

DevOps CI/CD 파이프라인 구축

– LAST EMD



Index

1. 팀원소개

- DevOps CI/CD
- 모니터링
- IaC

2. 프로젝트 개요

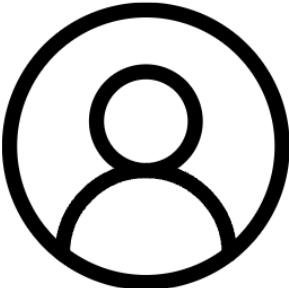
3. 프로젝트 진행 내용

- CI/CD 파이프라인 생성
- SCA, SAST, DAST 테스트 및 통합
- 모니터링 기능 및 IaC 구현

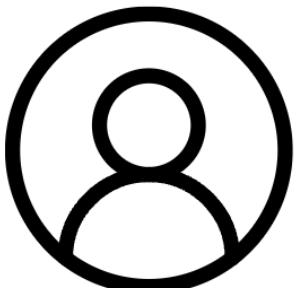
4. 프로젝트 결과

- 클라우드 및 DevOps 이해
- 파이프라인 구축 및 IaC
- DevOps 보안 요소
- AWS 비용
- 향후 발전 방향

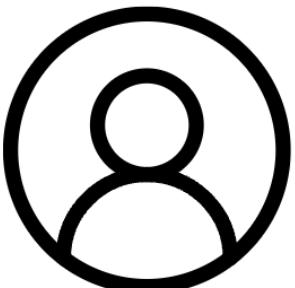
팀원 소개



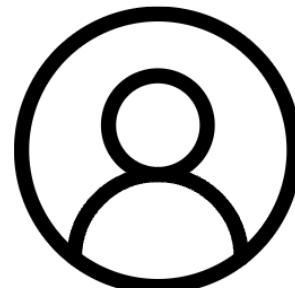
Project Leader(PL)
남지우



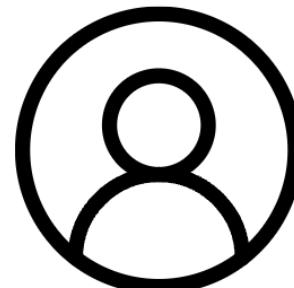
Project Manager(PM)
박혜수



Project Agent(PA)
곽민경



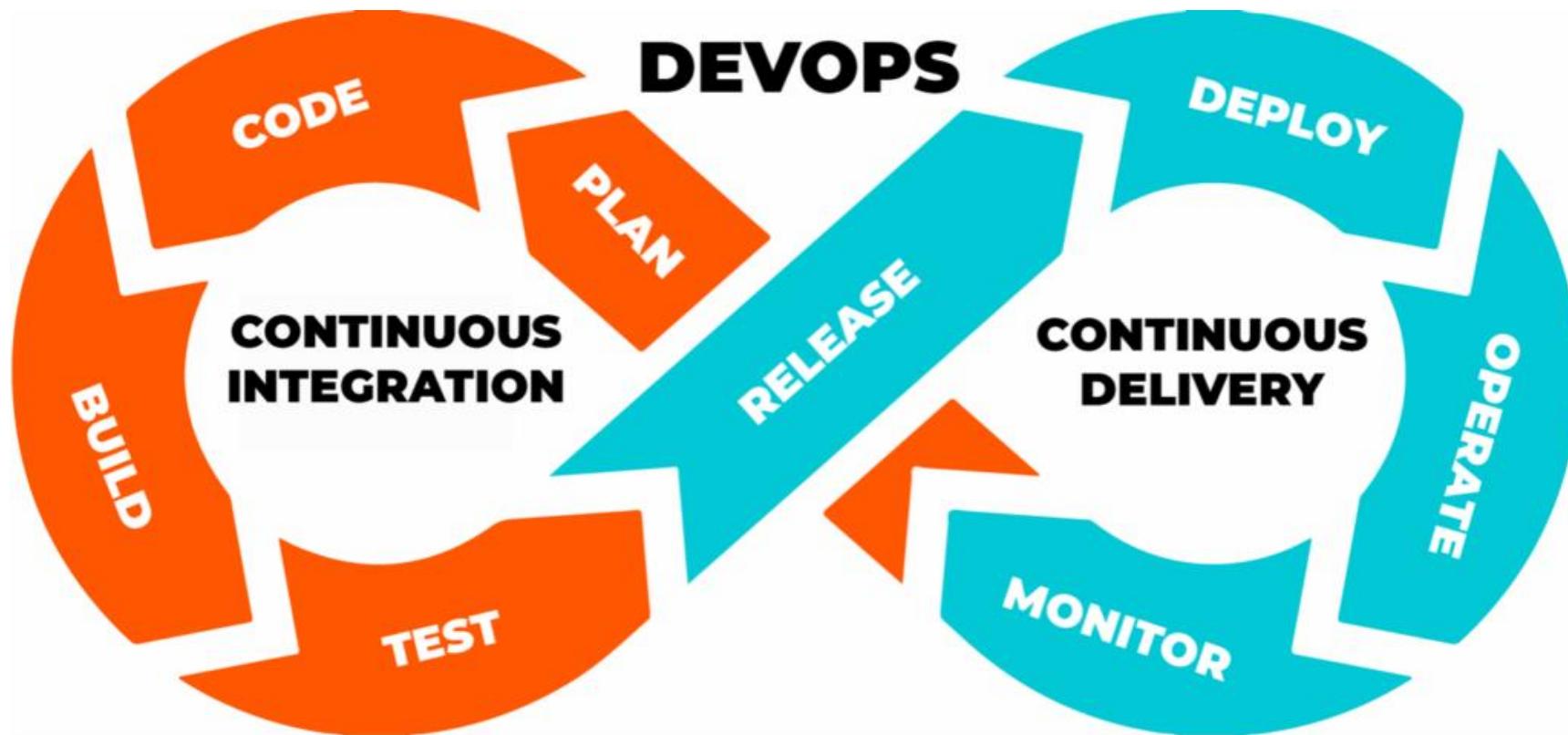
Project Agent(PA)
윤주원



Project Agent(PA)
천재권

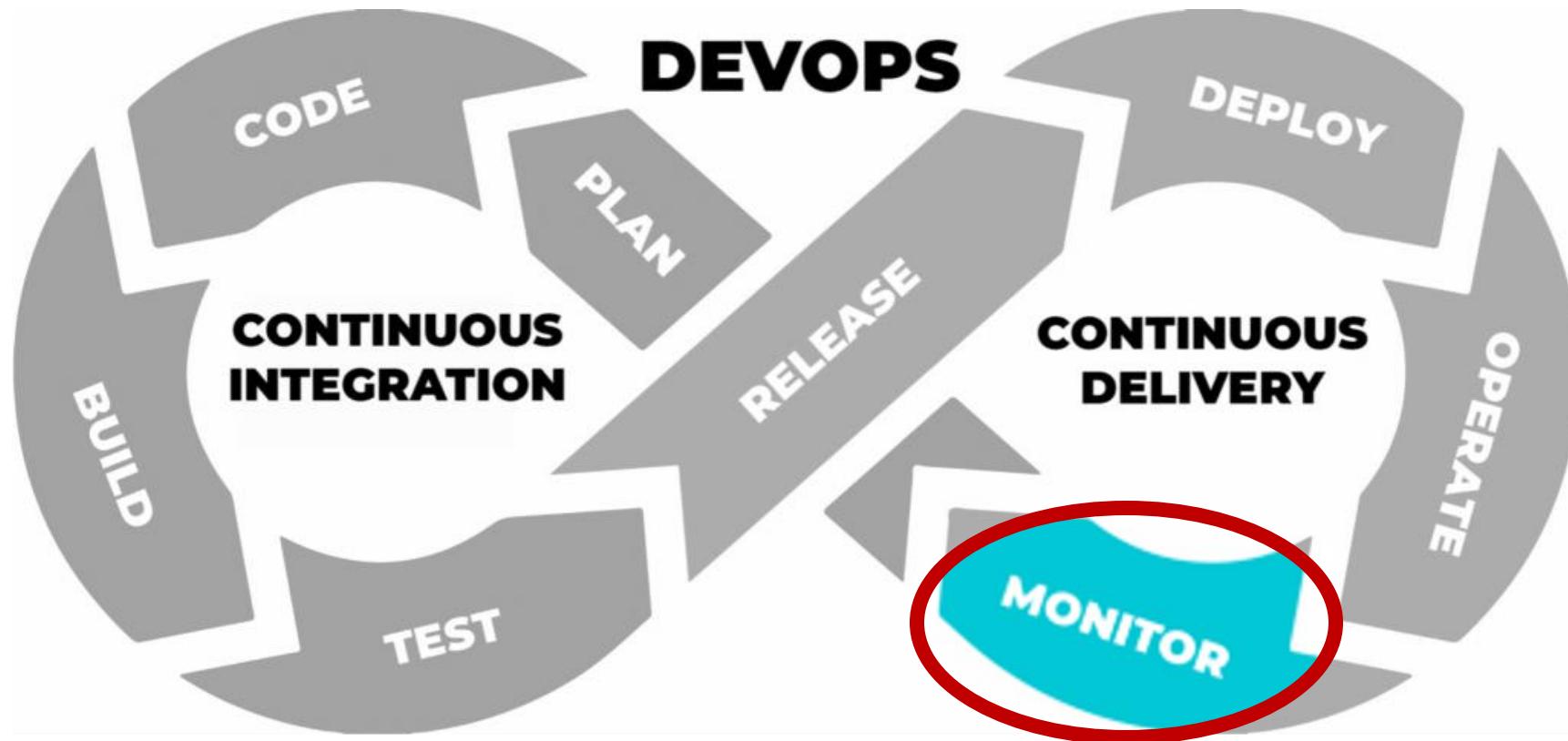
프로젝트 개요 – DevOps CI/CD

DevOps CI/CD 파이프라인 구축



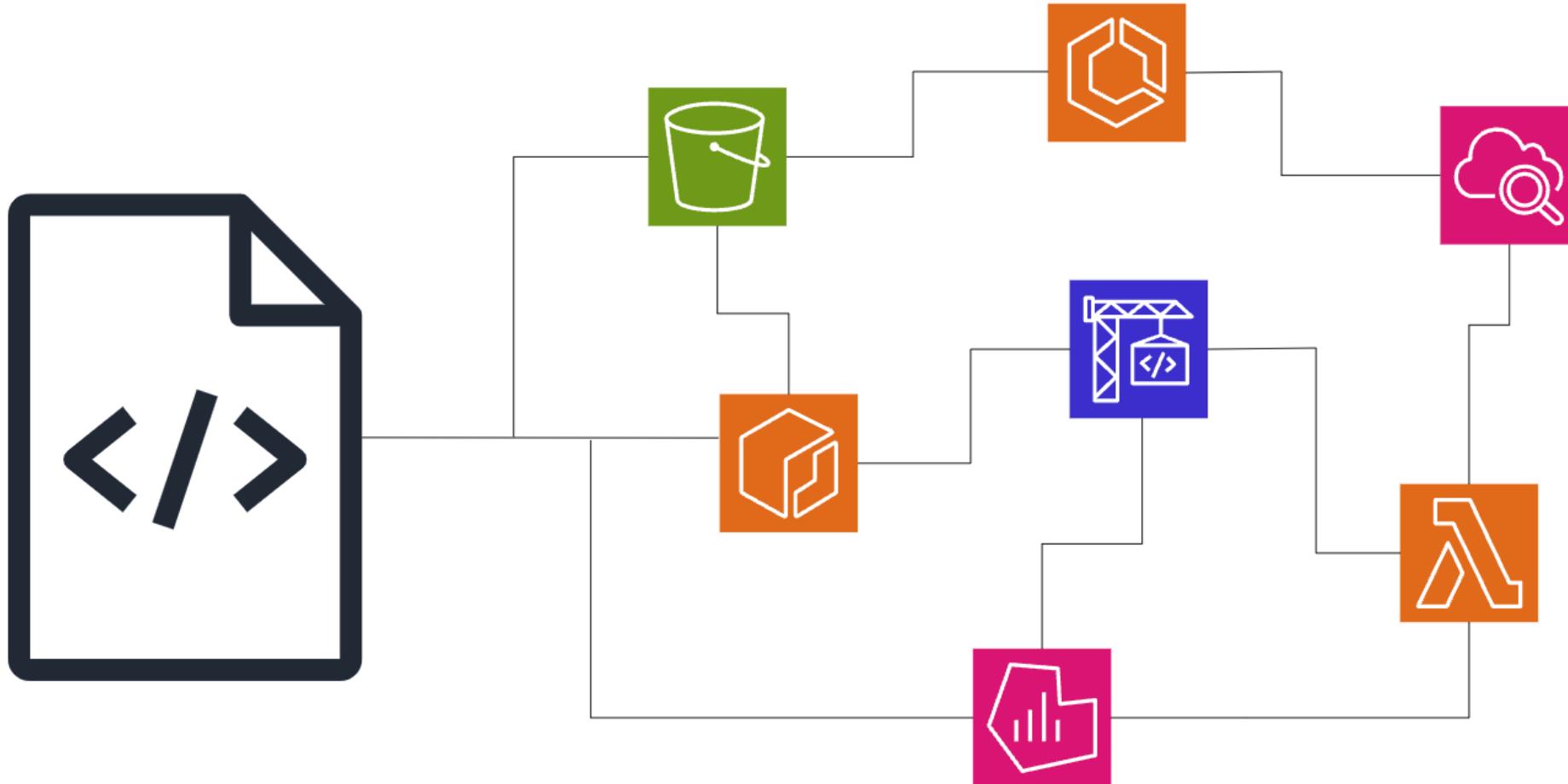
프로젝트 개요 - 모니터링

DevOps 모니터링 구현



프로젝트 개요 - IaC

IaC를 통해 인프라를 코드로 관리



