

Ask Django

AWS Elastic Beanstalk 시작하기

AWS Elastic Beanstalk #doc #faq

PaaS가 아니라, 자동화된 IaaS 관리 서비스

- AWS CloudFormation을 통한 배포 서비스
- 용량 프로비저닝, 로드 밸런싱, Auto Scaling, 애플리케이션 배포 등의 세부 정보를 자동으로 처리하여 사용자의 애플리케이션 버전을 실행할 수 있는 환경
 - EC2, RDS, Elastic Load Balancing, Auto Scaling, S3, SNS 등을 조합
- **Python**, Java, PHP, Node.js, Ruby, .NET, Go, Docker 지원

파이썬 설정 목록 (2017년 4월 현재) #doc

Configuration 이름	파이썬 버전	AMI	Proxy Server
Python 3.4 version 2.3.3	Python 3.4.3	2016.09.0	Apache 2.4.18 with mod_wsgi 3.5
Python 2.7 version 2.3.3	Python 2.7.10	동일	동일
Python 2.6 version 2.3.3	Python 2.6.9	동일	동일

Python Platform History

주요 용어

- Application : 다수 Environment을 담는 컨테이너
- Environment : 장고 애플리케이션 적용 환경
- Application Versions
 - 장고 애플리케이션 배포 히스토리
 - 언제라도 지정 버전으로 Environment에 배포 가능 (롤백)

AWS EB 배포 준비

EB에 파이썬 플랫폼 배포 순서 #doc #options

- 장고 소스코드 체크
 - .gitignore 파일 체크
 - requirements.txt 파일 체크 : 지정해두면 자동 설치
 - AWS EB용 settings 체크
- EB 설정
 - .ebextensions/options.config 파일 생성/체크
 - 파일명은 Don't care.
 - .ebextensions 디렉토리내 모든 config를 수행
- EB 배포
 - 방법1) 프로젝트를 zip 압축해서, EB 콘솔에서 배포
 - 방법2) EB CLI통해서 배포

체크 포인트

본 배포에서는

- static 파일 서빙은 Apache 웹서버를 통해 합니다.
 - 실 서비스에서는 AWS S3를 쓰도록 합니다.
- DB 를 SQLite3 를 씁니다.
 - 실 서비스에서는 AWS RDS를 쓰도록 합니다.

장고 소스코드 체크

requirements.txt 경로

django

장고 소스코드 체크

프로젝트/**settings.py** 경로

```
ALLOWED_HOSTS = [ '*.elasticbeanstalk.com' ]
```

```
STATIC_URL = '/static/'
```

```
STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')
```

.ebextensions/options.config 경로

파일명은 다른 이름이어도 상관없어요.

- container_commands : 배포 직후에 수행할 쉘 명령 목록
- option_settings
 - aws:elasticbeanstalk:application:environment:
 - 환경변수 목록
 - aws:elasticbeanstalk:container:python:
 - 파이썬 설정
 - aws:elasticbeanstalk:container:python:staticfiles:
 - apache웹서버 static 설정

Example

```
container_commands:
  01_migrate:
    command: "python3 manage.py migrate"
  02_collectstatic:
    command: "python3 manage.py collectstatic --noinput"
  03_chmod_sqlite:
    command: "chmod 666 db.sqlite3"

option_settings:
  aws:elasticbeanstalk:application:environment:
    DJANGO_SETTINGS_MODULE: 프로젝트명.settings
  aws:elasticbeanstalk:container:python:
    WSGIPath: 프로젝트명/wsgi.py
  aws:elasticbeanstalk:container:python:staticfiles:
    "/static/": "staticfiles/"
```

주의

yaml 포맷을 맞추지 않으시면, **eb create** 명령 시에 다음과 오류가 발생합니다. 콜론에 유의하세요.

ERROR: Each option setting in configuration file .ebextensions/options.config in application version app-57a7-170414_105422 must be a map. Update each option setting in the configuration file.

