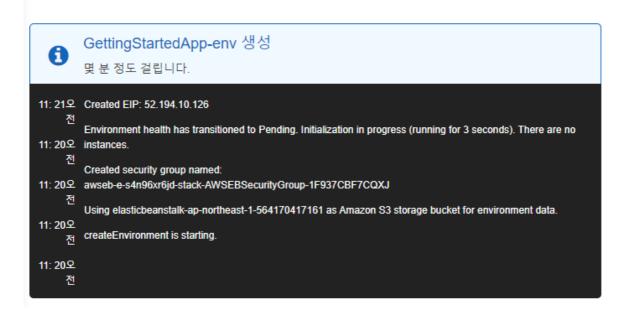
## Elastic Beanstalk 사용

#### 1. 예제 애플리케이션에 대해 생성

(platform은 도커)

- 1. <a href="https://console.aws.amazon.com/elasticbeanstalk/home#/gettingStarted?applicationName=getting-started-app 접속">https://console.aws.amazon.com/elasticbeanstalk/home#/gettingStarted?applicationName=getting-started-app 접속</a>
- 2. 플랫폼을 도커로 설정
- 3. 샘플 애플리케이션 실행을 위해서 5분정도 소요

모든 애플리케이션 > getting-started-app > GettingStartedApp-env (환경 ID: e-s4n96xr6jd)

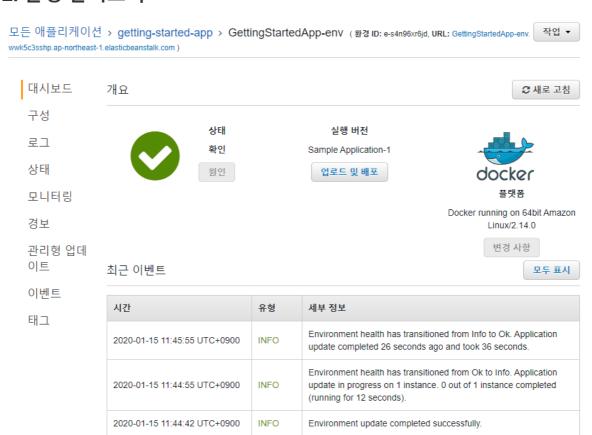


# 예제 애플리케이션을 생성할 때 Elastic Beanstalk는 다음 AWS 리소스를 생성합니다.

- EC2 인스턴스 선택한 플랫폼에서 웹 앱을 실행하도록 구성된 Amazon EC2 가상 머신입니다. 특정 언어 버전, 프레임워크, 웹 컨테이너 또는 조합을 지원하도록 각 플랫폼마다 실행하는 소프트 웨어, 구성 파일 및 스크립트 세트가 다릅니다. 대부분의 플랫폼에서는 웹 앱 앞의 웹 트래픽을 처 리하고, 웹 앱으로 요청을 전달하고, 정적 자산을 제공하고, 액세스 및 오류 로그를 생성하는 역방 향 프록시로 Apache 또는 nginx를 사용합니다.
- **인스턴스 보안 그룹** 포트 80에서 수신 트래픽을 허용하도록 구성된 Amazon EC2 보안 그룹입니다. 이 리소스를 통해 로드 밸런서의 HTTP 트래픽이 웹 앱을 실행하는 EC2 인스턴스에 도달할 수 있습니다. 기본적으로 다른 포트에서는 트래픽이 허용되지 않습니다.
- Amazon S3 버킷 Elastic Beanstalk 사용 시 생성된 소스 코드, 로그 및 기타 결과물의 스토리지 위치입니다.
- Amazon CloudWatch 경보 환경의 인스턴스에 대한 로드를 모니터링하는 두 개의 CloudWatch 경보로, 로드가 너무 높거나 너무 낮은 경우 트리거됩니다. 경보가 트리거되면 이에 대한 응답으로 Auto Scaling 그룹이 확장 또는 축소됩니다.

- AWS CloudFormation 스택 Elastic Beanstalk에서는 AWS CloudFormation을 사용하여 사용 자 환경의 리소스를 시작하고 구성 변경 사항을 전파합니다. 리소스는 AWS CloudFormation 콘솔에서 볼 수 있는 템플릿에서 정의됩니다.
- **도메인 이름** *subdomain. region.elasticbeanstalk.com* 형식으로 웹 앱으로 라우팅되는 도메인 이름입니다.

#### 2. 환경 살펴보기



• **구성** – 애플리케이션을 호스팅하는 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스 등이 환경에 대해 프로비저닝된 리소스가 표시됩니다. 이 페이지에서 일부 프로비저닝된 리소스를 구성할 수 있습니다.

app.

INFO

New application version was deployed to running EC2 instances.