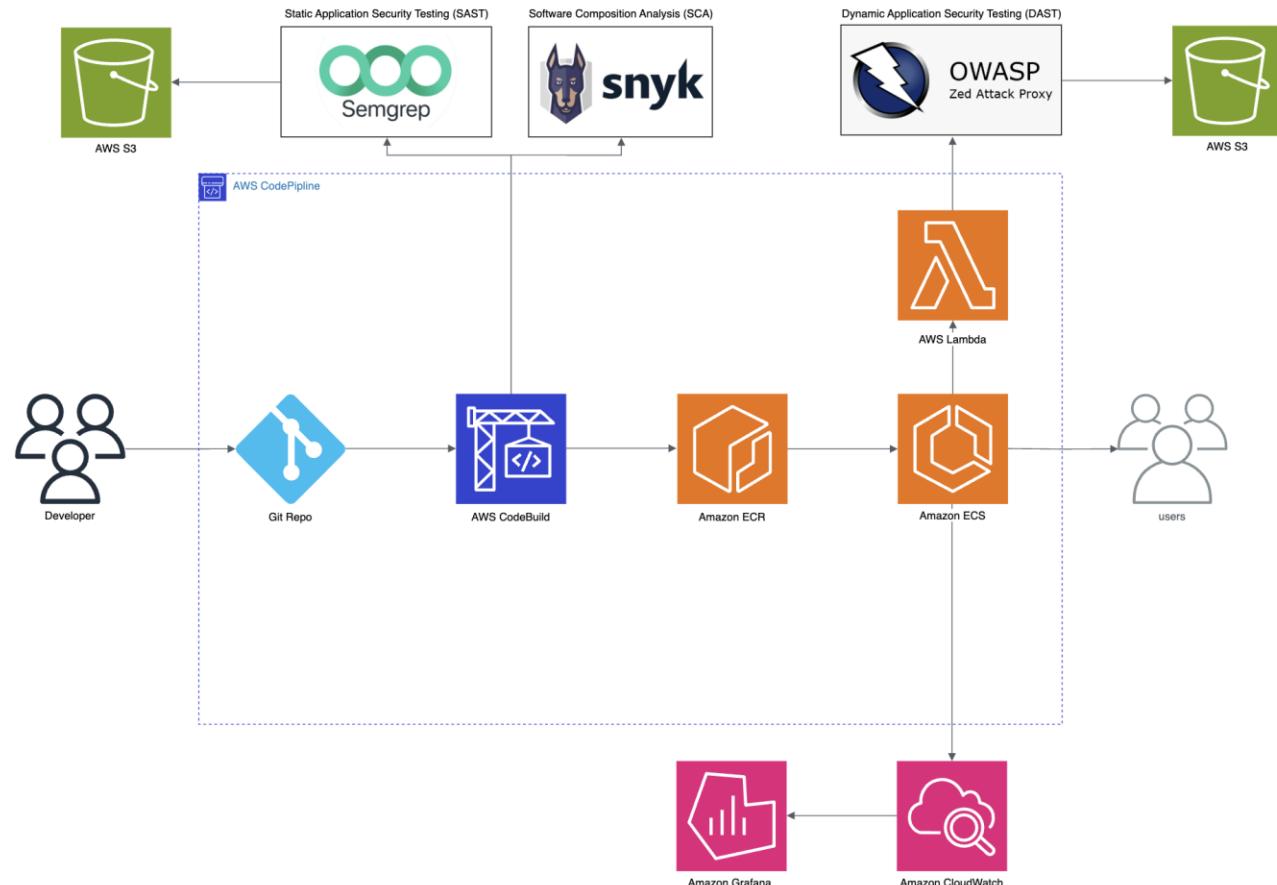
프로젝트 진행 내용

- CI/CD 파이프라인 생성
- SCA, SAST, DAST 테스트 및 통합
- 모니터링 기능 및 IaC 구현



CI/CD 파이프라인 생성

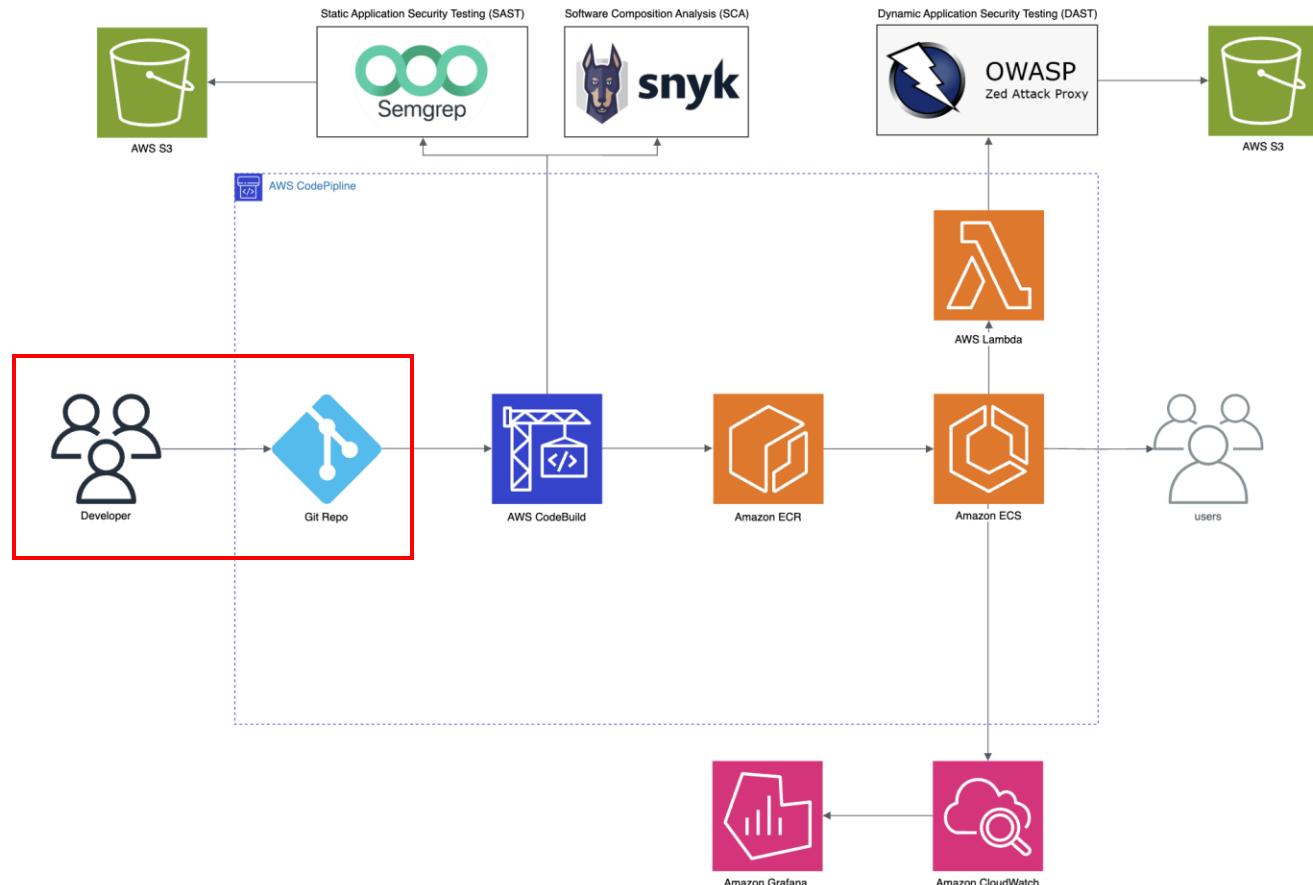
Security에 집중한 DevOps CI/CD 파이프라인



- Git에서 Code Push
- CodeBuild에서 Docker 이미지 생성, SCA, SAST 분석 진행
- ECR에 Build 이미지 저장
- ECS 서비스 배포
- Lambda에서 DAST ECS Task를 호출하여 서비스 분석
- 서비스의 인프라를 지속적으로 모니터링

CI/CD 파이프라인 생성

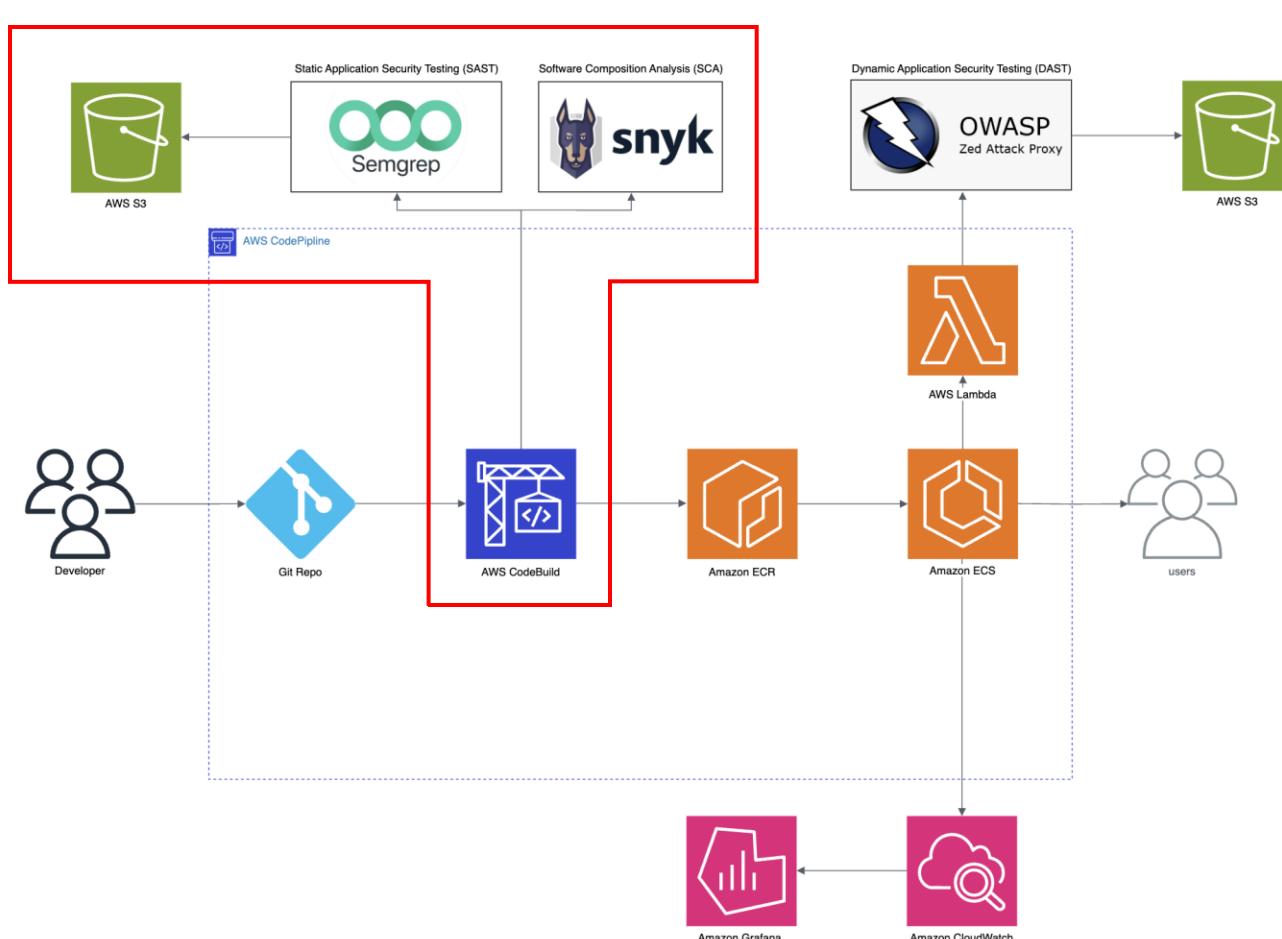
GitHub에서 Code가 Push되면 파이프라인이 트리거



- Git에서 Code Push
- CodeBuild에서 Docker 이미지 생성, SCA, SAST 분석 진행
- ECR에 Build 이미지 저장
- ECS 서비스 배포
- Lambda에서 DAST ECS Task를 호출하여 서비스 분석
- 서비스의 인프라를 지속적으로 모니터링

CI/CD 파이프라인 생성

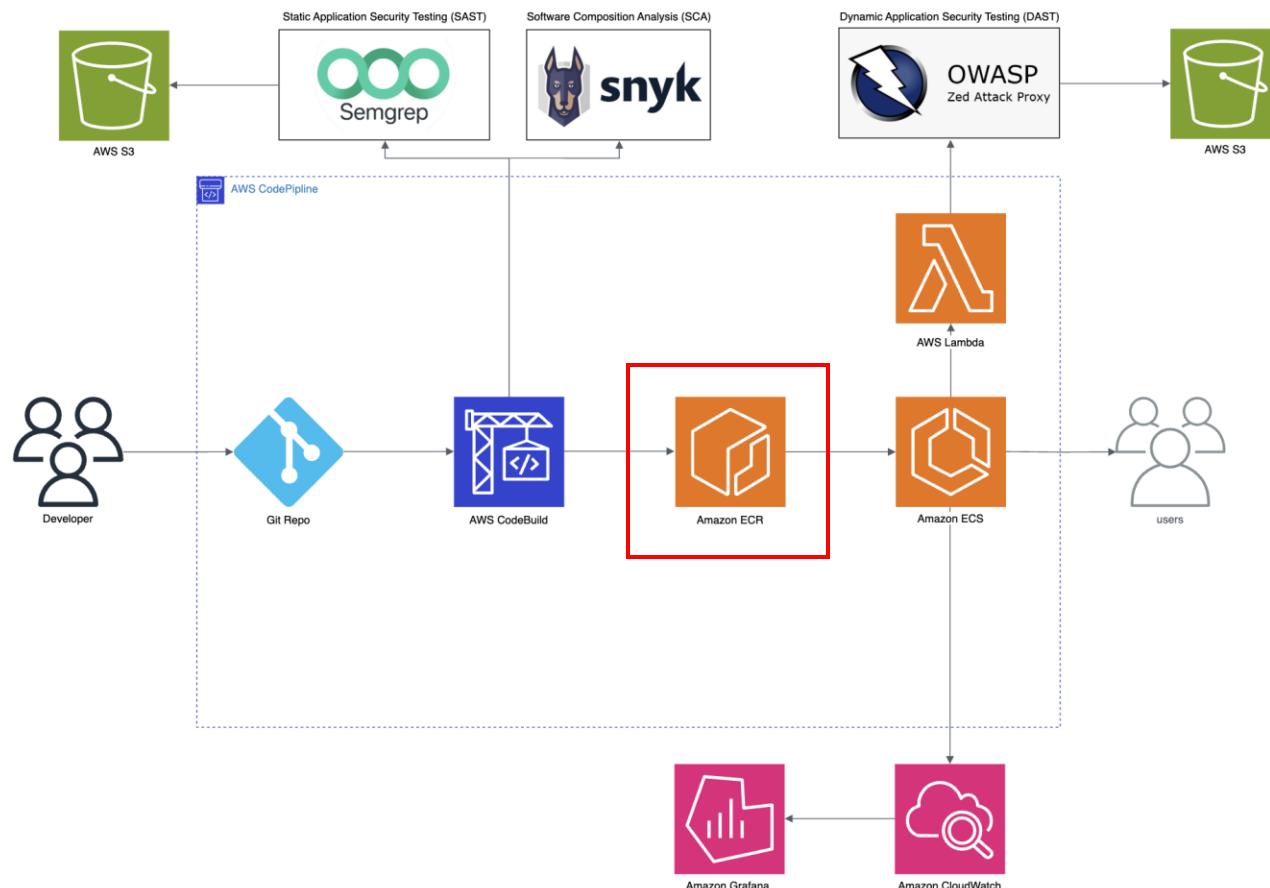
Docker 이미지 생성 및 SCA, SAST 분석 후 S3 버킷에 업로드



- Git에서 Code Push
- CodeBuild에서 Docker 이미지 생성, SCA, SAST 분석 진행
- ECR에 Build 이미지 저장
- ECS 서비스 배포
- Lambda에서 DAST ECS Task를 호출하여 서비스 분석
- 서비스의 인프라를 지속적으로 모니터링

