



# How To Use MyS3

Encrypt your files on the fly and sync with Amazon cloud

---

**Ove Bakken**

Revision 2  
September 7th, 2020  
Norway, Innlandet

REVISION 2

# Welcome!

Since you're here you're probably curious about how you use MyS3.

This user manual has two parts: The Amazon cloud resource setup (1) and how to use MyS3 (2). MyS3 itself is very simple (and intuitive) and shouldn't require reading a manual. But if you feel unsure you're welcome to study it, and to send me questions.

MyS3 is simple software, but also fairly new. Please let me know if you experience any problems you believe to be bugs. If you're running Windows you can use the console client or the graphical interface client. However, Linux and Mac OS users can only run the console client for now.

All executables can be found at [github.com/flaskevann/MyS3](https://github.com/flaskevann/MyS3).

If you don't trust the uploaded builds, feel free to download the necessary GitHub projects yourself and make your own. After installing the needed NET Core tools all you have to do is run the build command.

# Getting started

1) Do you have an Amazon Web Services (AWS) account? If not, you need to create one:  
[aws.amazon.com/premiumsupport/knowledge-center/create-and-activate-aws-account/](https://aws.amazon.com/premiumsupport/knowledge-center/create-and-activate-aws-account/)

*.. and then afterwards in this user guide:*

2) You setup your own resources in the Amazon cloud

Meaning you create your own space with Simple Storage Service (S3), and then you give yourself access with access credentials from Access Management (IAM). Everything can be done directly using AWS Management Console. It's very easy if you follow the instructions.

Don't worry too much about the expenses. S3 is very price competitive. You have to upload or download a good amount before even breaking \$1. Please see AWS S3 pricing for details:  
[aws.amazon.com/s3/pricing/](https://aws.amazon.com/s3/pricing/)

The rest of this user guide will assume you have access to AWS.

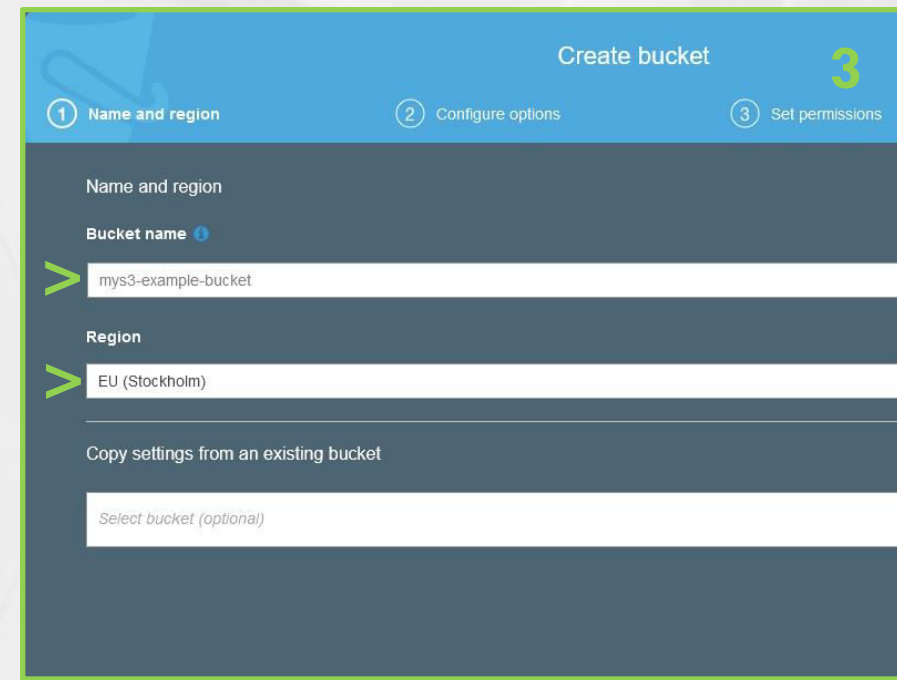
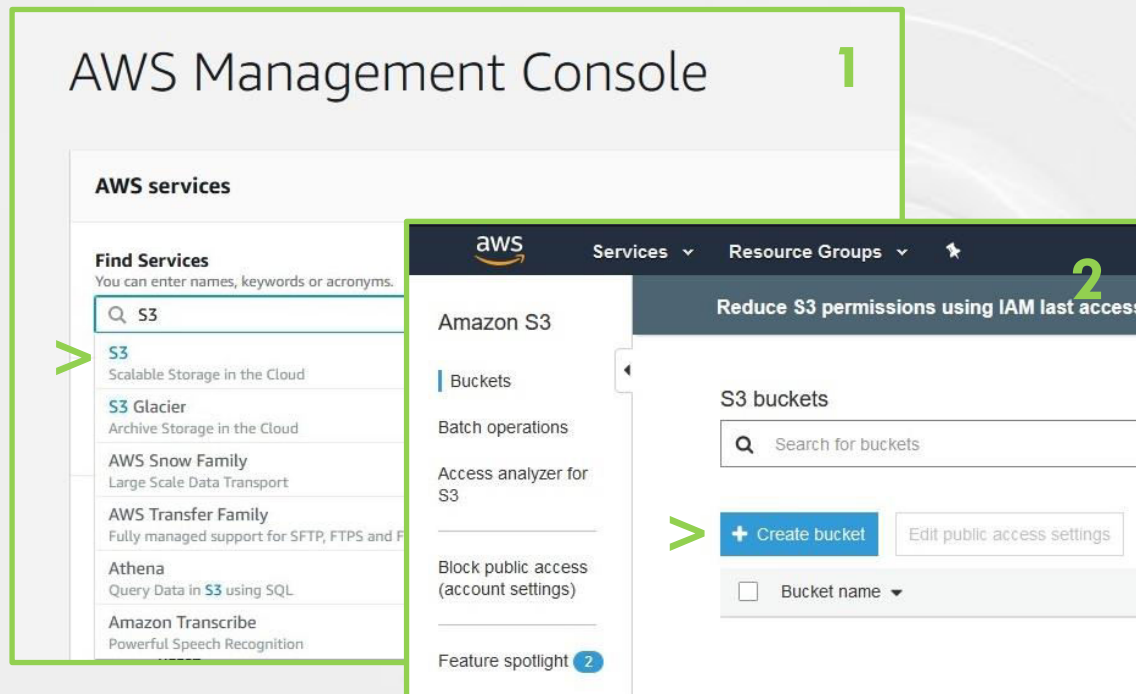
# Setup S3 Bucket

