

# AWS S3 연동하기

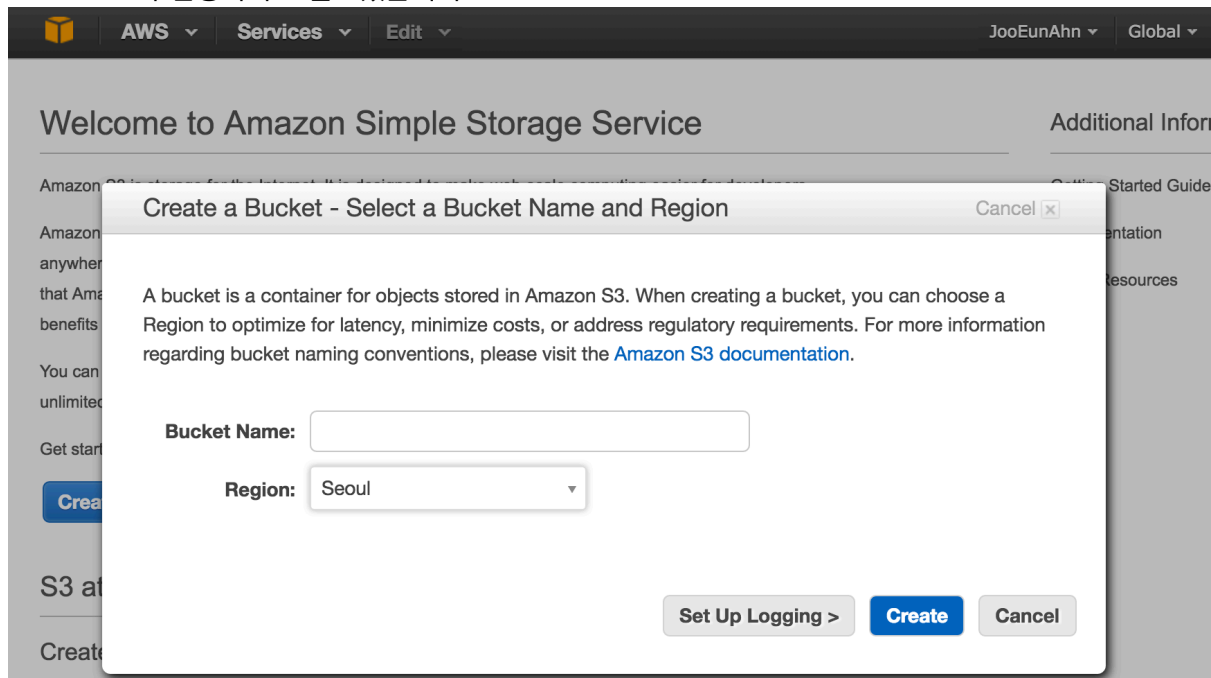
S3를 다루기 앞서... 스누프라이스의 웹을 보다보면 Static 파일들이 많고, 장기적으로 보자면 이 파일들을 관리해야 할 것이라고 생각되어서! 또저도 연동하는 것을 문서화를 할 겸해서 자료 보내드립니다. S3의 자세한 내용은 저보다 구글이 더 잘알고있으니.. 구글에 물어보시면 될 듯합니다 ㅎㅎㅎㅎ

먼저 스누프라이스의 서버안에 STATIC설정과 MEDIA설정이 되어있다고 가정하고 시작하겠습니다.

## 1. S3버킷만들기

1.1 먼저 AWS 콘솔에 접속해 bucket을 만들어주세요.

서울에 EC2를 두고 계시다면 S3도 서울에 두시는 것이 좋습니다. 서로 다른지역에 존재할경우 Time loss가 엄청나다고 알고있습니다.



1.2 <https://console.aws.amazon.com/iam/home?#users> 에 접속에 유저를 만들어주세요. 5개의 칸이 한꺼번에 뜨는데 한개만 만드셔도 됩니다. 또 Generate an access key for each user 를 선택해주세요.