CI/CD 파이프라인 생성

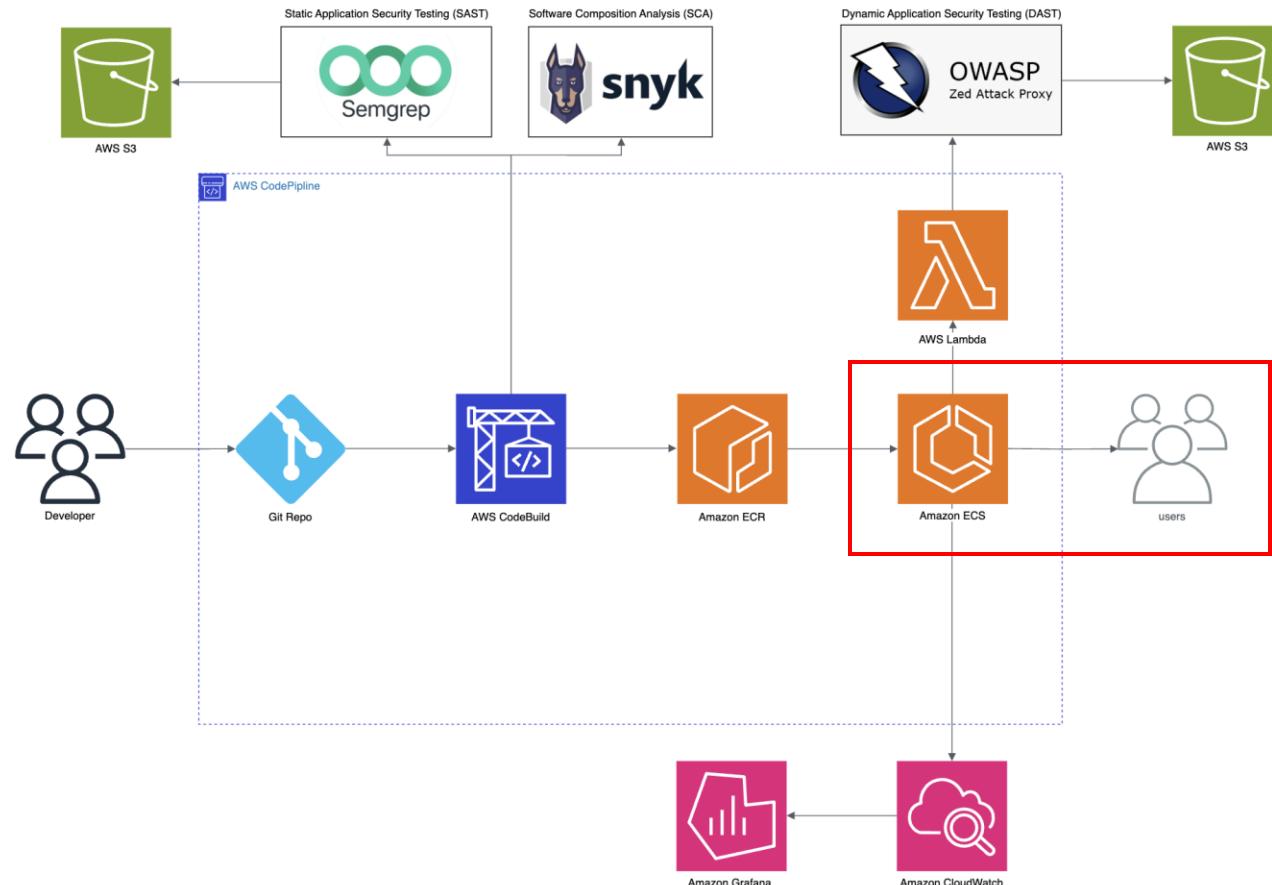
빌드된 Docker 이미지를 ECR 프라이빗 리포지토리에 저장



- Git에서 Code Push
- CodeBuild에서 Docker 이미지 생성, SCA, SAST 분석 진행
- ECR에 Build 이미지 저장**
- ECS 서비스 배포
- Lambda에서 DAST ECS Task를 호출하여 서비스 분석
- 서비스의 인프라를 지속적으로 모니터링

CI/CD 파이프라인 생성

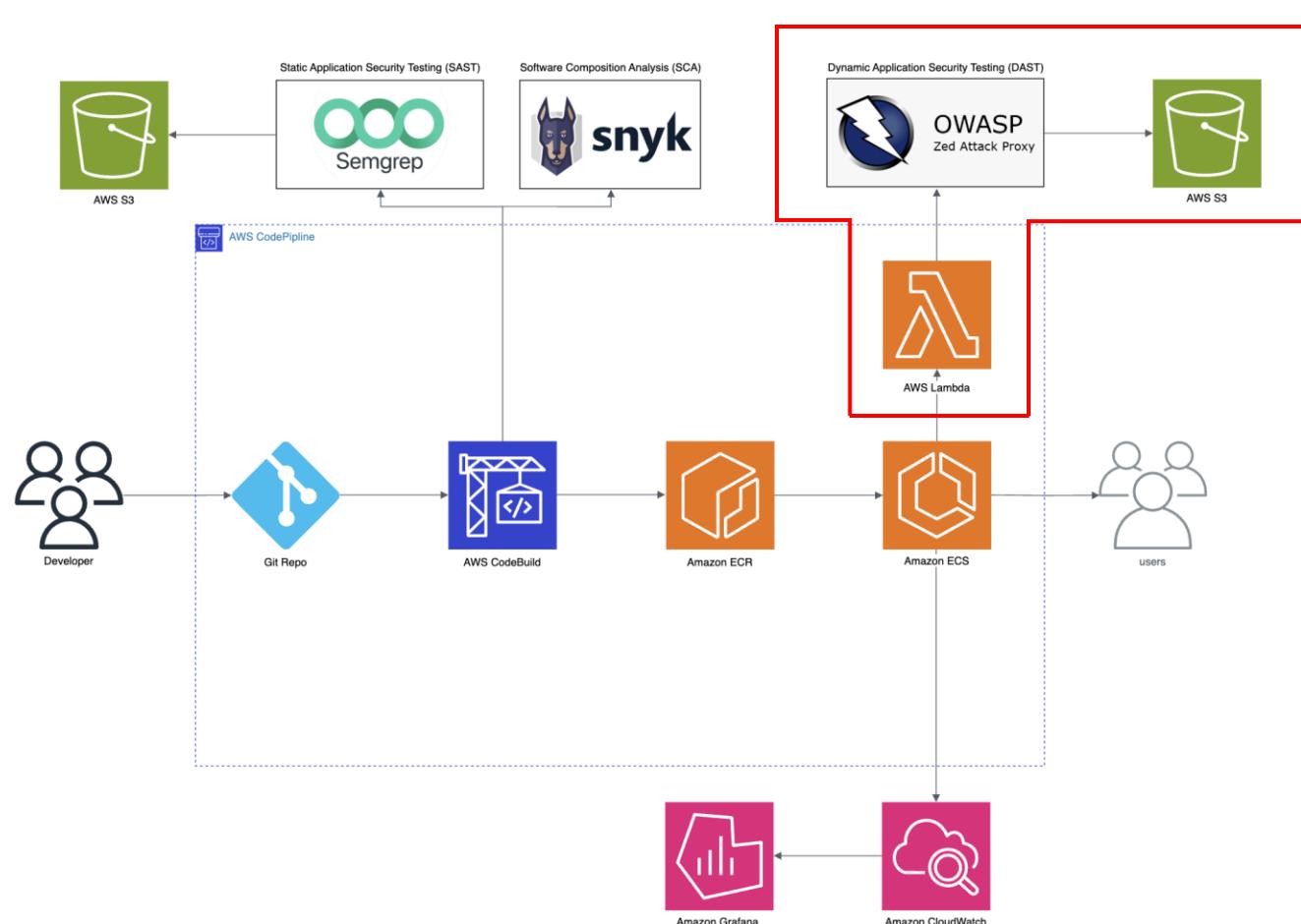
지정된 Cluster에 Task 정의를 통해 서비스를 Rolling Update 배포



- Git에서 Code Push
- CodeBuild에서 Docker 이미지 생성, SCA, SAST 분석 진행
- ECR에 Build 이미지 저장
- ECS 서비스 배포**
- Lambda에서 DAST ECS Task를 호출하여 서비스 분석
- 서비스의 인프라를 지속적으로 모니터링

CI/CD 파이프라인 생성

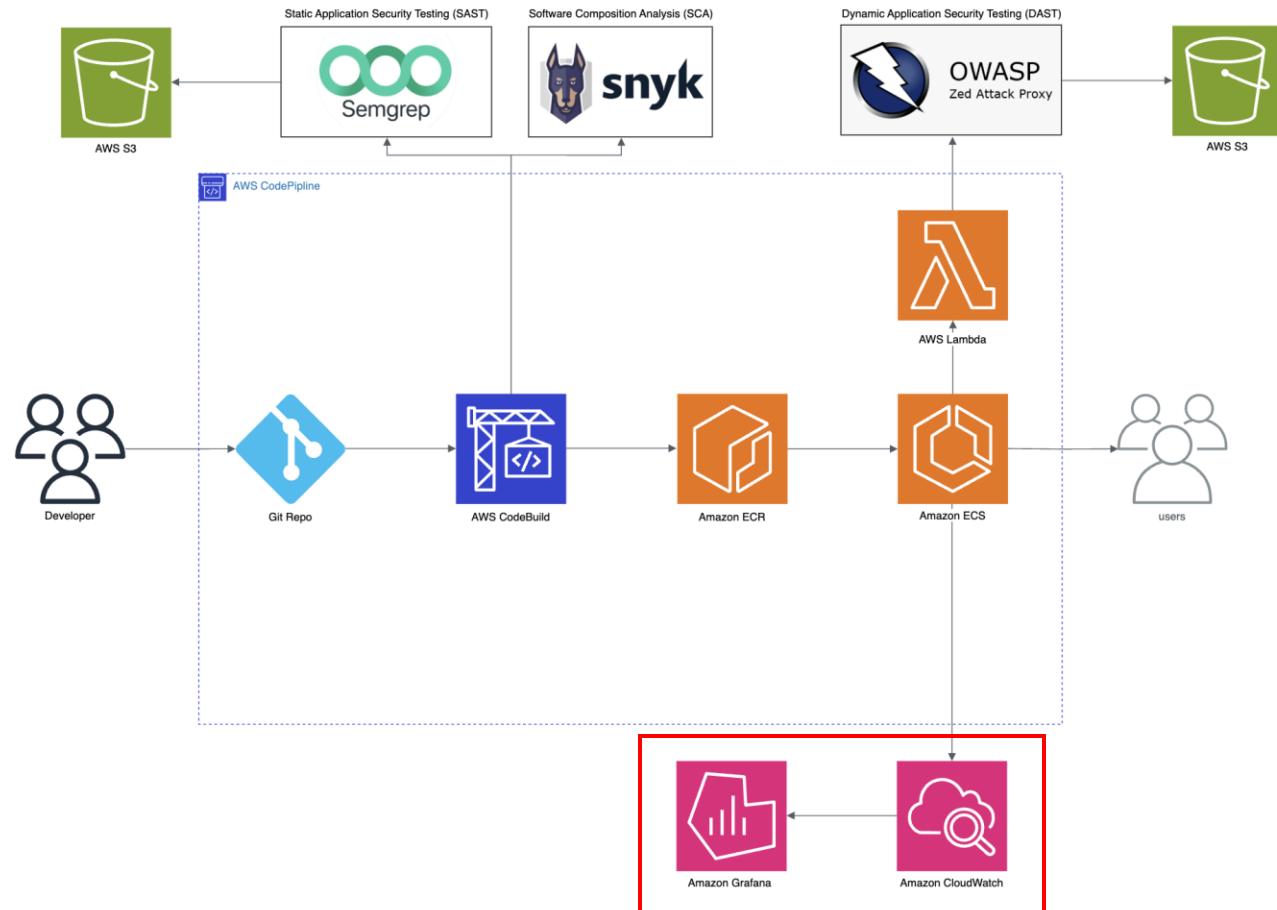
OWASP ZAP으로 ECS 서비스 스캔 후 S3 버킷에 업로드



- Git에서 Code Push
- CodeBuild에서 Docker 이미지 생성, SCA, SAST 분석 진행
- ECR에 Build 이미지 저장
- ECS 서비스 배포
- Lambda에서 DAST ECS Task를 호출하여 서비스 분석
- 서비스의 인프라를 지속적으로 모니터링

CI/CD 파이프라인 생성

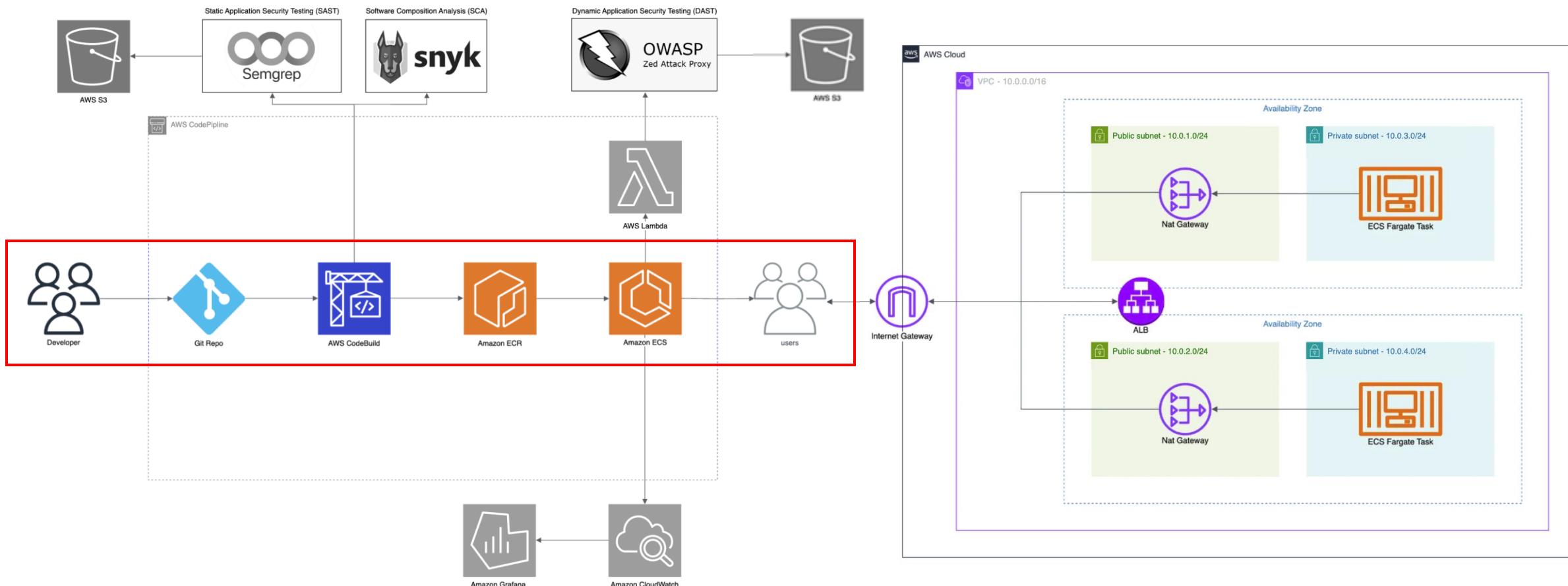
CloudWatch의 로그 및 메트릭 수집 내용을 Grafana로 시각화



- Git에서 Code Push
- CodeBuild에서 Docker 이미지 생성, SCA, SAST 분석 진행
- ECR에 Build 이미지 저장
- ECS 서비스 배포
- Lambda에서 DAST ECS Task를 호출하여 서비스 분석
- 서비스의 인프라를 지속적으로 모니터링

CI/CD 파이프라인 생성

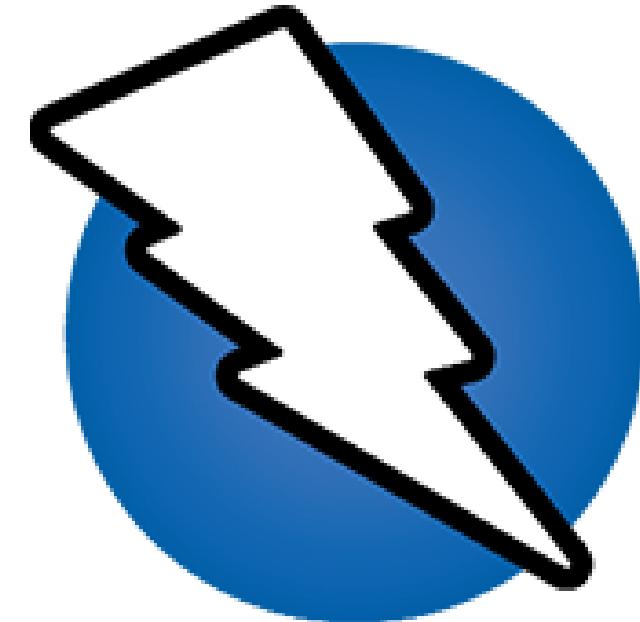
중간 발표 전 진행 내용



SCA, SAST, DAST 테스트 및 통합

SAST, DAST 도구 선정 결과

 Semgrep



OWASP ZAP

SCA, SAST, DAST 테스트 및 통합

SCA(Software Composition Analysis)란?