How To Use MyS3

Ove Bakken

First things first, you need to setup your own S3 space. (AWS calls this a 'bucket'.)

You log onto AWS Management Console (1), select S3 (2), and find a bucket name (3):



Region: Select closest or cheapest!



# Setup S3 Bucket

How To Use MyS3

Ove Bakken

Continue the setup. Do enable versioning (1), but use default settings for the rest:

The image displays three sequential screenshots of the AWS S3 bucket creation wizard, highlighting specific configuration steps with green annotations.

**Screenshot 1 (Step 1):** The 'Configure options' tab is active. A green arrow points to the 'Versioning' section, where the checkbox 'Keep all versions of an object in the same bucket' is checked. A green number '1' is placed next to this checkbox.

**Screenshot 2 (Step 2):** The 'Configure options' tab is active. A green arrow points to the 'Block public access (bucket settings)' section, where the checkbox 'Block all public access' is checked. A green number '2' is placed next to this checkbox.

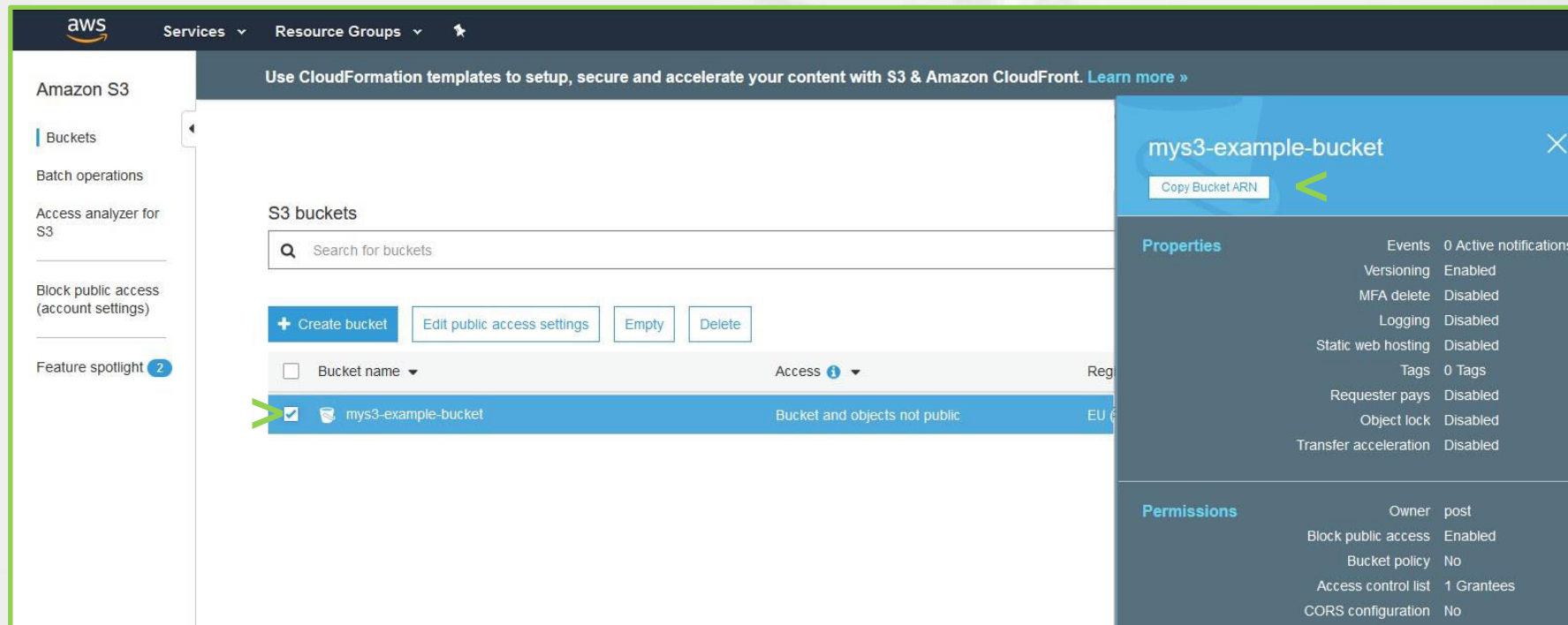
**Screenshot 3 (Step 3):** The 'Set permissions' tab is active. A green arrow points to the 'Block all public access' section, where the 'On' radio button is selected. A green number '3' is placed next to this section.

