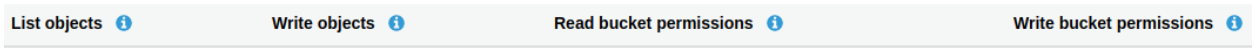
Insecure Cloud Storage

Stuck? Join our Discord for help! https://discord.gg/wvfe3XJ

Today we'll look at insecure cloud storage, more specifically, insecure amazon web services(AWS) s3 buckets. Today, a lot more companies are moving their computing and infrastructure to the 'cloud':

- Scalable: Most cloud service providers have the ability to not only create a large amount of resources on demand but they can also automatically manage creation of these resources.
- High availability: To ensure resources don't go down, cloud providers allow companies to manage failover by duplicating resources in different regions

While there are a large number of cloud providers(Microsoft Azure, Oracle Cloud), we'll focus on Amazon Web Services as they are fairly popular. AWS provides the ability for clients to store a lot of data using a service called Simple Storage Service(S3). Files are stored on what are called buckets and these buckets can have insecure permissions:



Here's a break down of the following permissions:

- List objects: user with permissions can list the files in the bucket
- Write objects: user with permissions can add/remove files on the bucket
- Read bucket permissions: users with permissions can read files on the bucket
- Write bucket permissions: users with permissions can edit files on the bucket

The permissions above apply to the bucket, but an administrator can also assign specific permissions to files/objects in the bucket.

An administrator can assign permissions in the following ways:

- For specific users

- For everyone

In the past, the default S3 permissions were weak and S3 buckets would be publicly accessible but AWS changed this to block public access by default.

Enumeration

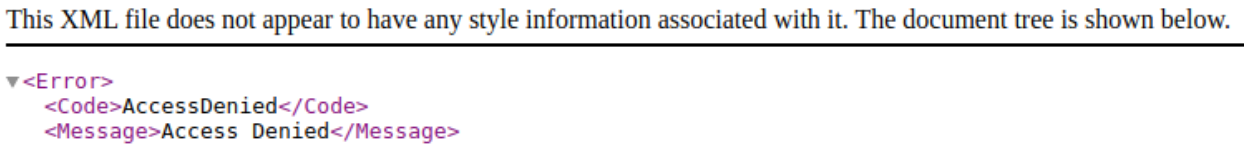
The first part of enumerating s3 buckets is having an s3 bucket name. How would you find an s3 bucket name:

- Source code on git repositories
- Analysing requests on web pages
    - Some pages retrieve static resources from s3 buckets
- Domain name of product names:
    - If a product or domain is called “servicename” then the s3 bucket may also be called “servicename”

Once we have an s3 bucket, we can check if it’s publicly accessible by browsing to the URL. The format of the URL is:
bucketname.s3.amazonaws.com

We talked about AWS supporting multiple regions before. Even though S3 buckets are global, we can still access them on their region:
bucketname.region-name.amazonaws.com

If the bucket is not accessible, you would get a similar image



If the bucket is accessible, then you will be able to view all the files on the bucket.

While uncommon, s3 buckets can be configured in a way such that any authenticated users can access the bucket(this is uncommon because an administrator would specifically have to write a policy to allow this). In this case, you would be able to list the objects in a bucket using the AWS CLI.

If you’ve found objects on an s3 bucket, you would want to download them to view their contents. You do this using the AWS CLI. To use the AWS CLI, you need to create an account.

Once you have created an AWS account, you can check the contents of the bucket using the command
aws s3 ls s3://bucket-name

To download the files, you can use the command:
aws s3 cp s3://bucket-name/file-name local-location

Alternatively, you can also use the following method to access a file:
bucketname.region-name.amazonaws.com/file-name

*Is this realistic: **very**. There have been a lot of breaches due to s3 bucket misconfigurations:*

- *https://arstechnica.com/information-technology/2017/05/defense-contractor-stored-intelligence-data-in-amazon-cloud-unprotected/*
- *https://www.infosecurity-magazine.com/news/accenture-leaked-data-another-aws/*
- *https://www.infosecurity-magazine.com/news/data-leak-exposes-750k-birth-cert/*