- 애플리케이션과 컨테이너 내 오픈 소스 및 서드 파티 구성 요소를 식별하고
보안 취약점과 라이선스 문제를 분석하는 도구



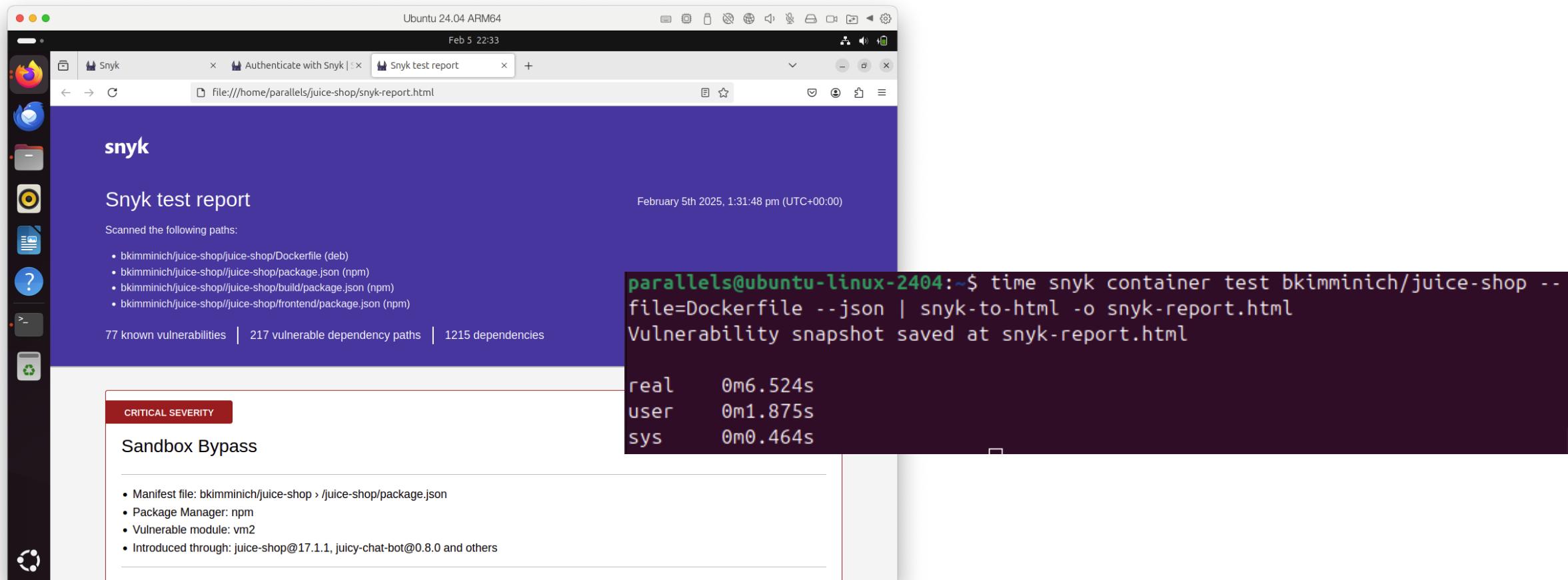
SCA, SAST, DAST 테스트 및 통합

Dependency-Check

```
README.md          config      encryptionkeys      node_modules      swagger.yaml
lo3ml@DESKTOP-IP8TULM:~/juice-shop$ time docker run --rm -v $(pwd):/src owasp/dependency-check --scan /src --format HTML
--out /src/report.html --nvdApiKey="d8a25c63-3a50-4c35-a9da-2a95938fbf6d"
[INFO] Checking for updates
[INFO] NVD API has 280,121 records in this update
[WARN] Retrying request /rest/json/cves/2.0?resultsPerPage=2000&startIndex=2000 : 3rd time
[INFO] Downloaded 10,000/280,121 (4%)
[INFO] Downloaded 20,000/280,121 (7%)
[INFO] Downloaded 30,000/280,121 (11%)
[INFO] Downloaded 40,000/280,121 (14%)
[INFO] Downloaded 50,000/280,121 (18%)
[INFO] Downloaded 60,000/280,121 (21%)
[WARN] Retrying request /rest/json/cves/2.0?resultsPerPage=2000&startIndex=68000 : 3rd time
[INFO] Downloaded 70,000/280,121 (25%)
[INFO] Downloaded 80,000/280,121 (29%)
[INFO] Downloaded 90,000/280,121 (32%)
[INFO] Downloaded 100,000/280,121 (36%)
[WARN] Retrying request /rest/json/cves/2.0?resultsPerPage=2000&startIndex=112000 : 3rd time
[WARN] Retrying request /rest/json/cves/2.0?resultsPerPage=2000&startIndex=114000 : 3rd time
[WARN] Retrying request /rest/json/cves/2.0?resultsPerPage=2000&startIndex=108000 : 3rd time
[INFO] Downloaded 110,000/280,121 (39%)
[INFO] Downloaded 120,000/280,121 (43%)
[INFO] Downloaded 130,000/280,122 (46%)
[INFO] Downloaded 140,000/280,122 (50%)
[INFO] Downloaded 150,000/280,122 (54%)
[INFO] Downloaded 160,000/280,122 (57%)
[INFO] Downloaded 170,000/280,122 (61%)
[WARN] Retrying request /rest/json/cves/2.0?resultsPerPage=2000&startIndex=176000 : 3rd time
[INFO] Downloaded 180,000/280,122 (64%)
```

SCA, SAST, DAST 테스트 및 통합

Snyk



SCA, SAST, DAST 테스트 및 통합

Trivy

Total: 28 (UNKNOWN: 1, LOW: 14, MEDIUM: 13, HIGH: 0, CRITICAL: 0)

Library	Vulnerability	Severity	Status	Installed Version
libc6	CVE-2023-4806	MEDIUM	will_not_fix	2.31-13+deb11u10
	CVE-2023-4813		affected	
	CVE-2025-0395		fix_deferred	
	CVE-2010-4756	LOW	affected	
	CVE-2018-20796			
	CVE-2019-1010022			
	CVE-2019-1010023			
	CVE-2019-1010024			

```
trivy image --scanners vuln,config bkimminich/juice-shop:latest
```

- ▶ 결과
- 탐지 시간 ⇒ 2.491초

```
real 0m2.491s
user 0m0.496s
sys 0m0.196s
```

SCA, SAST, DAST 테스트 및 통합

SCA 도구 테스트 결과 비교

	Dependency-Check	Snyk	Trivy
총 발견된 취약점 개수	45개	75개	28개
스캔 속도	약 3분 21초	약 6초	약 2초
리포트	CLI, HTML, JSON	CLI, HTML, JSON, 웹 UI	CLI, JSON

SCA, SAST, DAST 테스트 및 통합

SCA 도구 테스트 결과 비교

	DependencyCheck
총 발견된 취약점 개수	45개
스캔 속도	약 3분 2초
리포트	CLI, HTML, JSON



	Trivy
	28개
	약 2초
	CLI, JSON

SCA 파이프라인 통합

Snyk에서 Auth Token 생성

The screenshot shows the Snyk account settings page under the 'General' tab. On the left sidebar, 'Auth Token' is highlighted. The main content area displays the 'Auth Token' section, which includes a key field (redacted), a creation date ('09 February 2025, 17:01:36'), and a 'Revoke & Regenerate' button. Below this is the 'Authorized Applications' section, which currently shows 'No applications'. At the bottom is the 'Preferred Organization' section, where 'DevOpsSCP' is selected.

SnykToken

보안 암호 세부 정보

암호화 키

aws/secretsmanager

보안 암호 이름

SnykToken

보안 암호 ARN

arn:aws:secretsmanager:ap-northeast-2:711387094022:secret:SnykToken-BjPGuS

개요 교체 버전 복제 태그

보안 암호 값 정보

보안 암호 값을 검색하고 확인합니다.

키/값

일반 텍스트

보안 암호 키

보안 암호 값

SNYK_TOKEN

13181342-c49f-44e9-b9f4-2e5c

SecretManager를 통해 Token 암호화

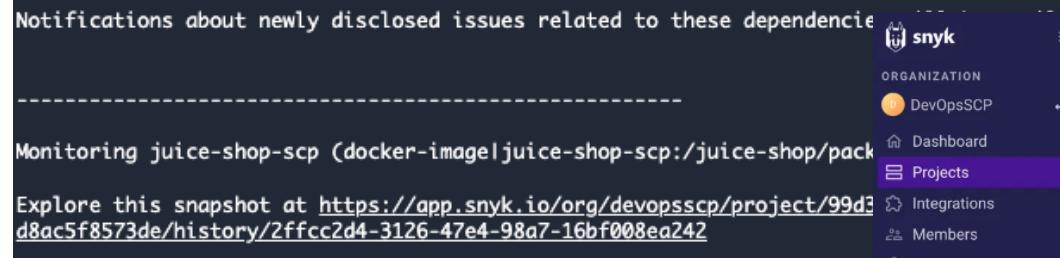
SCA 파이프라인 통합

```
[Container] 2025/02/23 06:34:36.713140 Entering phase POST_BUILD
[Container] 2025/02/23 06:34:36.714181 Running command echo Uploading Snyk scan results to the Snyk dashboard...
Uploading Snyk scan results to the Snyk dashboard...

[Container] 2025/02/23 06:34:36.720912 Running command snyk container monitor juice-shop-scp --file=Dockerfile || echo "Snyk monitor completed with issues"

Monitoring juice-shop-scp (docker-image|juice-shop-scp)...

Explore this snapshot at https://app.snyk.io/org/devopsscp/project/389243ea-2cac-4315-9fb4-732d2f4a0f47/history/ea5939e3-fb2f-41d4-8e1c-8b614faa8516
```



Snyk 대시보드에서
분석결과 확인 가능

DevOpsSCP > Projects

All projects

Add filter

Targets 1

juice-shop-scp

Reference Issues

juice-shop-scp

Project Imported Tested Issues

Project	Imported	Tested	Issues
docker-image juice-shop-scp:/juice-shop/package.json	3 minutes ago	3 minutes ago	5 C 22 H 29 M 19 L
juice-shop-scp	3 minutes ago	3 minutes ago	0 C 0 H 2 M 16 L

Ready to import another project?

Secure your entire stack with Snyk

Add projects

SAST 파이프라인 통합

Semgrep 로그인을 위한 Token 생성

The screenshot shows the Semgrep API tokens page under the Settings tab. It includes a note about using tokens for identity verification and saving them in GitHub Secrets. A search bar and a 'Create new token' button are also visible.

SemgrepToken

보안 암호 세부 정보

암호화 키
aws/secretsmanager

보안 암호 이름
SemgrepToken

보안 암호 ARN
arn:aws:secretsmanager:ap-northeast-2:711387094022:secret:SemgrepToken-5NMban

개요 교체 버전 복제 태그

보안 암호 값 정보

보안 암호 값을 검색하고 확인합니다.

키/값 일반 텍스트

보안 암호 키

SEMGREP_APP_TOKEN

보안 암호 값

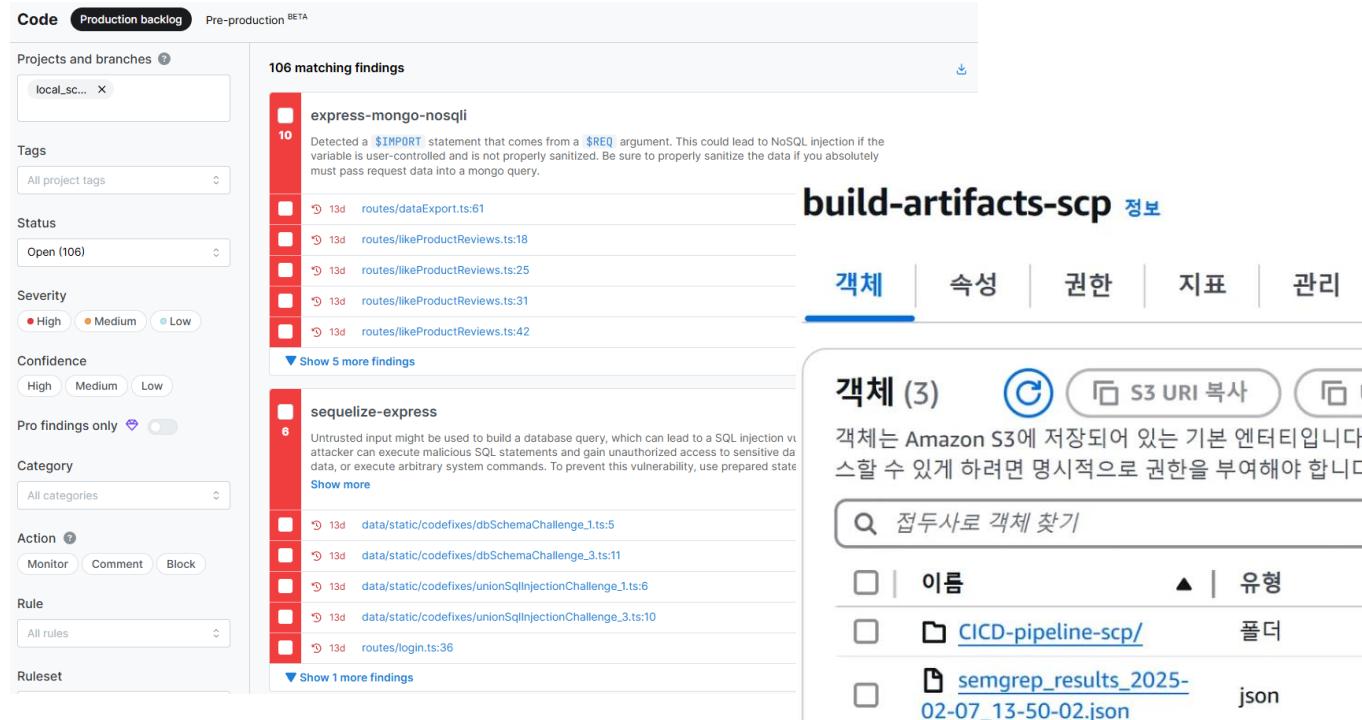
ddc821c271bd3ee68ab742e1b1b5f

SecretManager를 통해 Token 암호화

SAST 파이프라인 통합

```
[Container] 2025/02/23 06:34:17.243697 Running command echo "Uploading Semgrep results to S3..."  
Uploading Semgrep results to S3...  
  
[Container] 2025/02/23 06:34:17.249365 Running command aws s3 cp semgrep_results_$TIMESTAMP.json s3://build-artifacts-scp/semgrep_results_$TIMESTAMP.json  
upload: ./semgrep_results_2025-02-23_06-34-05.json to s3://build-artifacts-scp/semgrep_results_2025-02-23_06-34-05.json
```

Buildspec.yml 파일에
Semgrep을 추가하고
Build가 진행되면



The screenshot shows two log entries from a container. The first entry is a status message about uploading Semgrep results to S3. The second entry is a command using the AWS CLI to copy the Semgrep results JSON file from the local directory to an S3 bucket named 'build-artifacts-scp'.