# Get S3 Bucket ARN

How To Use MyS3

Ove Bakken

After you've created your bucket (Yey!), copy bucket address (ARN) to a text document:



(Bucket ARN is needed when setting up your access credentials next!)

# Setup IAM User

How To Use MyS3

Ove Bakken

Now that you have your S3 bucket ready, you need a sort of user profile MyS3 can assume:

Go to AWS Management Console. Select IAM (1), Users (2), Add user (3) and start setup (4):

The image displays four sequential screenshots of the AWS Management Console, illustrating the steps to create an IAM user:

- Screenshot 1:** The AWS Management Console home page. The search bar is used to find the IAM service, which is highlighted with a green arrow and the number 1.
- Screenshot 2:** The IAM service page. The 'Users' link in the left-hand navigation menu is highlighted with a green arrow and the number 2.
- Screenshot 3:** The 'Add user' button in the IAM 'Users' section is highlighted with a green arrow and the number 3.
- Screenshot 4:** The 'Add user' setup page. The 'User name' field is filled with 'UserForMyS3' (highlighted with a green arrow and the number 1), and the 'Programmatic access' checkbox under 'Select AWS access type' is checked (highlighted with a green arrow and the number 2).

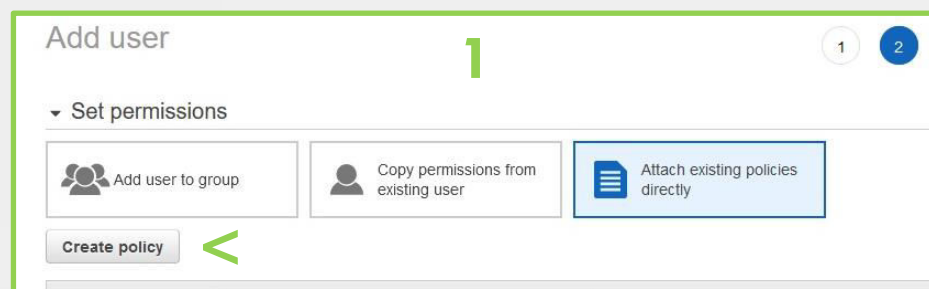
Select a user name and check Programmatic access.

# Setup IAM Policy

How To Use MyS3

Ove Bakken

Click create policy (1), this opens a new browser tab. Go to JSON (2) and put in text (3):



```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:AbortMultipartUpload",
        "s3:DeleteObjectVersion",
        "s3:ListBucketVersions",
        "s3:ListBucket",
        "s3:GetBucketVersioning",
        "s3:DeleteObject",
        "s3:PutBucketVersioning",
        "s3:GetObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3:::mys3-example-bucket",
        "arn:aws:s3:::mys3-example-bucket/*"
      ]
    }
  ]
}
```

3



Put in your own bucket ARN twice at bottom.

**Don't delete /\***



# Setup IAM Policy

How To Use MyS3

Ove Bakken

Give your policy a fitting name (1) and finish it. IAM should then say it's created (2).

Create policy 1

Review policy

Name\*  <  
Use alphanumeric and '+=, @-\_' characters. Maximum 128 characters.

Description   
Maximum 1000 characters. Use alphanumeric and '+=, @-\_' characters.

Summary

Service	Access level	Resource
Allow (1 of 238 services) <a href="#">Show remaining 237</a>		
S3	Limited: List, Read, Write	Multiple

aws Services Resource Groups ★

Identity and Access Management (IAM) 2

MyS3Policy has been created.

