Sync Dropbox files to AWS S3

By launching an EC2 instance dedicated to data transfer, it will sync to my Dropbox account, downloading all the data, and then push that data to S3 for long term storage.

Dropbox Account

Estimate how much data you have on DB via user dropdown menu on DB account page.

EC2 instance

The instance that I used was Ubuntu Server on a t2.micro machine. My data was around 6 GB so I need not had to add any extra storage.

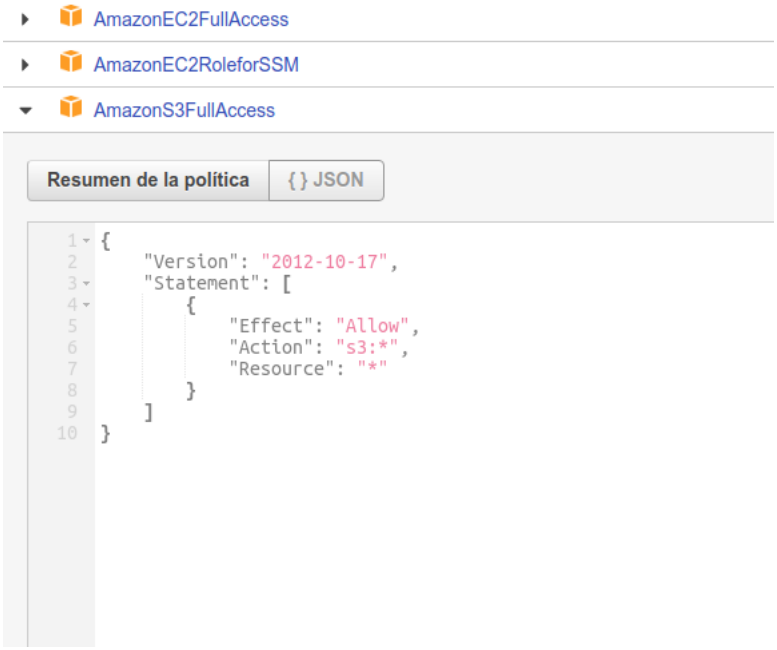
computing needs.

Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
☐	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes

Note - It's important that this instance is assigned a role that has read/write privileges to the S3 bucket where you will store the data. The quick and dirty method would be to give the role AmazonS3FullAccess. However, the more secure and preferred way is to modify the *Resource* field within the JSON file to contain only the target bucket.



S3 Bucket

You can use an existing S3 Bucket or create a new one according to your use. It makes sense to keep the bucket location as same in the EC2 region.

The Setup

After launching all the resources we will need to ssh into the EC2 instance.

```
ssh -i *YOUR-PEM-FILE-LOCATION* ubuntu@**YOUR-EC2-ADDRESS*
```

The first step is to modify the *inotify.max_users.* These entities are used to monitor individual files or directories within the Linux OS. However, in our case, we are going to move so many files that the default number of max users will cause a system error and halt the DropBox sync.

To avoid that, increase the setting to at least 100,000 and reboot the instance.

```
echo fs.inotify.max_user_watches=100000 | sudo tee -
a /etc/sysctl.conf; sudo sy$

sudo reboot
```

The reboot will terminate the ssh session, so you will need to connect again.

Syncing Dropbox

Now that the instance is ready we can sync the Dropbox account to it. Download/extract the official Linux64 tar file and run the preconfigured installation script (*dropboxd*).

```
cd ~ && wget -
O - "https://www.dropbox.com/download?plat=lnx.x86_64" | tar xzf -

~/.dropbox-dist/dropboxd
```

The dropboxd will ask you to verify that you are a real person, you can do that by signing via the link into the browser and confirm.

Syncing to S3

After our entire DropBox drive has been downloaded to the instance the final step is to transfer the data from the EC2 instance to the S3 bucket.

To start we must install the AWS command line.

```
sudo apt -y install awscli
```

And to push the data –

```
aws s3 sync ~/Dropbox s3://**YOUR-BUCKET-LOCATION**/Dropbox
```

Finished

That’s it. Depending on how much data you're moving, the process could take between a few minutes and several days. It’s best to monitor the S3 bucket size (bytes) metrics to see how what percentage of the data has been transferred.