Semgrep Results in AWS CloudWatch Logs:

```
[Container] 2025/02/23 06:34:17.243697 Running command echo "Uploading Semgrep results to S3..."  
Uploading Semgrep results to S3...  
  
[Container] 2025/02/23 06:34:17.249365 Running command aws s3 cp semgrep_results_$TIMESTAMP.json s3://build-artifacts-scp/semgrep_results_$TIMESTAMP.json  
upload: ./semgrep_results_2025-02-23_06-34-05.json to s3://build-artifacts-scp/semgrep_results_2025-02-23_06-34-05.json
```

Semgrep UI and AWS S3 Bucket Details:

The UI on the left shows the Semgrep dashboard with filtering options like 'local_sc...', 'All project tags', 'Open (106)', and 'Medium'. It lists 106 matching findings under categories like 'express-mongo-nosql' and 'sequelize-express', with details such as file paths and line numbers.

The UI on the right shows the 'build-artifacts-scp' S3 bucket details. It has tabs for '객체' (Objects), '속성' (Attributes), '권한' (Permissions), '지표' (Metrics), '관리' (Management), and '액세스 지점' (Access Points). The '객체' tab shows 3 objects: 'CICD-pipeline-scp/' (Type: 폴더, Last modified: 2025. 2. 7. pm 10:55:43 PM KST) and 'semgrep_results_2025-02-07_13-50-02.json' (Type: json, Last modified: 2025. 2. 7. pm 10:55:43 PM KST).

S3 버킷에서
Semgrep json 파일
또는 대시보드에서
확인 가능

SCA, SAST, DAST 테스트 및 통합

SCA, SAST 통합 전 빌드 시간 : 14m 39s

이름	상태	컨텍스트	기간	시작 시간	종료 시간
SUBMITTED	✅ 성공함	-	<1 sec	2월 10, 2025 10:30 오후 (UTC+9:00)	2월 10, 2025 10:30 오후 (UTC+9:00)
QUEUED	✅ 성공함	-	<1 sec	2월 10, 2025 10:30 오후 (UTC+9:00)	2월 10, 2025 10:30 오후 (UTC+9:00)
PROVISIONING	✅ 성공함	-	5 secs	2월 10, 2025 10:30 오후 (UTC+9:00)	2월 10, 2025 10:30 오후 (UTC+9:00)
DOWNLOAD_SOURCE	✅ 성공함	-	2 secs	2월 10, 2025 10:30 오후 (UTC+9:00)	2월 10, 2025 10:30 오후 (UTC+9:00)
INSTALL	✅ 성공함	-	<1 sec	2월 10, 2025 10:30 오후 (UTC+9:00)	2월 10, 2025 10:30 오후 (UTC+9:00)
PRE_BUILD	✅ 성공함	-	12 secs	2월 10, 2025 10:30 오후 (UTC+9:00)	2월 10, 2025 10:30 오후 (UTC+9:00)
BUILD	✅ 성공함	-	839 secs	2월 10, 2025 10:30 오후 (UTC+9:00)	2월 10, 2025 10:44 오후 (UTC+9:00)
POST_BUILD	✅ 성공함	-	18 secs	2월 10, 2025 10:44 오후 (UTC+9:00)	2월 10, 2025 10:45 오후 (UTC+9:00)
UPLOAD_ARTIFACTS	✅ 성공함	-	<1 sec	2월 10, 2025 10:45 오후 (UTC+9:00)	2월 10, 2025 10:45 오후 (UTC+9:00)
FINALIZING	✅ 성공함	-	<1 sec	2월 10, 2025 10:45 오후 (UTC+9:00)	2월 10, 2025 10:45 오후 (UTC+9:00)
COMPLETED	✅ 성공함	-	-	2월 10, 2025 10:45 오후 (UTC+9:00)	-

SCA, SAST, DAST 테스트 및 통합

SCA, SAST 통합 후 빌드 시간 : 17m 44s

이름	상태	컨텍스트	기간	시작 시간	종료 시간
SUBMITTED	✓ 성공함	-	<1 sec	2월 23, 2025 3:17 오후 (UTC+9:00)	2월 23, 2025 3:17 오후 (UTC+9:00)
QUEUED	✓ 성공함	-	<1 sec	2월 23, 2025 3:17 오후 (UTC+9:00)	2월 23, 2025 3:17 오후 (UTC+9:00)
PROVISIONING	✓ 성공함	-	5 secs	2월 23, 2025 3:17 오후 (UTC+9:00)	2월 23, 2025 3:17 오후 (UTC+9:00)
DOWNLOAD_SOURCE	✓ 성공함	-	2 secs	2월 23, 2025 3:17 오후 (UTC+9:00)	2월 23, 2025 3:17 오후 (UTC+9:00)
INSTALL	✓ 성공함	-	<1 sec	2월 23, 2025 3:17 오후 (UTC+9:00)	2월 23, 2025 3:17 오후 (UTC+9:00)
PRE_BUILD	✓ 성공함	-	73 secs	2월 23, 2025 3:17 오후 (UTC+9:00)	2월 23, 2025 3:18 오후 (UTC+9:00)
BUILD	✓ 성공함	-	945 secs	2월 23, 2025 3:18 오후 (UTC+9:00)	2월 23, 2025 3:34 오후 (UTC+9:00)
POST_BUILD	✓ 성공함	-	35 secs	2월 23, 2025 3:34 오후 (UTC+9:00)	2월 23, 2025 3:35 오후 (UTC+9:00)
UPLOAD_ARTIFACTS	✓ 성공함	-	<1 sec	2월 23, 2025 3:35 오후 (UTC+9:00)	2월 23, 2025 3:35 오후 (UTC+9:00)
FINALIZING	✓ 성공함	-	<1 sec	2월 23, 2025 3:35 오후 (UTC+9:00)	2월 23, 2025 3:35 오후 (UTC+9:00)
COMPLETED	✓ 성공함	-	-	2월 23, 2025 3:35 오후 (UTC+9:00)	-

DAST 파이프라인 통합

Task 정의 생성

태스크 정의 구성

태스크 정의 패밀리 | 정보
고유한 태스크 정의 패밀리 이름을 지정합니다.

dast-zap-task

최대 255개의 문자(대문자 및 소문자), 숫자, 하이픈 및 밑줄이 허용됩니다.

▼ 인프라 요구 사항

태스크 정의에 대한 인프라 요구 사항을 지정합니다.

시작 유형 | 정보
시작 유형을 선택하면 작업 정의 파라미터가 변경됩니다.

AWS Fargate
컨테이너를 서비스 컴퓨팅입니다.

Amazon EC2 인스턴스
Amazon EC2 인스턴스를 사용하는 자체 관리형 인프라입니다.

OS, 네트워크 모드
네트워크 모드는 작업에 사용되는 선택한 컴퓨팅 유형에 따라 달라집니다.

운영 체제/아키텍처 | 정보
aws vpc

네트워크 모드 | 정보
Linux/X86_64

태스크 크기 | 정보
태스크를 위해 예약할 CPU 및 메모리의 양을 지정합니다.

CPU
1 vCPU

메모리
4 GB

▼ 컨테이너 - 1 정보

컨테이너 세부 정보
이름과 컨테이너 이미지를 지정하고 컨테이너를 필수로 표시할지 지정합니다. 각 태스크 정의에는 필수 컨테이너 하나 이상 있어야 합니다.

이름 | 정보
owasp-zap

이미지 URI | 정보
711387094022.dkr.ecr.ap-northeast-2.amazonaws.com/owasp-zap:latest

필수 컨테이너 | 정보
에

프라이빗 레지스트리 | 정보
Secrets Manager에 보안 인증 정보를 저장한 다음 보안 인증 정보를 사용하여 프라이빗 레지스트리의 이미지를 참조합니다.

프라이빗 레지스트리 인증

포트 맵핑 | 정보
포트 맵핑을 추가하여 컨테이너가 호스트의 포트에 액세스하여 트래픽을 보내거나 받을 수 있도록 합니다. 포트 이름은 비워 두면 기본값이 할당됩니다.

포트 맵핑 추가

위기 전용 루트 파일 시스템 | 정보
이 파라미터를 헌 경우 컨테이너에는 루트 파일 시스템에 대한 위기 전용 엑세스가 부여됩니다.

위기 전용

리소스 할당 제한 - 조건부 | 정보
컨테이너 수준 CPU, GPU 및 메모리 제한은 태스크 수준 갑과 다르게 컨테이너에 할당되는 리소스의 양을 정의합니다. 컨테이너가 하드 제한에 지정된 메모리를 초과하거나 시도하면 컨테이너가 종료됩니다.

CPU
1

GPU
1

GPU
4

메모리 하드 제한
단위 기가바이트
2

메모리 소프트 제한
단위 기가바이트
2

이름
owasp-zap

최대 255개의 문자(대문자 및 소문자), 숫자, 하이픈 및 밑줄이 허용됩니다.

DAST 파이프라인 통합

Task 실행 시 명령어가 실행되도록 설정

▼ 컨테이너 재정의 [정보](#)

▼ owasp-zap

명령 재정의

컨테이너로 전달되는 Docker 명령입니다. 자세한 내용은 [Docker 실행 참조](#) 를 확인하세요.

```
sh,-c,"mkdir -p /zap/wrk && zap-baseline.py -t http://10.0.3.217:3000 -r /zap/wrk/zap_baseline_report.html && aws s3 cp /zap/wrk/zap_full_report.html  
s3://dast-owasp-zap/zap_baseline_report_${(date +%Y-%m-%d_%H-%M-%S)}.html"
```

쉼표로 구분된 명령 및 매개변수

환경 변수 재정의

기본 컨테이너 변수를 재정의하는 환경 변수입니다.

[환경 변수 추가](#)

환경 파일 재정의

기본 컨테이너 파일을 재정의하는 환경 파일입니다.

[환경 파일 추가](#)

환경 파일을 10개 더 추가할 수 있습니다.

DAST 파이프라인 통합

VPC 내에서 서로 접근을 위한 CloudMap 네임스페이스 및 서비스 생성

네임스페이스 구성

네임스페이스 구성은 애플리케이션이 서비스 인스턴스를 검색하는 방법을 결정합니다.

네임스페이스 이름

기억하기 쉬운 이름을 사용하면 대시보드에서 네임스페이스를 쉽게 찾을 수 있습니다.

juice-shop.local

네임스페이스 이름은 최대 1,024자로 지정할 수 있으며 글자로 시작하고 끝나야 합니다.

유효한 문자: a-z, A-Z, 0-9, _(마침표), _(밑줄) 및 -(하이픈)

네임스페이스 설명 - 선택 사항

이 설명은 대시보드에 표시되며 네임스페이스의 용도를 빠르게 식별하는 데 도움이 됩니다.

새 결제 애플리케이션의 네임스페이스입니다.

설명은 최대 1,024자까지 입력할 수 있습니다.

인스턴스 검색

인스턴스 검색은 애플리케이션이 등록된 인스턴스를 검색하는 방법을 결정합니다.

API 호출

애플리케이션은 API 호출을 통해 등록된 인스턴스를 검색합니다.

VPC에서 API 호출 및 DNS 쿼리

애플리케이션이 VPC에서 API를 호출하거나 DNS 쿼리를 제출하여 등록된 인스턴스를 검색합니다. 추가 요금이 발생합니다.

API 호출 및 퍼블릭 DNS 쿼리

애플리케이션이 API를 호출하거나 퍼블릭 DNS 쿼리를 제출하여 등록된 인스턴스를 검색합니다. 추가 요금이 발생합니다.

VPC

네임스페이스를 연결하려는 Amazon VPC입니다.

vpc-0df030e7c3f1f54fb (juice-shop-scp)

TTL

네임스페이스를 사용하여 생성된 Route 53 호스팅 영역의 SOA TTL 값입니다. 부정적으로 캐시된 결과가 저장되는 기간을 결정합니다. TTL 값이 낮을수록 DNS에 대한 [요금 영향](#)이(가) 발생합니다.

60

값은 초 단위입니다.

서비스 정보

서비스 이름

기억하기 쉬운 이름을 사용하면 대시보드에서 서비스를 쉽게 찾을 수 있습니다.

scp-juice-shop

서비스 이름은 최대 1,024자까지 가능합니다.

서비스 설명 - 선택 사항

설명은 이 서비스에 대한 세부 정보를 기억하는 데 도움이 될 수 있습니다.

결제 환급을 위한 서비스입니다.

서비스 검색 구성

검색 가능 방법

서비스 검색 방법

API 및 DNS

API 호출 및 DNS 쿼리로 검색 가능

API 전용

API 호출로만 검색 가능

DNS 구성

라우팅 정책

이 서비스를 사용하여 인스턴스를 등록할 때 Cloud Map이 생성하는 Route 53 DNS 레코드에 대한 라우팅 정책을 선택합니다.

WEIGHTED

레코드 유형

레코드 유형은 DNS 쿼리에 대한 응답으로 반환되는 값의 형식을 결정합니다.

A

TTL

TTL(Time to Live)은 DNS recursive resolver가 레코드의 설정을 캐싱하는 시간(초)을 결정합니다.

60

DAST 파이프라인 통합

Juice shop ECS 서비스 설정에 서비스 검색으로 네임스페이스 추가

▼ 서비스 검색- 선택 사항

서비스 검색에서는 Amazon Route 53을 사용하여 서비스의 네임스페이스를 생성하며, 이 네임스페이스는 DNS를 통해 찾을 수 있습니다.

서비스 검색 사용
서비스 검색을 구성하여 DNS를 통해 찾을 수 있는 네임스페이스를 생성하세요.

네임스페이스 구성
네임스페이스는 일반적으로 동일한 도메인 이름을 공유하는 애플리케이션에 대한 서비스를 포함합니다.

새 네임스페이스 생성
 기존 네임스페이스 선택

기존 네임스페이스
네임스페이스를 선택하여 애플리케이션을 구성하는 서비스 그룹을 지정합니다. 네임스페이스가 없는 경우 AWS Cloud Map [?]에서 네임스페이스를 생성할 수 있습니다.

새 서비스 검색 서비스 생성
 기존 서비스 검색 서비스 선택

기존 서비스 검색 서비스
서비스 인스턴스를 등록하는 데 사용할 서비스 검색 서비스를 선택합니다. 서비스가 없는 경우 AWS Cloud Map [?]에서 서비스를 생성할 수 있습니다.

DNS 레코드
AWS Cloud Map에서 인스턴스를 등록할 때 생성해야 하는 DNS 레코드를 지정합니다.

DNS 레코드 - 1개

DNS 레코드 유형

A

TTL

60

초

DAST 파이프라인 통합

Lambda 함수 생성

함수 생성

다음 옵션 중 하나를 선택하여 함수를 생성합니다.

새로 작성
간단한 Hello World 예제는 시작하십시오.

블루프린트 사용
샘플 코드 및 구축 Lambda 애플리케이션을 위한 구성 사전 설정을 일반적인 사용 사례를 살펴봅니다.

컨테이너 이미지
함수에 대해 배포할 컨테이너 이미지를 선택합니다.

기본 정보

함수 이름

함수의 용도를 설명하는 이름을 입력합니다.