Create policy Policy actions

Filter policies Search

	Policy name	Type	Used as	Description
<input type="radio"/>	AccessAnalyzerServiceRole...	AWS managed	None	Allow Access A
<input type="radio"/>	AdministratorAccess	Job function	None	Provides full ac
<input type="radio"/>	AlexaForBusinessDeviceSe...	AWS managed	None	Provide device

(The policy giving access to your S3 bucket now exists, and only has to be added to MyS3's user.)

# Setup IAM User

How To Use MyS3

Ove Bakken

Go back to the browser tab where you began setting up your user and refresh the list.

Find your newly created policy by typing the first few letters, then check it and hit next:

The screenshot shows the AWS IAM console 'Add user' wizard, step 2: Set permissions. The 'Attach existing policies directly' option is selected. A search for 'my' shows one result, 'MyS3Policy', which is checked. The 'Next: Tags' button is highlighted.

aws Services Resource Groups

Add user 1 2 3 4 5

Set permissions

Add user to group Copy permissions from existing user Attach existing policies directly

Create policy

Filter policies my Showing 1 result

	Policy name	Type	Used as
<input checked="" type="checkbox"/>	MyS3Policy	Customer managed	None

Set permissions boundary

Cancel Previous Next: Tags

# Setup IAM User

How To Use MyS3

Ove Bakken

The next step is tags (1). Please skip this. Then comes a review (2) and finally the credentials screen (3) which gives you your access credentials:

Add user **1**

Add tags (optional)

IAM tags are key-value pairs you can add to your user. Tags can include user information, such as an email address, or can be descriptive of your job title. You can use the tags to organize, track, or control access for this user. [Learn more](#)

Key

Add new

You can add

Add user **3**

**Success**

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

[Download .csv](#)

User	Access key ID	Secret access key
▶  UserForMyS3	AKIA5MC54425SKSETO3V	SD1+smZuKccYYdqO0hyIP

Add user **2**

**Review**

Review your choices. After you create the user, you can view and download the autogenerated credentials.

**User details**

<b>User name</b>	UserForMyS3
<b>AWS access type</b>	Programmatic access - with an access key
<b>Permissions boundary</b>	Permissions boundary is not set

**Permissions summary**

The following policies will be attached to the user shown above.

Type	Name
Managed policy	<a href="#">MyS3Policy</a>

**Tags**

No tags were added.

Please download csv-file and keep it safe. (It gives access to your S3 bucket!)

# The MyS3 Clients

How To Use MyS3

Ove Bakken

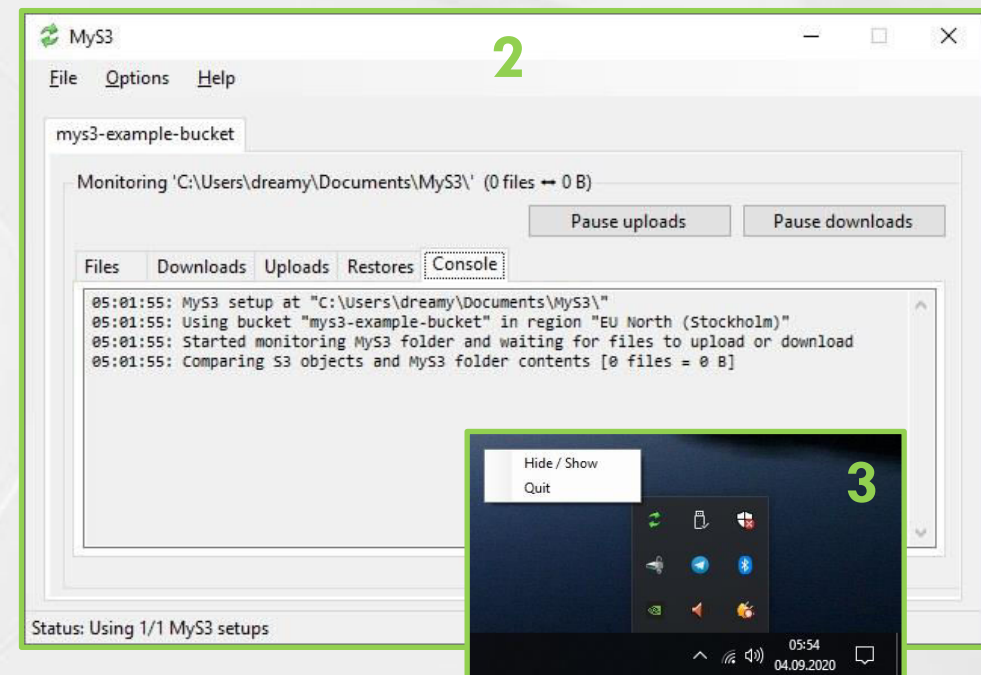
With your own S3 bucket and access credentials you can now setup MyS3 and start uploading whatever you want, without being surveilled.