**Enter User Names:**

1.
2.
3.
4.
5.

Maximum 64 characters each

☒ **Generate an access key for each user**

Users need access keys to make secure REST or Query protocol requests to AWS service APIs.

*For users who need access to the AWS Management Console, create a password in the Users panel after completing the wizard.*

1.3 Credential은 다운받으셔도 좋고, 복사하셔도 좋습니다. 하지만 없어지지 않을까요? 다운받는것이 좋을 것 같아요.

☑ Your 1 User(s) have been created successfully.

This is the last time these User security credentials will be available for download.

You can manage and recreate these credentials any time.

▶ [Show User Security Credentials](#)

[Close](#)

[Download Credentials](#)

1.4 생성하고 유저카테고리에 새로 생성한 user의 디테일에 들어가 User ARN을 복사해두세요. 세팅에 쓰입니다.

[IAM](#) > [Users](#) > **cofin**



### ▼ Summary

<b>User ARN:</b>	arn:a
<b>Has Password:</b>	No
<b>Groups (for this user):</b>	0
<b>Path:</b>	/
<b>Creation Time:</b>	2016

1.5 이제 S3 콘솔로 돌아가 Properties의 Permissions에서 bucket policy를 클릭해 아래의 코드를

넣고 수정해주세요.

BUCKET-NAME과 USER-ARN만 수정해주시면 됩니다.

위의 코드는 콘텐츠를 Public하게 readable하게 만들어주는 설정이구요, 밑에는 어떤 유저가 S3에서 작업할지를 설정하는 코드입니다.

```
{
  "Statement": [
    {
      "Sid": "PublicReadForGetBucketObjects",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": ["s3:GetObject"],
      "Resource": ["arn:aws:s3:::BUCKET-NAME/*"]
    },
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::BUCKET-NAME",
        "arn:aws:s3:::BUCKET-NAME/*"
      ],
      "Principal": {
        "AWS": [
          "USER-ARN"
        ]
      }
    }
  ]
}
```

\*만약에 누군가에게 추가권한을 주고싶다면 아래코드처럼 permission에 저장해주시면 됩니다.  
아래의 코드는 특정인에게 버킷에있는 콘텐츠를 복제할 수 있는 권한을 주는 것입니다.

```
{
  "Action": "s3:ListBucket",
  "Effect": "Allow",
  "Resource": "arn:aws:s3:::BUCKET-NAME",
  "Principal": {
    "AWS": [
      "USER-ARN"
    ]
  }
}
```

## 2. S3를 사용하기 위한 장고 설정

스누프라이스의 웹서버가 {% load staticfiles %}로 static url을 서빙한다는 전제 하에 진행할게요.

시작하기에 앞서 django-storages패키지와 boto 패키지를 설치해주세요.(pip3 install django-storages boto)

또 settings.py에 INSTALLED\_APPS란에 'storages'를 추가 해주세요.

그리고 이건 옵션인데요. user에게 전달할 콘텐츠의 caches에 얼마나 담아놓을지 설정할 수 있습니다.

```
        AWS_HEADERS = {
# see http://developer.yahoo.com/performance/rules.html#expires
    'Expires': 'Thu, 31 Dec 2099 20:00:00 GMT',
    'Cache-Control': 'max-age=94608000',
}
```

이제 본격적으로 S3에 연동해 볼텐데요. 서울 리전을 사용하는데 있어 조금 이슈가 있어 말씀드리려합니다. 기존 리전들의 S3는 Signature Version 4와 Version2를 모두 지원하는 반면 최근에 생긴 서울, 프랑크푸르트같은 리전들은 Version4만 지원합니다. django-sotrages패키지의 경우 기본으로 Signature Version2를 이용하므로 문서대로 사용하게되면 모두 http 400를 반환하게 됩니다. 그래서 Signature Version 4를 사용하도록 바꾸어 주어야 합니다. 아래 코드는 모두 Signature Version 4를 이용한 코드입니다.