DASTScan

함수 이름은 1~64자여야 하고, 리전에 고유해야 하며, 공백을 포함할 수 없습니다. 유효한 문자는 a~z, A~Z, 0~9, 하이픈(-), 밑줄(_)입니다.

レン타임 정보

Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Python 3.9

아키텍처 정보

함수 코드에 대해 원하는 명령 세트 아키텍처를 선택합니다.

x86_64

arm64

함수 DASTScan이(가) 생성되었습니다. 이제 함수의 코드 및 구성을 변경할 수 있습니다. 테스트 이벤트를 사용하여 함수를 호출하려면 [테스트]를 선택하십시오.



DASTScan

조절

ARN 복사

작업 ▾

권한 정보

기본적으로 Lambda는 Amazon CloudWatch Logs에 로그를 업로드하-

▶ 기본 실행 역할 변경

다이어그램

템플릿



+ 트리거 추가

+ 대상 추가

설명

-

마지막 수정
4분 전

함수 ARN

arn:aws:lambda:ap-northeast-2:711387094022:function:DASTScan

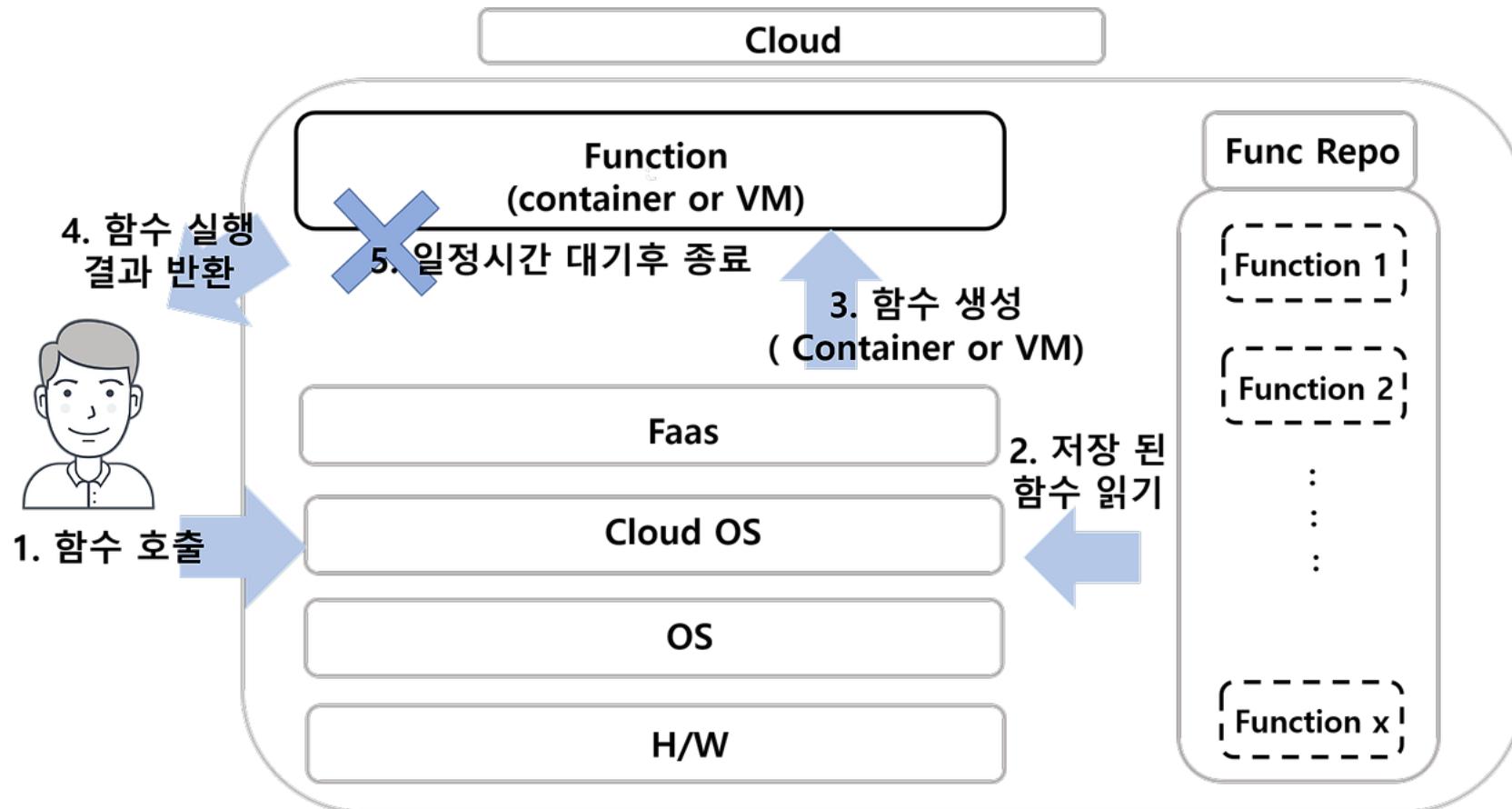
함수 URL

정보

DAST 파이프라인 통합

Lambda 동작 과정

- Lambda는 대표적인 FaaS(Function as a Service) 프로젝트로, 이벤트가 발생 시 실행되는 서비스 컴퓨팅



DAST 파이프라인 통합

ECS Task 호출 후

```
#ECS Task 실행 함수
def trigger_ecs_task():
    response = ecs_client.run_task(
        cluster='juice-shop-cluster',
        taskDefinition='dast-zap-task',
        count=2,
        capacityProviderStrategy=[{'capacityProvider': 'FARGATE_SPOT', 'weight': 1}],
        networkConfiguration={
            'awsvpcConfiguration':{
                'subnets':[subnet_id],
                'securityGroups':[security_group_id],
                'assignPublicIp':'DISABLED'
            }
        }
    )
```

DAST 파이프라인 통합

종료 코드가 0(정상 종료)으로 종료되면 성공 반환

```
if exit_code==0:  
    return {"status":"success","message":"Good"}  
else:  
    return {"status":"failure","message":f"exit code:{exit_codes},reasons:{stopped_reasons}"}
```

DAST 파이프라인 통합

CodePipeline에 DAST 스테이지 추가

스테이지 추가

스테이지 이름
DAST
100자 이하

취소 **스테이지 추가**

생성한 Lambda 함수를 선택 ->

작업 편집

작업 이름
작업의 이름을 선택합니다.

100자 이하

작업 공급자

리전

입력 아티팩트
이 작업의 입력 아티팩트를 선택합니다. [자세히 알아보십시오](#)

100자 이하

함수 이름
AWS Lambda 콘솔에서 이미 생성한 함수를 선택합니다. 또는 AWS Lambda 콘솔에서 함수를 생성한 후 이 작업으로 돌아옵니다.

함수 이름에는 공백 없이 문자, 숫자, 하이픈 또는 밑줄만 포함됩니다. 함수 별칭 또는 함수 ARN은 포함되지 않습니다.

사용자 파라미터 - 선택 사항
이 문자열은 AWS Lambda의 핸들러로 전달되는 이벤트 데이터 파라미터에 사용됩니다.

변수 네임스페이스 - 선택 사항
이 작업의 출력 변수에 대해 네임스페이스를 선택합니다. 이 작업에서 생성된 변수를 구성에 사용하려면 네임스페이스를 선택해야 합니다. [자세히 알아보십시오](#)

출력 아티팩트
이 작업의 출력에 사용할 이름을 선택합니다.

100자 이하

취소 **완료**

DAST 파이프라인 통합

dast-owasp-zap 정보

객체 속성 권한 지표 관리 액세스 지점

객체 (2)

객체는 Amazon S3에 저장되어 있는 기본 엔티티입니다. [Amazon S3 인벤토리](#) 를 사용하여 버킷에 있는 모든 객체의 목록을 얻을 수 있습니다. 다른 사용자가 2

Q 접두사로 객체 찾기

□ 이름	▲ 유형	▼ 마지막 수정	▼ 크기
zap_report_2025-02-14_09-22-14.html	html	2025. 2. 14. pm 6:22:16 PM KST	
zap_report_2025-02-14_09-22-15.html	html	2025. 2. 14. pm 6:22:17 PM KST	

스캔이 끝나면 S3 버킷에
HTML 리포트 업로드

ZAP by Checkmarx ZAP Scanning Report

Site: <http://scp-juice-shop.juice-shop.local:3000>

Generated on Fri, 14 Feb 2025 09:22:08

ZAP Version: 2.16.0

ZAP by Checkmarx

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	2
Low	5
Informational	4
False Positives:	0

Summary of Sequences

For each step: result (Pass/Fail) - risk (of highest alert(s) for the step, if any).

Alerts

Name	Risk Level	Number of Instances
Content Security Policy (CSP) Header Not Set	Medium	11
Cross-Domain Misconfiguration	Medium	10
Cross-Domain JavaScript Source File Inclusion	Low	10
Dangerous JS Functions	Low	2
Deprecated Feature Policy Header Set	Low	11
Insufficient Site Isolation Against Spectre Vulnerability	Low	10
Timestamp Disclosure - Unix	Low	1
Information Disclosure - Suspicious Comments	Informational	2
Modern Web Application	Informational	11
Storable and Cacheable Content	Informational	2

모니터링 기능 및 IaC 구현

모니터링 대상 : ECS 서비스, ALB



AWS CloudWatch
ECS, ALB등의 메트릭 정보 수집



Amazon Managed Grafana
수집된 정보를 시각화



Amazon SNS
문제 발견 시 알람 전송

모니터링 기능 및 IaC 구현

AMG 권한 설정 및 데이터 원본(CloudWatch)과 알림 채널(SNS) 설정

권한 유형 정보

서비스 관리형
다음 단계에서 선택한 AWS 서비스에 따라 권한을 자동으로 프로비저닝합니다.

고객 관리형
제안된 정책에 따라 고유한 IAM 역할을 수동으로 생성합니다.

워크스페이스 구성 옵션 - 선택 사항

워크스페이스 구성 옵션을 활성화합니다. 이 옵션은 기본 구성 옵션으로, 워크스페이스가 생성된 후 더 많이 캔 수 있습니다.

Grafana 알림 커기 정보
Grafana 알림을 켜고 Grafana 워크스페이스에서 Prometheus 알림을 봅니다. Grafana 알림을 켜면 Grafana 알림에 대해 여러 통지가 전송됩니다. Grafana 알림이 꺼진 경우에도 기존 Classic Grafana 알림은 계속 평가됩니다.

플러그인 관리 커기 정보
새로운 Grafana v9+ 작업 공간은 Prometheus 또는 Cloudwatch와 같은 핵심 데이터 소스 플러그인만 함께 제공됩니다. 플러그인 관리를 켜서 작업 공간 관리자에서 플러그인을 검색, 설치, 업데이트 및 제거할 수 있도록 하세요.

네트워크 액세스 제어 - 선택 사항

네트워크 액세스 제어를 사용하면 Amazon Grafana 도메인에 도달할 때 수락할 트래픽 소스를 설정할 수 있습니다.

오픈 액세스
워크스페이스 URL에 공개적으로 연결할 수 있습니다.

제한된 액세스
구성된 리소스만 워크스페이스에 액세스할 수 있습니다.

① 워크스페이스 URL에 공개적으로 연결할 수 있습니다. 네트워크 액세스 제어를 구성하지 않았습니다.

취소 이전 **다음**

데이터 원본

아래에서 AWS 데이터 원본을 선택하면 Amazon Grafana가 현재 계정의 해당 리소스에 액세스할 수 있는 IAM 역할이 생성됩니다. 선택한 서비스 데이터 원본으로 설정되지 않습니다. 일부 리소스의 경우, 액세스할 수 있으려면 GrafanaDataSource를 태그 지정해야 합니다.

데이터 원본 이름

AWS IoT SiteWise

AWS X-Ray

Amazon CloudWatch

Amazon OpenSearch Service

Amazon Managed Service for Prometheus

Amazon TimeStream

Amazon Redshift

Amazon Athena

알림 채널

아래에서 알림 채널을 선택하면 Amazon Grafana가 현재 계정의 해당 리소스에 액세스할 수 있는 IAM 역할이 생성됩니다. 선택한 서비스는 알림 채널로 설정되지 않습니다. "Amazon SNS"를 선택할 때는 "grafana"로 시작하는 SNS 주제만 액세스할 수 있습니다.

알림 채널 이름

Amazon SNS

모니터링 기능 및 IaC 구현

AMG 대시보드로 이동하여 로그인 후

Welcome to Amazon Managed Grafana

Need help? Documentation Tutorials Community

Dashboards

Starred dashboards

Recently viewed dashboards

Latest from the blog

2월 14 Introducing Learning journeys: New step-by-step guides to get started with Grafana Our Big Tent philosophy provides the foundation for our broad, modular, and flexible observability platform. With Grafana's powerful ability to integrate with a wide range of data sources, tools, and plugins, you can create customized solutions tailored to your unique needs.

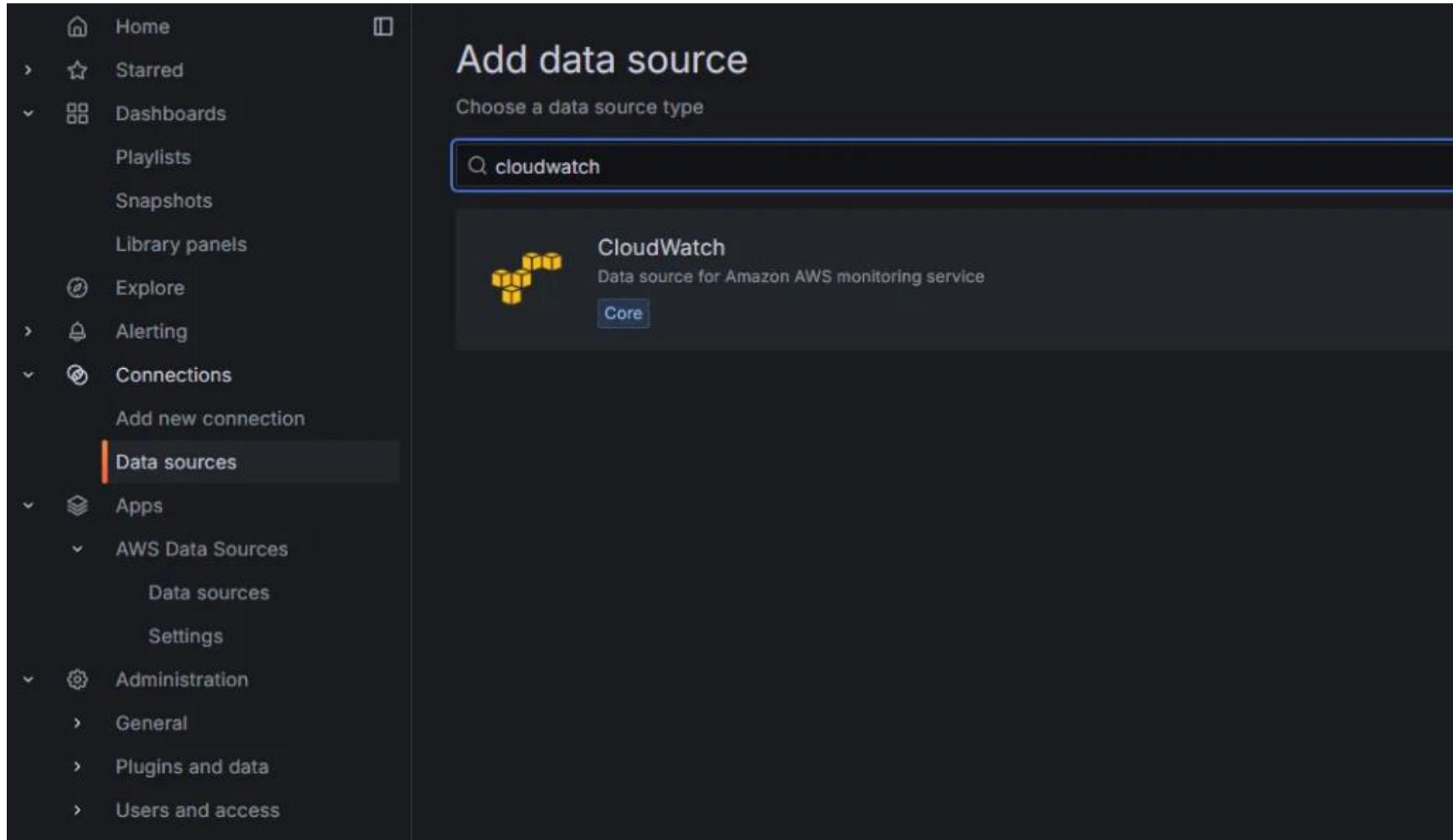
2월 13 Grafana Loki 3.4: Standardized storage config, sizing guidance, and Promtail merging into Alloy The Grafana Loki 3.4 release is here, and it brings a fresh wave of enhancements aimed at standardizing Loki's object storage, helping you right size your instance, and improving the ability to ingest out-of-order logs.