How it looks when you get the console client (1) or GUI client (2-3), and you run your setup:

A terminal window titled "MyS3" showing the output of the command `./MyS3.CLI --verbose --run-tests --bucket=mys3-example-bucket --region=eu-north-1 --aws-access-key=AKIA5MC54425SKSET03V,SD1+smZuKcCYdq00hYLP ZuKcCYdq00hYLP --encryption-password="This is my password with strange chars @#&!"`. The output shows the version (1.0.0.0), test results (all passed), and the setup path (`/home/debian/Documents/MyS3/`). It also displays instructions for pausing, quitting, and monitoring the folder.

```
File Edit View Search Terminal Help
debian@debian-pc:~$ ./MyS3.CLI --verbose --run-tests --bucket=mys3-example-bucket --region=eu-north-1 --aws-access-key=AKIA5MC54425SKSET03V,SD1+smZuKcCYdq00hYLP ZuKcCYdq00hYLP --encryption-password="This is my password with strange chars @#&!"
9/4/2020 6:58:07 AM: MyS3.CLI version 1.0.0.0
9/4/2020 6:58:07 AM: MyS3 version 2.0.0.0
9/4/2020 6:58:07 AM: Running tests to check certain settings
9/4/2020 6:58:11 AM: Every test succeeded so running MyS3 is proceeding
9/4/2020 6:58:11 AM: MyS3 setup at "/home/debian/Documents/MyS3/"
9/4/2020 6:58:11 AM: Using bucket "mys3-example-bucket" in region "EU North (Stockholm)"
9/4/2020 6:58:11 AM: Started monitoring MyS3 folder and waiting for files to upload or download
9/4/2020 6:58:11 AM: Press 'p' to pause or continue MyS3's uploads and downloads
9/4/2020 6:58:11 AM: Press 'q' at any time to quit MyS3 gracefully and then wait for work to finish
9/4/2020 6:58:11 AM: .....
9/4/2020 6:58:12 AM: Comparing S3 objects and MyS3 folder contents [0 files = 0 B]
```

(A working console client setup can of course be put into a script file later for convenience.)





# Console Client

How To Use MyS3

Ove Bakken

Use of the console client is straightforward; just type in the necessary settings:

```
debian@debian-pc: ~  
File Edit View Search Terminal Help  
debian@debian-pc:~$ ./MyS3.CLI  
MyS3.CLI settings:  
--verbose  
--run-tests  
--bucket=<name of your bucket>*  
--region=<name of bucket region>*  
--aws-access-key=<your bucket user's key id>,<your bucket user's secret access key>*  
--encryption-password=<your own private file encryption password>*  
--mys3-folder=<absolute path to preferred mys3 folder>  
--shared-bucket  
--pause-downloads  
--pause-uploads  
--restore-removed-files=<year-month-day-hour>  
--restore-file-versions=<year-month-day-hour>  
  
(All fields marked * are mandatory because MyS3 will not work otherwise)  
  
Use '--run-tests' to check your settings if this is your first run!  
  
If run by multiple clients use '--shared-bucket' to enable more S3 and MyS3 comparisons  
  
Windows example:  
MyS3.CLI --verbose --bucket=myfiles --region=eu-west-1 --aws-access-key=AKIA123etc,abc123etc  
--encryption-password="my password" --mys3-folder="C:\Users\Smiley\Documents\MyS3\  
--restore-removed-files=2020-07-01-15 --restore-file-versions=2020-01-01-12'  
  
*nix example:  
MyS3.CLI --verbose --bucket=myfiles --region=eu-west-1 --aws-access-key=AKIA123etc,abc123etc  
--encryption-password="my password" --mys3-folder="/home/Smiley/Documents/MyS3/"  
--restore-removed-files=2020-07-01-15 --restore-file-versions=2020-01-01-12'  
debian@debian-pc:~$
```

