=details

TL;DR

Default security config for new buckets now have public access disabled.

Key takeaways

- This is a really simple attack vector
 - Scary how easy it was: Awareness is important
- Impact can be catastrophic for businesses & customers
 - Financial, reputational, privacy violations
- Humans are the weakest link
 - The tech works fine, if used right
- A lot of companies are doing it right:
 - <1% of the scraped endpoints were public
(excluding buckets that were obviously meant to be public – e.g. static website hosting)
- Cloud providers are taking steps
 - Making it even harder for users to leak data

Q&A