2월 12 How to cut costs for metrics and logs: a guide to lowering expenses in Grafana Cloud Observability is essential to maintaining system reliability, but as your infrastructure scales, so do your costs. Between metrics and logs, managing telemetry data can become overwhelming and expensive.

2월 11 Monitor Google Cloud: simplify and centralize your cloud provider observability with Grafana Cloud Organizations increasingly rely on Google Cloud to power critical parts of their businesses, but managing those environments often involves navigating a labyrinth of disparate data, tools, and processes.

모니터링 기능 및 IaC 구현

AMG Data source를 CloudWatch로 선택



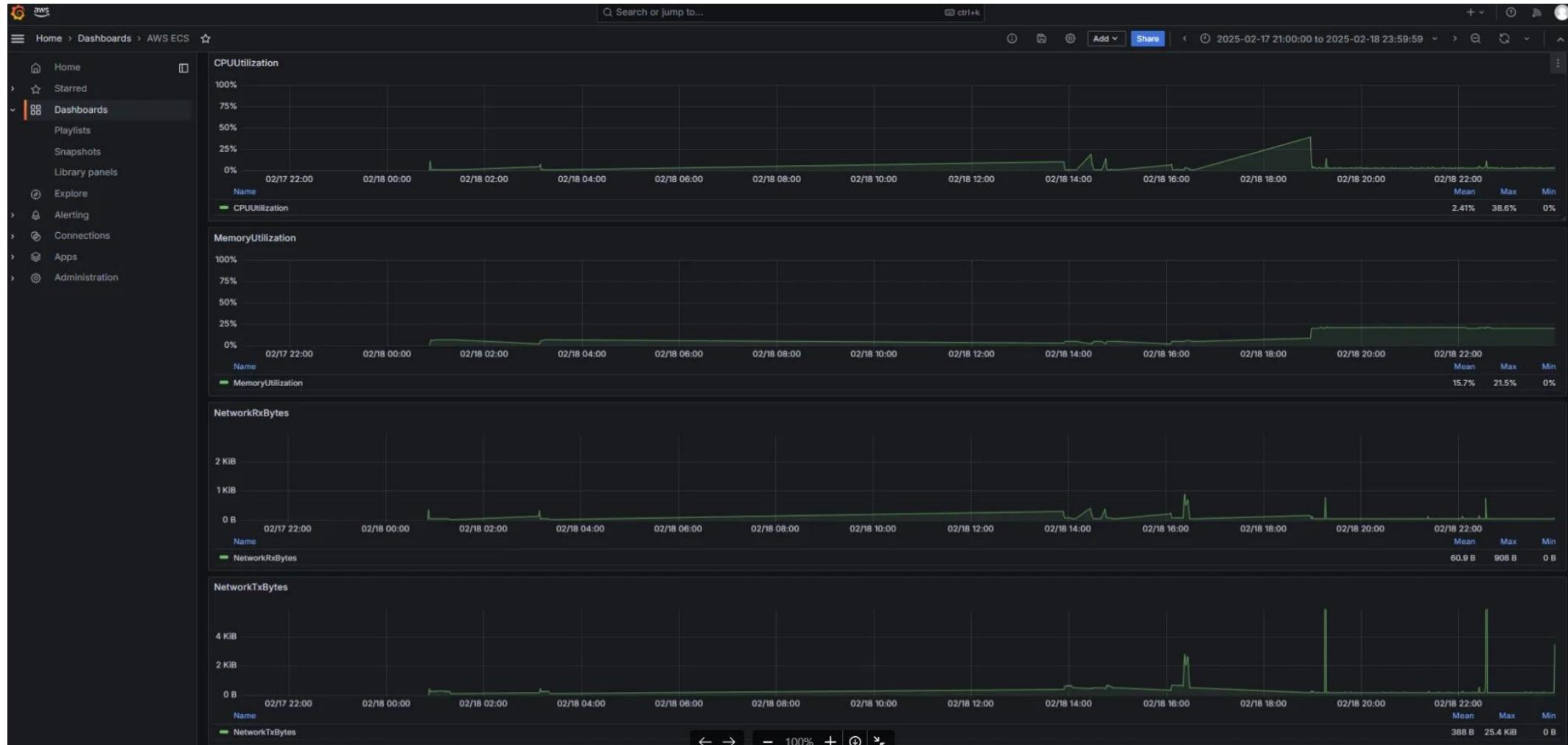
모니터링 기능 및 IaC 구현

AMG 대시보드 설정

The screenshot shows the AWS Metrics Explorer interface. At the top, it displays "Data source: cloudwatch", "Query options: MD = auto = 904 Interval = 10m", and a "Query inspector" button. Below this, there's a search bar with the prefix "A (cloudwatch)". The main area has tabs for "Region: ap-northeast-2", "CloudWatch Metrics", and "Metric Search". On the right, there are buttons for "Run queries", "Builder", and "Code". The "Builder" tab is selected. The configuration section includes fields for "Namespace: ECS/ContainerInsights", "Metric name: CpuUtilized", and "Statistic: Maximum". Under "Dimensions", there are two dropdowns: "ClusterName: juice-shop-cluster" and "ServiceName: scp-juice-shop", both with an equals sign between them. A toggle switch is turned on next to the text "Match exact - optional". At the bottom, there are fields for "ID - optional" (empty), "Period: auto", and "Label - optional" (empty).

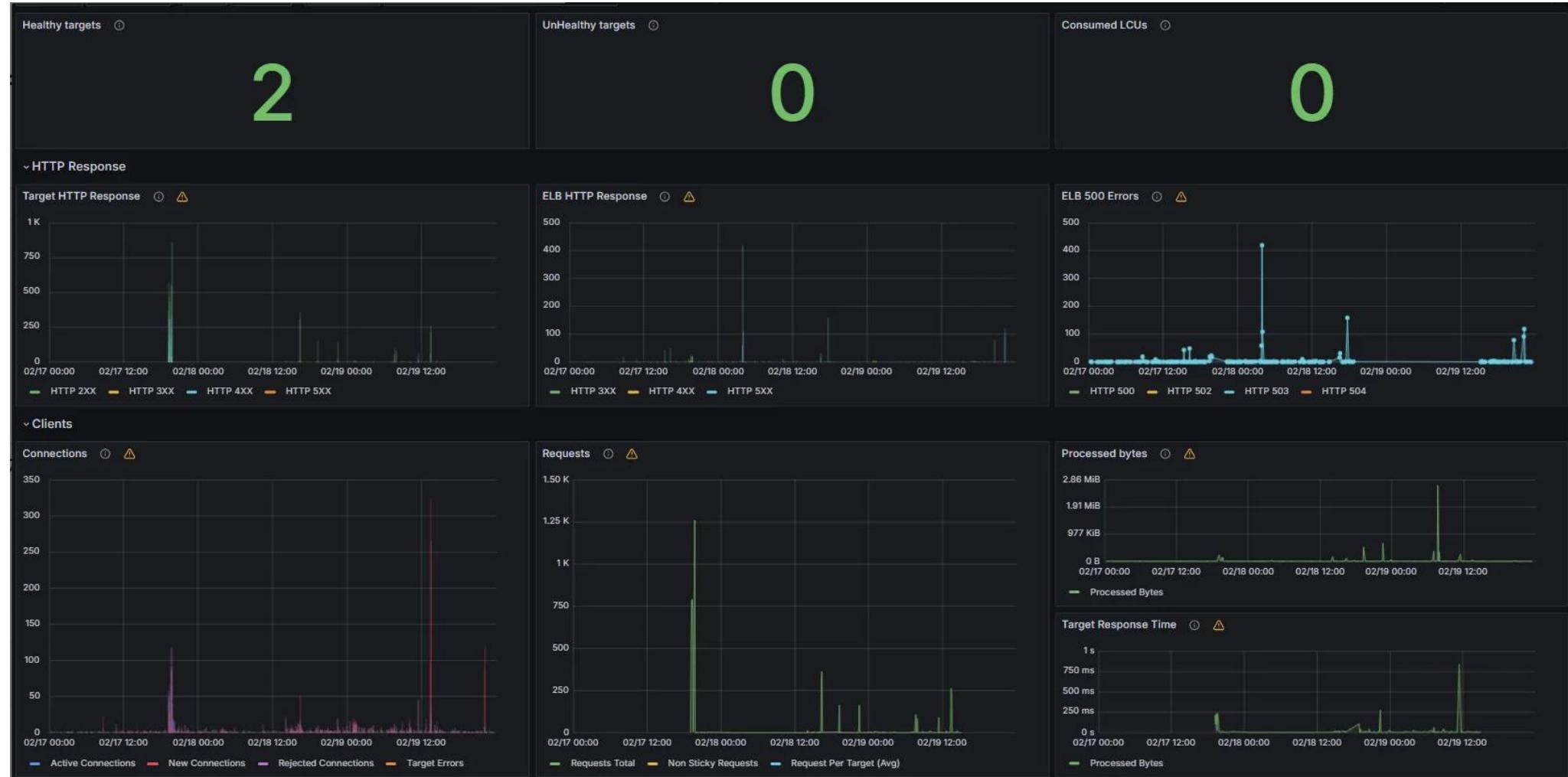
모니터링 기능 및 IaC 구현

시스템 대시보드 (CPU, 메모리, 네트워크 데이터 등) => 커스텀 or 템플릿



모니터링 기능 및 IaC 구현

ALB 대시보드 (Health 정상 태스크, 처리한 데이터 등)

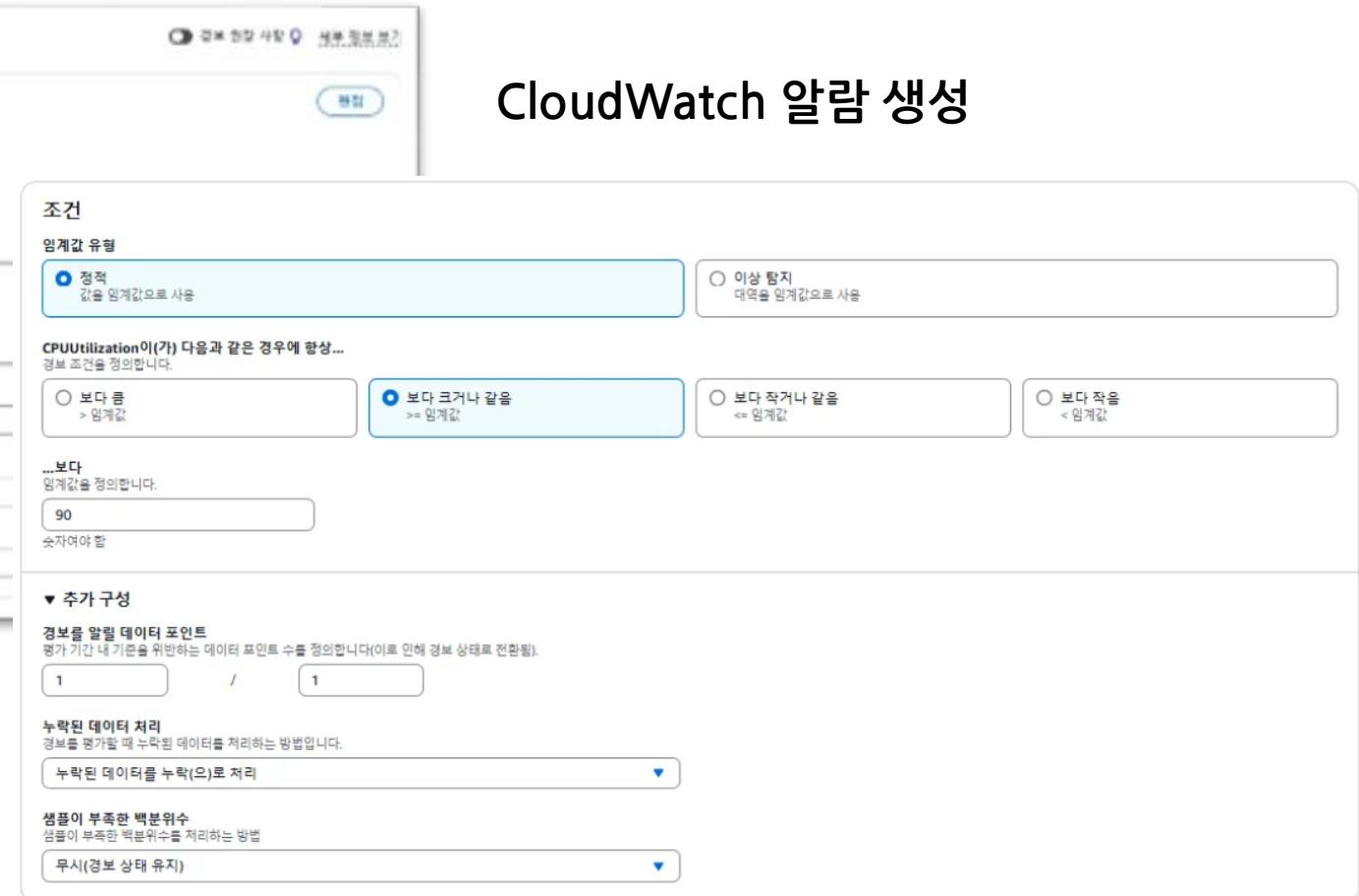


모니터링 기능 및 IaC 구현

CloudWatch → Amazon SNS 알림



CPU 사용량 90% 이상 사용 시 경보



모니터링 기능 및 IaC 구현

CloudWatch -> Amazon SNS 알림

test 받은편지함 x

 AWS Notifications <no-reply@sns.amazonaws.com> 오후 1:42 (0)

나에게 ▾

Alarm Test

--

If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
<https://sns.ap-northeast-2.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:ap-northeast-2:711387094022:CPUUtilization-Alarm:a0f42f30-c000-43ac-93b3-c35dbd4d5691&Endpoint=wornjss1002@gmail.com>

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at <https://aws.amazon.com/support>

◀ 답장 ▶ 전달 😊

모니터링 기능 및 IaC 구현

IaC(Infrastructure as Code)란?

