



FREE eBook

LEARNING amazon-s3

Free unaffiliated eBook created from
Stack Overflow contributors.

#amazon-s3

Table of Contents

About	1
Chapter 1: Getting started with amazon-s3	2
Remarks.....	2
Summary from the Documentation	2
Language and Scripting Support	2
The S3 API	2
Versions	2
The Examples	2
Security	3
Versions.....	3
Examples.....	4
Installation of AWS CLI for accessing S3.....	4
AWS CLI S3 Commands List.....	5
Hello World Example using Java.....	7
Hello World Using PowerShell.....	8
Credits	9

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [amazon-s3](#)

It is an unofficial and free amazon-s3 ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official amazon-s3.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with amazon-s3

Remarks

Summary from the Documentation

From <http://docs.aws.amazon.com/AmazonS3/latest/dev/Welcome.html>

Amazon Simple Storage Service is storage for the Internet. It is designed to make web-scale computing easier for developers.

Amazon S3 has a simple web services interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of web sites. The service aims to maximize benefits of scale and to pass those benefits on to developers.

Language and Scripting Support

S3 is not a development language as such, but a cloud platform that supports web service requests. There is an assortment of [tools and SDK's](#) that are published by Amazon. The language SDK's provide transparent access to S3 by handling tasks such as web services requests, authentication, session management, token renewal, etc. There are also command line interfaces for [bash/windows/ios](#) and [powershell](#).

The S3 API

The S3 web services API has been [supported by competing vendors](#). This topic does not currently cover the API directly, so the examples in here would not be useful for building applications that connect to competing systems that leverage the S3 API.

Versions

As of 28 March 2017, the AWS CLI has 174 versions, which are cleanly documented in the [CLI Release Notes](#). Amazon S3 has 66 versions, of which some are to announce addition of a new region, and others are to add functionality. These are documented in the [S3 Release Notes](#).

The Examples

With respect to the examples shown so far in this "Getting Started" section, Amazon S3 is useful to developers for the following use cases:

- **Store or back up files** in a high-performing, durable system, thus offloading this task from non-cloud architectures: linux and windows file systems. It is expensive to recreate the durability and performance levels of S3 using on premise servers or EC2 instances.
- When network bandwidth is an issue, for example, in cases where multiple simultaneous users must download large files, moving data to S3 can be used as a way for an application to **mitigate bandwidth shortages to a datacenter or on-premises server**. This is a way of distributing a large code repository, virtual machine images, video, or software installers. User upload times and user download times can be improved. [For additional performance in very large user base scenarios, a content delivery system such as cloudfront can be used to cache files closer to the users.]
- **Your application needs to create or consume a big file** and you need a way to allow users to access or deposit it.
- **Your application distribution is very big** and you need to share it with users.
- **You are putting together a continuous delivery pipeline** and for example hosting portions of your website on Amazon S3.

At this point the examples do not show how to do the following:

- The examples, although faster and clearer than typing `aws s3 help`, do not mention some of the commands covered in `help`, such as `aws s3 website``.
- How to share or restrict user access. Without explicit restriction, the examples would work only for users sharing the same AWS account.
- How to secure data via encryption. Note that AWS does position S3 as having a higher level of security than data stored in EC2. [AWS Security Best Practices, August 2016, p. 27](#)

Security

AWS recommends viewing S3 as a secure platform:

Unless you have more stringent business or compliance requirements, you don't need to introduce additional layers of protection beyond those provided by the AWS secure global infrastructure. [ibid. p.2](#)

In their [security guide](#), AWS recommends using AWS authentication as suitable for S3. [ibid. p. 27](#)

Additionally, S3 provides server-side encryption or client-side encryption. Client side encryption is provided transparently by the AWS Java SDK; keys need not be stored on AWS. [ibid. p. 28](#)

Versions

version name	description	notes	release date
Amazon S3 on 2016-12-13	Adds London Region	notes	2016-12-13

Examples

Installation of AWS CLI for accessing S3

Installing aws cli in Ubuntu / Debian Instance

```
sudo apt-get install -y python-dev python-pip
sudo pip install awscli
aws --version
aws configure
```

Installing aws cli using python

Using pip you can install aws cli in windows, OS X and Linux

```
sudo pip install awscli
```

Configuring the AWS Command Line Interface

This section explains how to configure settings that the AWS Command Line Interface uses when interacting with AWS, such as your security credentials and the default region.

```
$ aws configure
AWS Access Key ID [None]: <Your access key >
AWS Secret Access Key [None]: <Your secret key>
Default region name [None]: us-west-2
Default output format [None]: json
```

Get the Access key and Secret key from the account page in AWS

Creating Buckets

Use the `aws s3 mb` command to create a new bucket. Bucket names must be unique and should be DNS compliant. Bucket names can contain lowercase letters, numbers, hyphens and periods

```
aws s3 mb s3://bucket-name
```

Removing Buckets

To remove a bucket, use the `aws s3 rb` command. By default bucket should be empty.

```
aws s3 rb s3://bucket-name
```

To remove a non-empty bucket, you need to include the `--force` option.

```
aws s3 rb s3://bucket-name --force
```

Listing Buckets

To list all buckets or their contents, use the aws s3 ls command

```
aws s3 ls
aws s3 ls s3://bucket-name
```