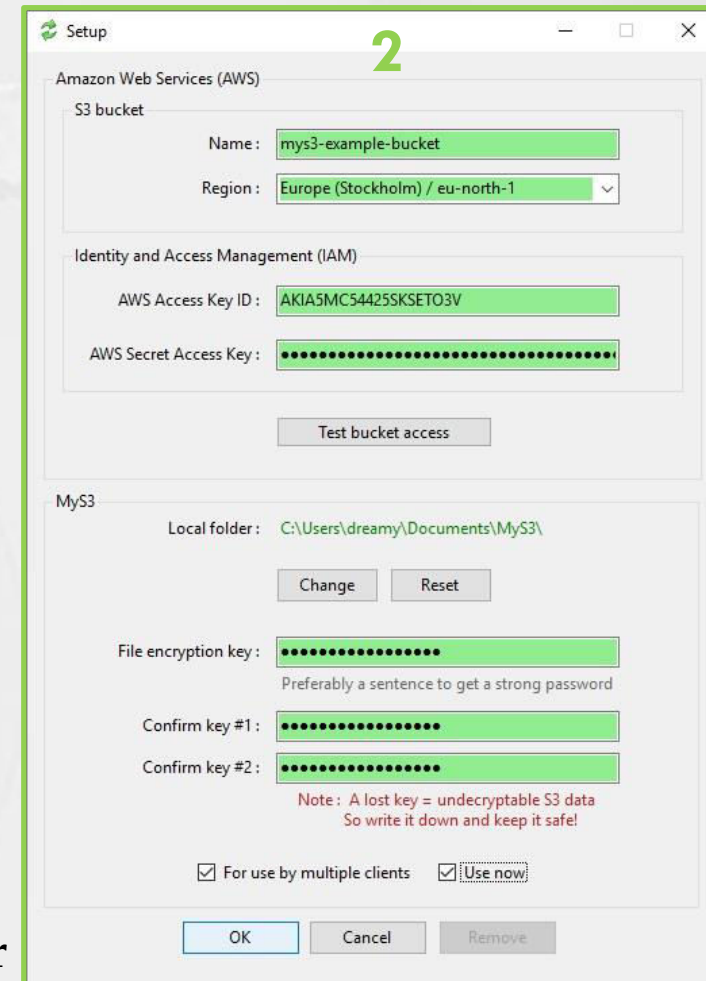
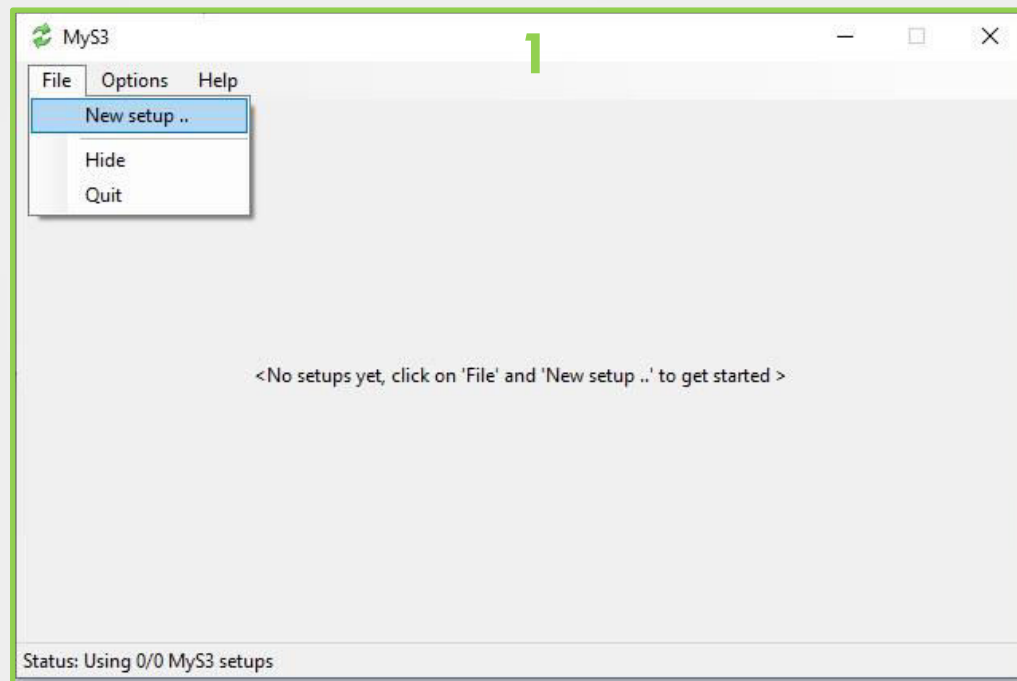
Don't be afraid to make a mistake. MyS3 will tell you if there's a problem.

# Graphical Interface Client Setup

How To Use MyS3

Ove Bakken

When you run the graphical interface client (1)  
you get a lot of help with your setup (2):