아래의 코드를 배포용 세팅에 넣어주세요. ACCESS KEY와 씨크릿 키는 credential에 들어있습니다.

```
## AWS S3
DEFAULT_FILE_STORAGE = 'storages.backends.s3boto.S3BotoStorage'
AWS_ACCESS_KEY_ID = '*****' # access key
AWS_SECRET_ACCESS_KEY = '*****' # secret access key
AWS_REGION = 'ap-northeast-2' ## Seoul Region
AWS_STORAGE_BUCKET_NAME = 'bucket_name' # AWS S3 버킷 이름
AWS_QUERYSTRING_AUTH = False
AWS_S3_HOST = 's3.%s.amazonaws.com' % AWS_REGION
AWS_S3_SECURE_URLS = False ## https
```

또한 개발중일경우 S3를 이용하는 것보다 django에서 static을 serve하는 것이 옳은 방향이므로 setting 파일중 동시에 읽어드리는 common.py에 아래의 코드처럼 넣어주세요

```
if DEBUG:
    STATICFILES_DIRS = [
        os.path.join(BASE_DIR, 'cofin', 'static'),
    ]

    STATIC_URL = '/static/'
else:
    STATICFILES_STORAGE = 'storages.backends.s3boto.S3BotoStorage'
```

```
AWS_S3_CUSTOM_DOMAIN = '%s.s3.amazonaws.com' %  
AWS_STORAGE_BUCKET_NAME  
STATIC_URL = "http://%s/" % AWS_S3_CUSTOM_DOMAIN
```

### 3. ubuntu

AWS EC2콘솔에 접속해 파이썬 패키지를 설치해주시고 collectstatic을 해주시면!

```
sudo python3 manage.py collectstatic
```

```
You have requested to collect static files at the destination  
location as specified in your settings.  
  
This will overwrite existing files!  
Are you sure you want to do this?  
  
Type 'yes' to continue, or 'no' to cancel: yes  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/scrolltopcontrol/scrolltopcontrol.js'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/contact-form/contact.js'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/isotope/isotope.pkgd.min.js'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/isotope/imagesloaded.pkgd.min.js'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/HELP-US-OUT.txt'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_larger.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_variables.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_bordered-pulled.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_core.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_path.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_fixed-width.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_icons.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_rotated-flipped.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_animated.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_stacked.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/font-awesome.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_list.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/scss/_mixins.scss'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/css/font-awesome.min.css'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/css/font-awesome.css'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/fonts/fontawesome-webfont.woff2'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/fonts/fontawesome-webfont.svg'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/fonts/fontawesome-webfont.ttf'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/fonts/fontawesome-webfont.eot'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/fonts/FontAwesome.otf'  
Copying '/home/ubuntu/cafeory/cafeory/cafeory/static/assets/plugins/font-awesome/fonts/fontawesome-webfont.woff'
```

이런식으로 S3로 static파일들을 옮기게됩니다.

그후 웹서버를 restart해주시면!

```
...  == $0
```

이런식으로 static 파일을 s3에서 서빙하게됩니다.

All Buckets / coffin

	Name	Storage Class	Size	Last Modified
<input type="checkbox"/>	admin	--	--	--
<input type="checkbox"/>	assets	--	--	--
<input type="checkbox"/>	css	--	--	--
<input type="checkbox"/>	facebook	--	--	--
<input type="checkbox"/>	images	--	--	--
<input type="checkbox"/>	img	--	--	--
<input type="checkbox"/>	js	--	--	--
<input type="checkbox"/>	owl-carousel	--	--	--
<input type="checkbox"/>	star-ratings	--	--	--

콘솔에서도 이미지를 꺼내보고 정책을 정할 수 있습니다.

이상으로 S3연동은 끝났습니다. 다시 한번 스누프라이스가 한양대까지 오는 날까지! 멀리서 응원하겠습니다. 긴 글 읽어주셔서 감사합니다.