잘못쓴 예)

option_settings:

aws:elasticbeanstalk:container:python:

WSGI 프로젝트명/wsgi.py

웹 콘솔을 통한 배포

웹 콘솔을 통한 배포 순서

- 프로젝트 디렉토리 zip 압축
 - 주의: 프로젝트 최상위 디렉토리 대상이 아니라, 프로젝트 파일들이 최상위에 오도록 압축
- Create New Application
- New Environment : Web Server Environment [#doc](#)
- Environment Type
 - Predefined configuration: Python (디폴트 : Python 3.4 on 64bit Amazon Linux 2016.09 v2.3.3)
 - Environment type: Single Instance
- Application Version
 - "Upload your own" 을 통해 압축한 프로젝트 zip 업로드

- Environment Information
 - Environment name : 프로젝트명-dev
 - Environment URL : 원하시는sub도메인
- Additional Resources 부분은 Skip
 - Create an RDS DB Instance with this environment
 - Create this environment inside a VPC
- Configuration Details
 - Instance type: t2.micro (Free-tier 지키기)
 - EC2 key pair : SSH 접속을 하기 위해 필요
- Environment Tags
- Permissions [#doc](#)
- Review

Tip: 숨김파일 보기

- 맥 파인더

```
셸> defaults write com.apple.finder AppleShowAllFiles TRUE  
셸> killall Finder
```

- 윈도우 탐색기

- 윈8 이전 : 도구 -> 폴더 옵션 "숨김 파일, 폴더 및 드라이브 표시" 체크
- 윈8 ~ : 보기 -> "숨긴 항목" 체크

배포 완료

Services ▾ Resource Groups ▾ ↗

Elastic Beanstalk [Create New Environment](#)

All Applications > helloyuji > helloyuji-env (Environment ID: e-hicwmzpcda, URL: [helloyuji-env.ap-northeast-2.elasticbeanstalk.com](#)) Actions ▾

Dashboard
Configuration
Logs
Health
Monitoring
Alarms
Managed Updates
Events
Tags

Overview [Refresh](#)

Health **Ok** Causes

Running Version
helloyuji
[Upload and Deploy](#)

Configuration
64bit Amazon Linux 2016.09
v2.3.3 running Python 3.4
[Change](#)

Recent Events [Show All](#)

Time	Type	Details
2017-04-14 12:04:05 UTC+0900	INFO	Environment health has transitioned from Info to Ok. Application update completed 30 seconds ago and took 81 seconds.
2017-04-14 12:03:05 UTC+0900	INFO	Environment health has transitioned from Degraded to Info. Application update in progress. 1 out of 1 instance completed (running for 81 seconds).
2017-04-14 12:02:41 UTC+0900	INFO	Environment update completed successfully.
2017-04-14 12:02:41 UTC+0900	INFO	New application version was deployed to running EC2 instances.
2017-04-14 12:01:40 UTC+0900	INFO	Deploying new version to instance(s).

배포시에 **superuser** 계정 만들기

이는 간단한 배포를 위한 임시적인 방법입니다. 실서비스에서는 **X**

```
# create_superuser.py
import django
django.setup()

from django.contrib.auth import get_user_model

def main():
    username = 'admin'
    email = 'admin@example.com'
    password = 'askdjangovod'
    User = get_user_model()
    if not User.objects.filter(username=username).exists():
        User.objects.create_superuser(username, email, password) # 로그인 후에 비밀번호를 필히 변경해주세요.
        print('Sucessfully create superuser')

if __name__ == '__main__':
    main()
```

.ebextensions/options.config 에 추가

container_commands:

04_create_superuser:

command: "python3 create_superuser.py"

- zip 압축하고, 재 업로드/배포
- 주의사항 : VM 로컬 sqlite3 DB이기에 배포 시마다 DB 초기화됩니다.
 - 별도 DB서버 (RDS) 를 쓰는 것이 좋습니다.

EB 애플리케이션 삭제하기

EB 웹 콘솔에서 아래 순서대로 삭제

- Application Versions
 - Delete versions from Amazon S3 체크 확인
 - 배포 파일은 S3에 저장되기 때문입니다.
- Environment
- Application

Reference

- [Deploying a Django Application to Elastic Beanstalk](#)
- [Amazon Web Services Korea] [Djangogirls를 위한 Django App을 AWS Elastic Beanstalk에 배포하기](#)

*Life is short,
use Python3/Django.*