The following command lists the objects in bucket-name/path

```
aws s3 ls s3://bucket-name/path
```

Synchronize files between local file system and S3

```
aws s3 sync . s3://my-bucket/path
```

It will upload all the files in the current directory to S3. To download the files from S3 to the current directory execute

```
aws s3 sync s3://my-bucket/path .
```

AWS CLI S3 Commands List

List of commonly used S3 AWS CLI Commands

Create Bucket

```
aws s3 mb s3://bucket-name
```

Remove Bucket

```
aws s3 rb s3://bucket-name
```

List Buckets

```
aws s3 ls
```

List contents inside the bucket

```
aws s3 ls s3://bucket-name
```

List Bucket with a path

```
aws s3 ls s3://bucket-name/path
```

Copy file

```
aws s3 cp file.txt s3://my-bucket/
```

Synchronize files

```
aws s3 sync . s3://my-bucket/path
```

Delete local file

```
rm ./MyFile1.txt
```

Attempt sync without --delete option - nothing happens

```
aws s3 sync . s3://my-bucket/path
```

Sync with deletion - object is deleted from bucket

```
aws s3 sync . s3://my-bucket/path --delete
```

Delete object from bucket

```
aws s3 rm s3://my-bucket/path/MySubdirectory/MyFile3.txt
```

Sync with deletion - local file is deleted

```
aws s3 sync s3://my-bucket/path . --delete
```

Sync with Infrequent Access storage class

```
aws s3 sync . s3://my-bucket/path --storage-class STANDARD_IA
```

Copy MyFile.txt in current directory to s3://my-bucket/path

```
aws s3 cp MyFile.txt s3://my-bucket/path/
```

Move all .jpg files in s3://my-bucket/path to ./MyDirectory

```
aws s3 mv s3://my-bucket/path ./MyDirectory --exclude '*' --include '*.jpg' --recursive
```

List the contents of my-bucket

```
aws s3 ls s3://my-bucket
```

List the contents of path in my-bucket

```
aws s3 ls s3://my-bucket/path
```

Delete s3://my-bucket/path/MyFile.txt

```
aws s3 rm s3://my-bucket/path/MyFile.txt
```


Delete s3://my-bucket/path and all of its contents

```
aws s3 rm s3://my-bucket/path --recursive
```

Hello World Example using Java

This example attempts to create a bucket called 'hello-world' and, as the bucket hello-world has already been created by someone else in S3's global namespace, throws the following exception. Change 'hello-world' to something else to avoid the exception by creating a uniquely named bucket. [The new bucket so created can be deleted using the AWS console](#)

Exception in thread "main" com.amazonaws.services.s3.model.AmazonS3Exception: The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again. (Service: Amazon S3; Status Code: 409; Error Code: BucketAlreadyExists; Request ID: ...

```
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.CreateBucketRequest;
import com.amazonaws.services.s3.model.Bucket;

/** S3 "hello world" example. */
public class S3Hello {

    /** Name of hello-world bucket -- must be globally unique. The
     * bucket namespace is shared by all users of the system.
     */
    static final String BUCKET_NAME = "hello-world";

    /** Creates bucket
     * @param args Command line arguments
     */
    public static void main(final String[] args) {

        AmazonS3 s3 = AmazonS3ClientBuilder.defaultClient();

        CreateBucketRequest request
            = new CreateBucketRequest(BUCKET_NAME);

        Bucket bucket = s3.createBucket(request);
        System.out.println("S3 Hello World completed.");
    }
}
```

This example requires the following dependencies:

- Java installed with console working.
- AWS Java SDK installed. <https://aws.amazon.com/sdk-for-java/>
- Credentials file `credentials` set up in `.aws` under your home directory. <https://aws.amazon.com/developers/getting-started/java/>
- Credential to have admin or 'create bucket' rights in S3. <http://docs.aws.amazon.com/AmazonS3/latest/dev/using-with-s3-actions.html#using-with-s3-actions-related-to-buckets>

Hello World Using PowerShell

This example expects an error, as the hello-world bucket already exists and S3 uses a global namespace.

```
New-S3Bucket -BucketName "hello-world"
```

New-S3Bucket : The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again.

If you replace hello-world with something else that is unique, the bucket will be created without error, and you will get the following result:

CreationDate	BucketName
-----	-----
3/30/2017 11:43:03 PM	hello-world-832jklsdJF

This example requires the following dependencies:

- PowerShell. See <http://docs.aws.amazon.com/powershell/latest/userguide/pstools-getting-set-up.html>
- Credentials. These can be created using the AWS console. There are several options for managing these using PowerShell. Below is a simple example for setup. See <http://docs.aws.amazon.com/powershell/latest/userguide/specifying-your-aws-credentials.html>

```
PS C:\> Set-AWSCredentials -AccessKey AKIAIOSFODNN7LAJD8A  
-SecretKey "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEjw9JFKS3" -StoreAs default
```

Read [Getting started with amazon-s3](https://riptutorial.com/amazon-s3/topic/2879/getting-started-with-amazon-s3) online: <https://riptutorial.com/amazon-s3/topic/2879/getting-started-with-amazon-s3>

Credits

S. No	Chapters	Contributors
1	Getting started with amazon-s3	Community , Foolish Brilliance , John Meyer , Nithin K Anil