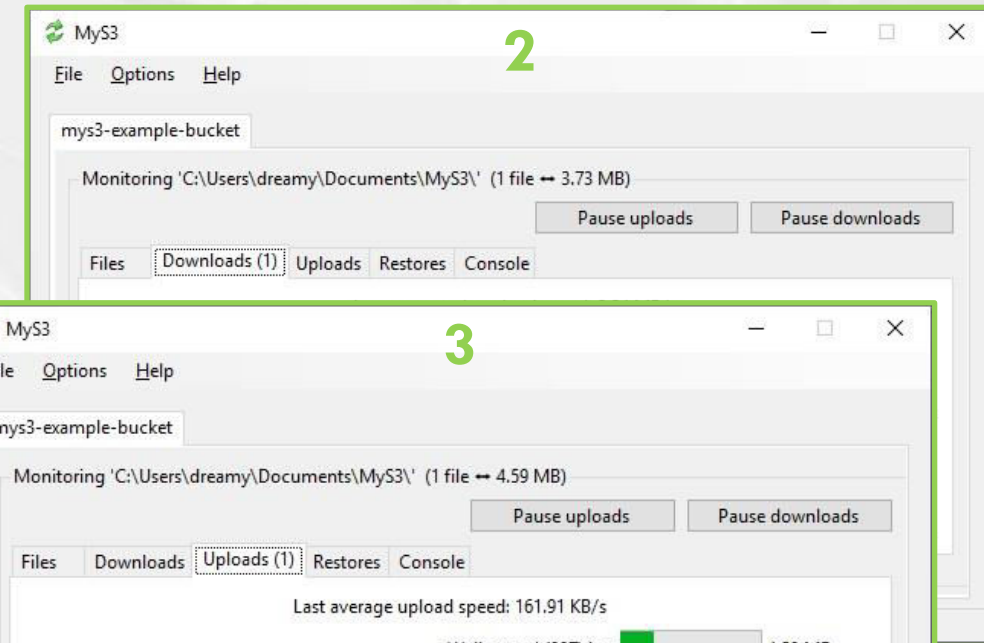
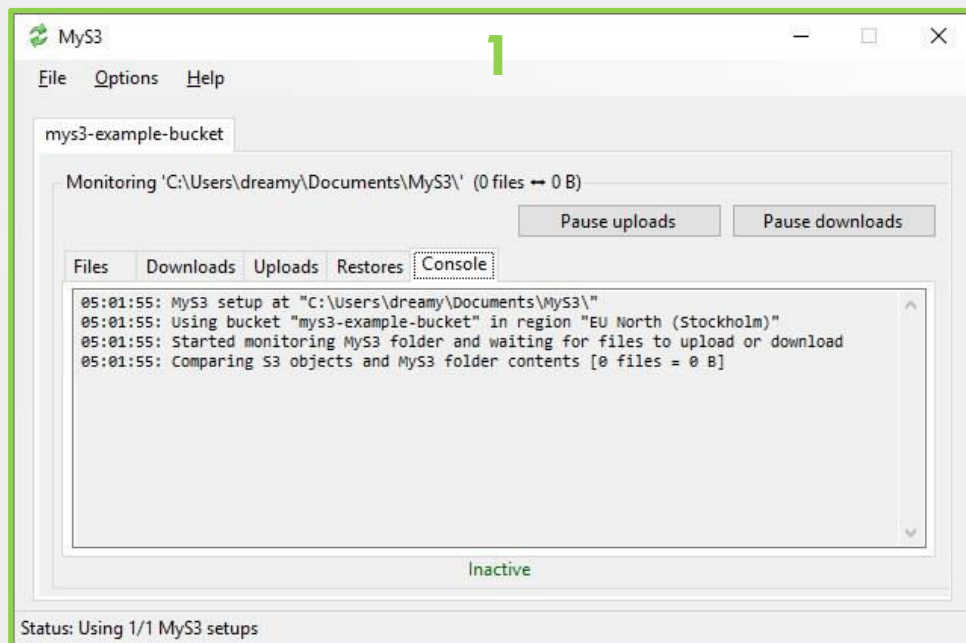
(Tip: Copy and paste in all your settings for  
greater speed and accuracy!)

# Graphical Interface Client Sync

How To Use MyS3

Ove Bakken

When MyS3 runs a setup it starts looking for new S3 bucket objects *and* it starts monitoring file activity (1). Then it does the necessary work (2-3) when something finally happens:

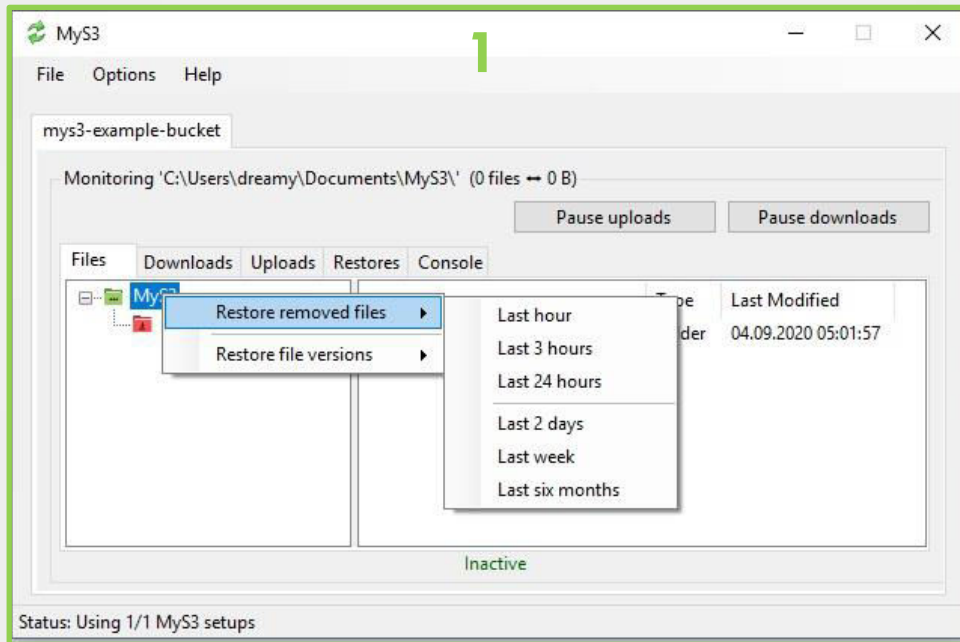


# Graphical Interface Client Restore

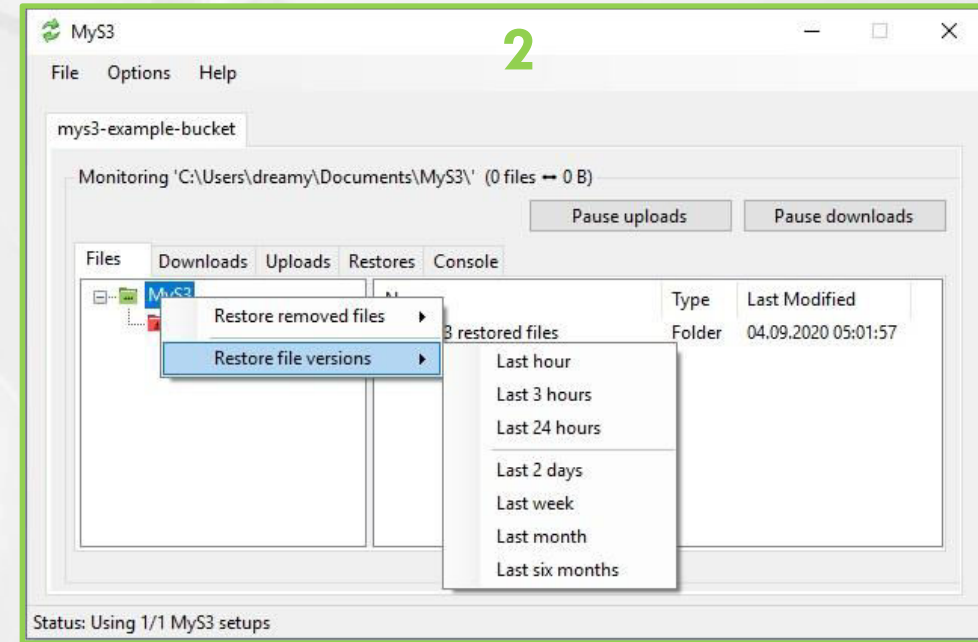
How To Use MyS3

Ove Bakken

To restore last removed files (1) or earlier file versions (2), right click on the root in the Files tab:



(Removed files gets added back to the same folder they were removed from.)



(Earlier file versions also gets added back, but root becomes the MyS3 restored files folder.)



# Missing Features

**File Sharing:** Since MyS3 encrypts all files and file paths *before* upload, sharing file links is not possible. However, you can always create another bucket and share it with somebody you trust. Because MyS3 is able to run multiple setups at the same time.

**Password Recovery:** MyS3 is not a file hosting service like Dropbox, etc. Meaning you have nowhere to turn if you lose your file encryption password. *Which is why you should write it down and keep it in a safe place!* You can also not change your encryption password whenever you feel like it, because all your files and file paths have already been encrypted with the current password and then uploaded.

To set a new file encryption password: 1) Remove your setup from your MyS3 client and quit the client. 2) Log onto the AWS Management Console and empty your entire S3 bucket. 3) Restart your MyS3 client and add back your setup with the new file encryption password, and click OK. 4) Wait for MyS3 to sync your files all over again, this is everything in your MyS3 folder.

# A Few Gotchas

MyS3 only compares last changed timestamps when deciding if a file needs to be uploaded, or an S3 object downloaded. This is a problem if you still have copies of old files (perhaps in your computer's memory), that you accidentally save. MyS3 will then overwrite your newest S3 file content with the old. If this happens you have to use MyS3's function for restoring earlier file versions. Then you check each file version to find the correct one. You rename it and put it back where it belongs.

It's also best to let MyS3 run all the time you make changes to the content in your MyS3 folder. Particularly if you are sharing your S3 bucket. You can easily hide MyS3's main window. MyS3 will still run in the background.

*\* "Client-side encryption is the cryptographic technique of encrypting data on the sender's side, before it is transmitted to a server such as a cloud storage service."*

*-[https://en.wikipedia.org/wiki/Client-side\\_encryption](https://en.wikipedia.org/wiki/Client-side_encryption)*



# Thank You

I appreciate your interest and welcome questions and feedback!

---

[post\(a\)ovebakken.no](mailto:post(a)ovebakken.no)