Docker container c864ece63c26 is running aws\_beanstalk/current-

- **상태** 애플리케이션을 실행하는 Amazon EC2 인스턴스에 대한 상태와 세부 상태 정보가 표시됩니다.
- **모니터링** 평균 지연 시간 및 CPU 사용률 등 환경에 대한 통계가 표시됩니다. 이 페이지를 사용하여 모니터링 중인 측정치에 대한 경보를 만들 수 있습니다.
- 이벤트 이 환경에서 사용하는 Elastic Beanstalk 서비스 및 리소스가 있는 다른 서비스의 정보 또는 오류 메시지를 표시합니다.
- 태그 환경 태그를 표시하고 관리할 수 있습니다. 태그는 환경에 적용되는 키-값 쌍입니다.

## 3. 새 버전의 애플리케이션 배포

- 단일 컨테이너 Docker docker-singlecontainer-v1.zip
- 멀티컨테이너 Docker docker-multicontainer-v2.zip

2020-01-15 11:44:42 UTC+0900

2020-01-15 11:44:33 UTC+0900

• 미리 구성된 Docker(Glassfish) - docker-glassfish-v1.zip

- **Go** <u>go-v1.zip</u>
- Java SE <u>java-se-jetty-gradle-v3.zip</u>
- Tomcat(기본값) java-tomcat-v3.zip
- Tomcat 7 java7-tomcat7.zip
- .NET dotnet-asp-v1.zip
- Node.js nodejs-v1.zip
- PHP php-v1.zip
- Python python-v1.zip
- Ruby(Passenger Standalone) <u>ruby-passenger-v3.zip</u>
- Ruby(Puma) ruby-puma-v3.zip
- 1. 환경 플랫폼과 일치하는 샘플 애플리케이션을 다운로드하십시오. 다음 애플리케이션 중 하나를 사용합니다.
- 2. 콘솔에서 GettingStartedApp-env를 클릭하고 들어갑니다.
- 3. 개요 섹션에서 업로드 및 배포를 선택합니다.
- 4. 파일 선택을 선택하고 다운로드한 샘플을 애플리케이션에 소스 번들로 업로드하십시오.
- 5. 배포를 선택

개요 ♂ 새로 고침



상태 정보

원인

실행 버전

Sample Application-1

업로드 및 배포



Docker running on 64bit Amazon Linux/2.14.0

변경 사항

Elastic Beanstalk가 Amazon EC2 인스턴스에 파일을 배포하는 동안, 환경의 대시보드에서 배치 상태를 볼 수 있습니다. 애플리케이션 버전이 업데이트되는 동안 Environment Health(환경 상태)는 회색입니다. 배포가 완료되면 Elastic Beanstalk에서 애플리케이션 상태 확인을 수행합니다. 애플리케이션이상태 확인에 응답하면 상태가 정상으로 간주되고 상태가 녹색으로 돌아갑니다. 환경 대시보드는 새로운 실행 버전—버전 레이블로 제공한 이름을 표시합니다.

### 4. 환경 구성

애플리케이션에 더 적합하도록 환경을 구성할 수 있습니다. 예를 들어 컴퓨팅 집약적인 애플리케이션이 있는 경우 애플리케이션을 실행 중인 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스의 유형 을 변경할 수 있습니다. 구성 변경을 적용하기 위해 Elastic Beanstalk는 환경 업데이트를 수행합니다.

일부 구성은 변경이 간단하고 빠르게 처리되지만 일부 변경의 경우 AWS 리소스를 삭제한 후 다시 만들어야 하며, 여기에는 몇 분 정도 걸릴 수 있습니다. 구성 설정을 변경하면 Elastic Beanstalk는 잠재적 인애플리케이션 중단 시간에 대해 경고합니다.

- 1. 관리 페이지로 이동
- 2. **구성**을 선택
- 3. 용량 구성 범주에서 수정을 선택



4. Auto Scaling 그룹 섹션에서 환경 유형을 로드 밸런싱 수행으로 변경합니다.

#### Auto Scaling 그룹

사용되는 인스턴스 수를 최적화할 수 있도록 환경의 컴퓨팅 파워와 Auto Scaling 설정을 구성합니다.

환경 유형 로드 밸런싱 수행 ▼

- 5. 인서턴스 행에서 최대를 4로 변경하고 최소는 2로 변경후 저장한다.
- 6. 용량 증가를 확인하려면 탐색창에서 상태를 클릭한다
- 7. 확장된 상태 개오 페이지를 보면 총 인스턴스 수가 2임을 알 수 있다.



- 8. 애플리케이션 버전을 삭제 하려면 getting-started-app에 접속후 애플리케이션 버전을 선택
- 9. 삭제할 모든 애플리케이션 버전을 선택한 후 삭제를 클릭하면 삭제

#### 모든 애플리케이션 > getting-started-app



10. 환경을 종료하려면 GettingStartedApp-env 에 들어가서 장업의 환경종료를 선택



11. 애플리케이션을 삭제하려면 Elastic Beanstalk에 접속해서 우측상단의 **작업 - 애플리케이션 삭제** 를 클릭 **3**015