- 서버나 네트워크 같은 IT 인프라를 사람이 직접 설정하는 대신, 코드로
자동으로 관리하고 구성하는 방식



AWS
CloudFormation



모니터링 기능 및 IaC 구현

	AWS CloudFormation	AWS CDK	Terraform
클라우드 환경	AWS 전용	AWS 전용	멀티 클라우드
코드 방식	선언형 (YAML/JSON)	명령형 (TS/Python)	선언형 (HCL)
상태 관리	AWS 관리	AWS 관리	tfstate 파일로 관리
변경 예측	Change Sets 제공	없음	terraform plan 제공

모니터링 기능 및 IaC 구현

	AWS CloudFormation	AWS CDK	Terraform
클라우드 환경	Yes	Yes	Yes
코드 방식	Template	JavaScript, TypeScript, Python, Go, C#	HCL
상태 관리	Yes	Yes	state 파일로 관리
변경 예측	Change Sets 제공	없음	terraform plan 제공

모니터링 기능 및 IaC 구현

디렉토리 구조

```
terraform/
  └── modules/
    ├── vpc/
    ├── ecs/
    ├── ecr/
    ├── alb/
    ├── sg/
    ├── cloudmap/
    ├── dast/
    ├── codebuild/
    ├── lambda/
    ├── cloudwatch/
    └── codepipeline/
  └── main.tf
  └── outputs.tf
  └── provider.tf
  └── backend.tf
  └── variables.tf
```

Terraform 공식 문서 (코드 참고)

The screenshot shows the HashiCorp Registry interface for the AWS provider. The top navigation bar includes links for 'Browse', 'Publish', 'Sign-in', and a 'Use HCP Terraform for free' button. The main content area is titled 'AWS Provider' and contains the following information:

- AWS DOCUMENTATION**: A sidebar with a 'Filter' search bar and a list of resources under 'aws provider', such as 'Guides', 'Functions', 'ACM (Certificate Manager)', and 'API Gateway'.
- AWS Provider**: The main content area describes the AWS provider, stating it supports many resources and requires configuration with AWS credentials. It mentions 1485 resources and 598 data sources.
- Example Usage**: A code snippet for Terraform 0.13 and later, showing how to declare the AWS provider in the `provider` block:

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 5.0"
    }
  }
}
```

On the right side, there are additional links for 'Multi-language provider docs', 'Terraform', and various documentation sections like 'Authentication and Configuration', 'AWS Configuration Reference', etc.

모니터링 기능 및 IaC 구현

Terraform 코드 작성 예시

```
#ECR 리포지토리 생성
resource "aws_ecr_repository" "juice_shop" {
    name          =var.container_name #리포지토리 이름
    image_tag_mutability ="MUTABLE" #이미지 태그 변경 가능

    image_scanning_configuration {
        scan_on_push=false #스캔 비활성화
    }
}

#dast용 ECR 리포지토리 생성
resource "aws_ecr_repository" "dast" {
    name          ="owasp_zap" #리포지토리 이름
    image_tag_mutability ="MUTABLE" #이미지 태그 변경 가능

    image_scanning_configuration {
        scan_on_push =false #스캔 비활성화
    }
}
```

프로젝트 결과

- 클라우드 및 DevOps 이해
- 파이프라인 구축 및 IaC (시연 영상)
- DevOps 보안 요소
- AWS 비용
- 향후 발전 방향

프로젝트 결과

프로젝트 시작 전 팀에서 결정했던 목표

Cloud, DevOps, CI/CD에 대한 전반적인 이해와 실습

CI/CD 파이프라인 구축과 안정적인 배포 환경 구현

DevOps 환경에서 보안 요소에 대한 인지

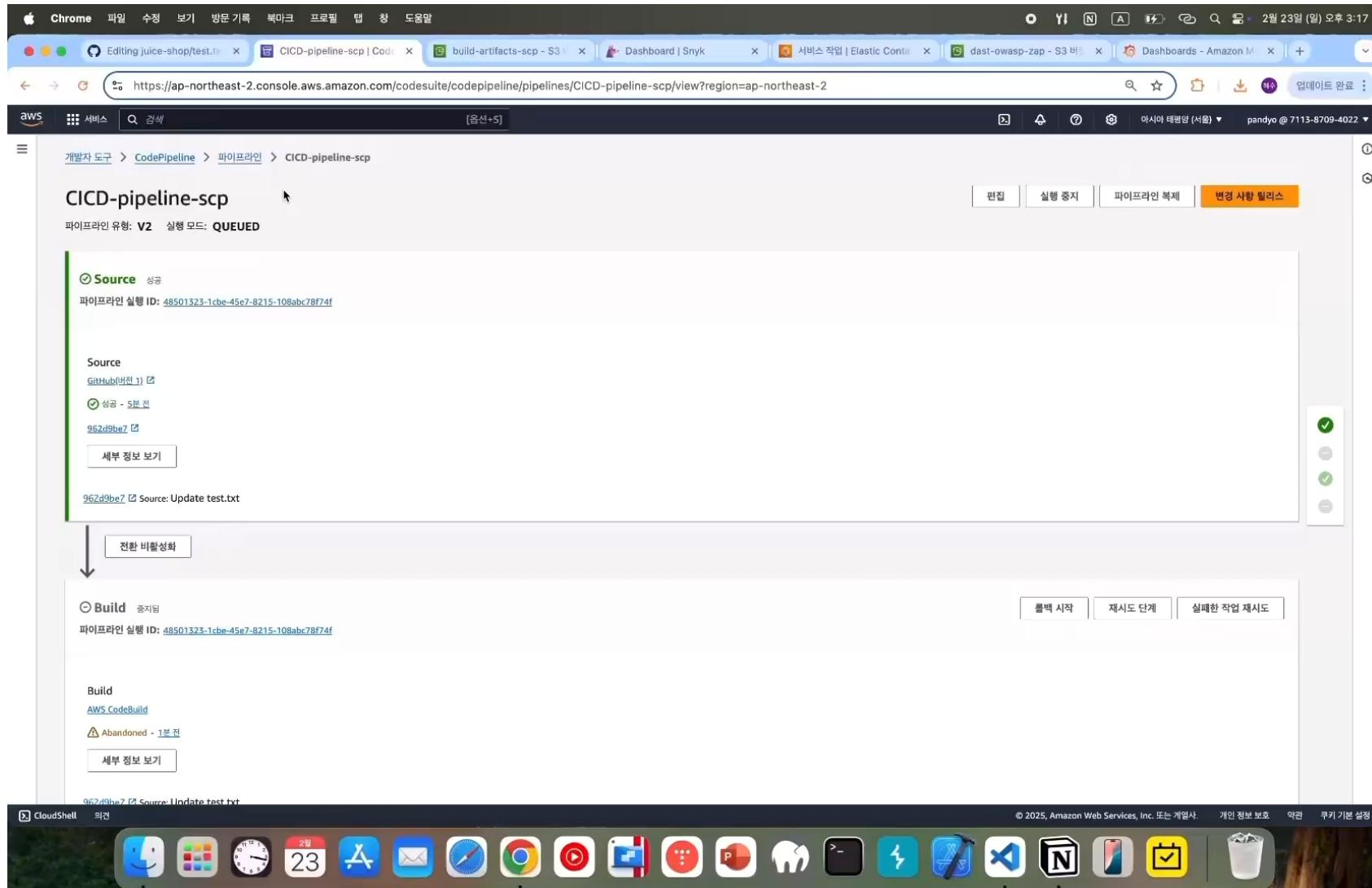
클라우드 및 DevOps 이해

The figure consists of four screenshots of a digital note-taking application, likely Evernote, illustrating various notes taken during the study of DevOps and Cloud services.

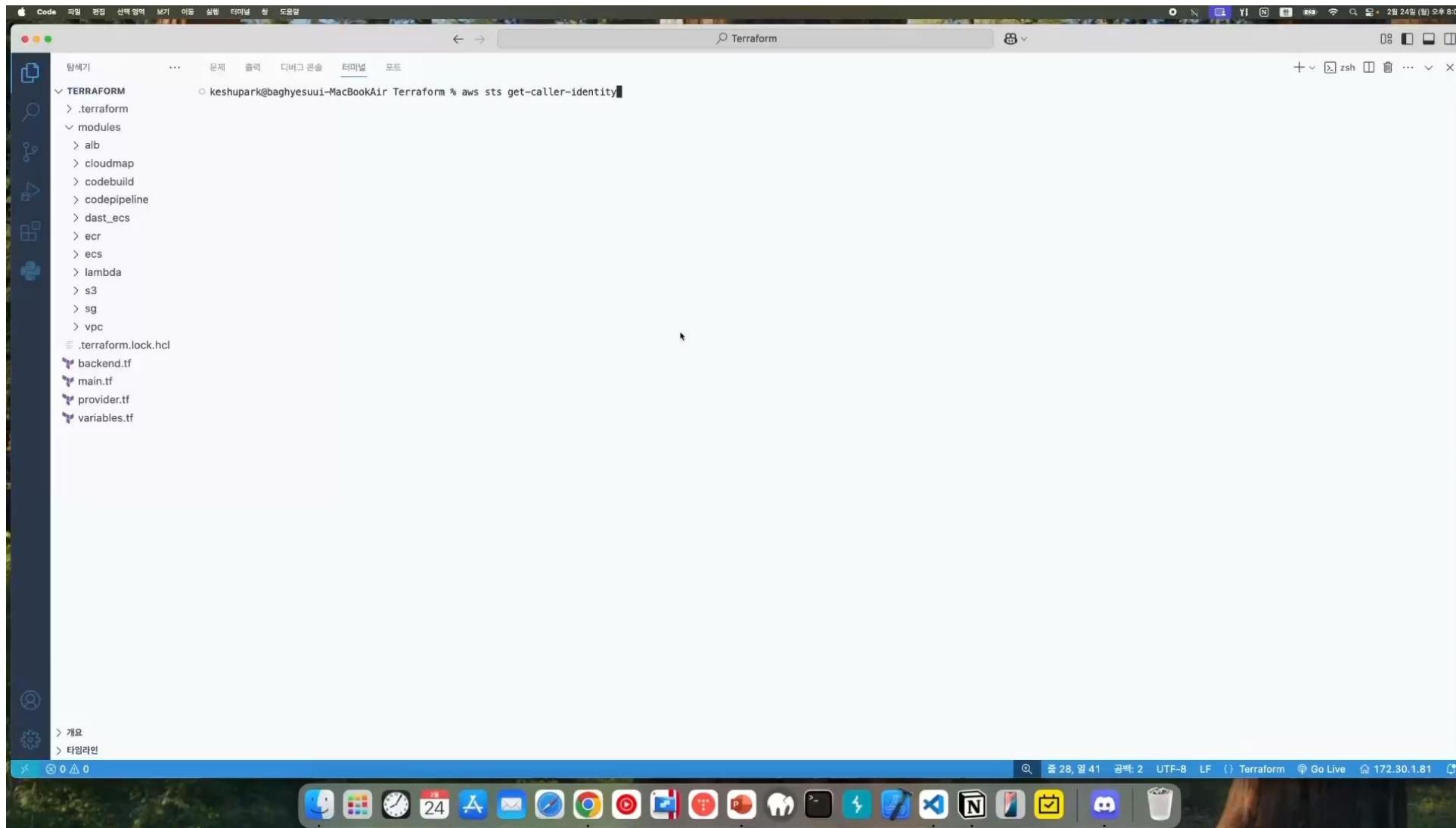
- Screenshot 1:** A note titled "인터넷 검색을 통해 알게 된 자식을 정리한 페이지" containing a list of DevOps-related terms and concepts.
- Screenshot 2:** A note titled "Cloud!" listing various cloud service providers and their specific services.
- Screenshot 3:** A note titled "Cloud!" listing specific cloud services and their details.
- Screenshot 4:** A note titled "Cloud!" listing specific cloud services and their details.

프로젝트를 진행하며
다양한 클라우드
서비스에 대한 이해와
DevOps라는 철학을
구현하는 여러 요소들에
대한 이해를 갖게됨

파이프라인 구축 및 IaC (시연 영상)



파이프라인 구축 및 IaC (시연 영상)



DevOps 보안 요소

접근 제어 및 네트워크



IAM을 통한
사용자 및 서비스
역할 권한 설정



VPC를 통한
안정적인 내부
네트워크 설정



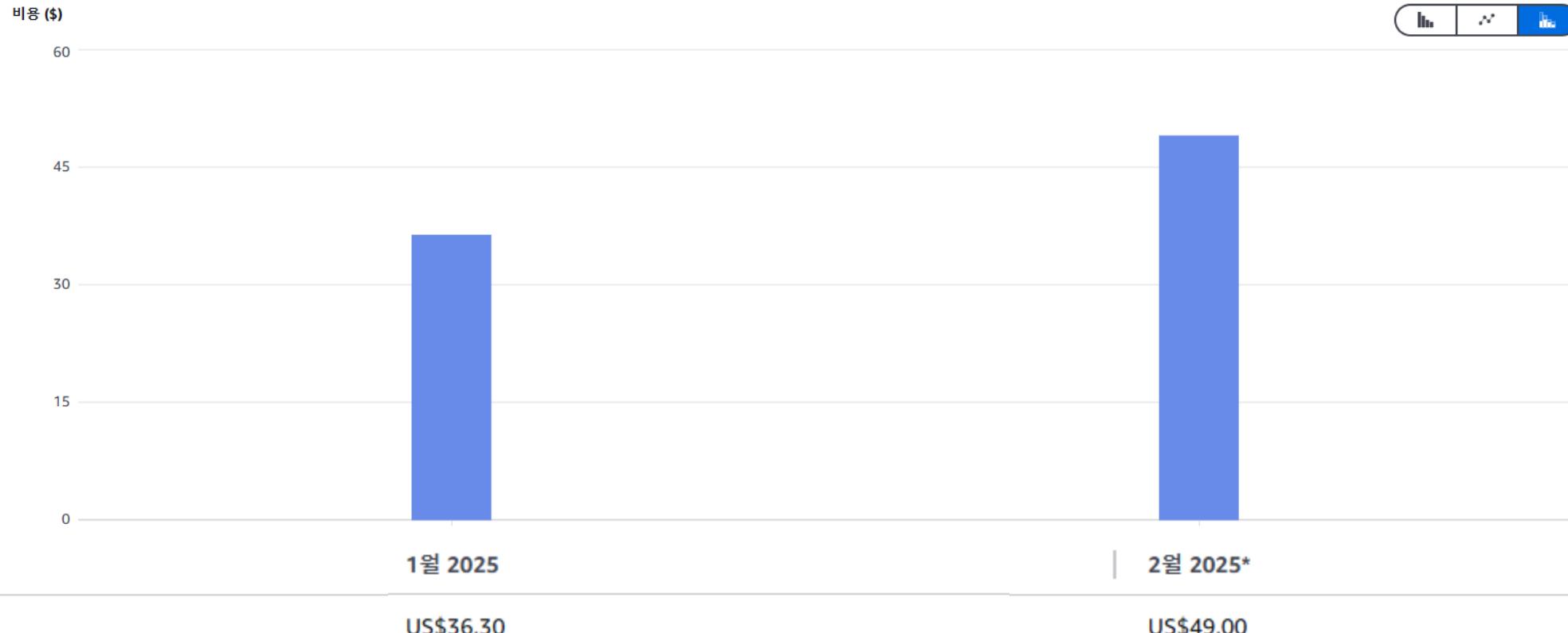
Secrets
Manager를
통한 민감 정보
암호화

AWS 비용

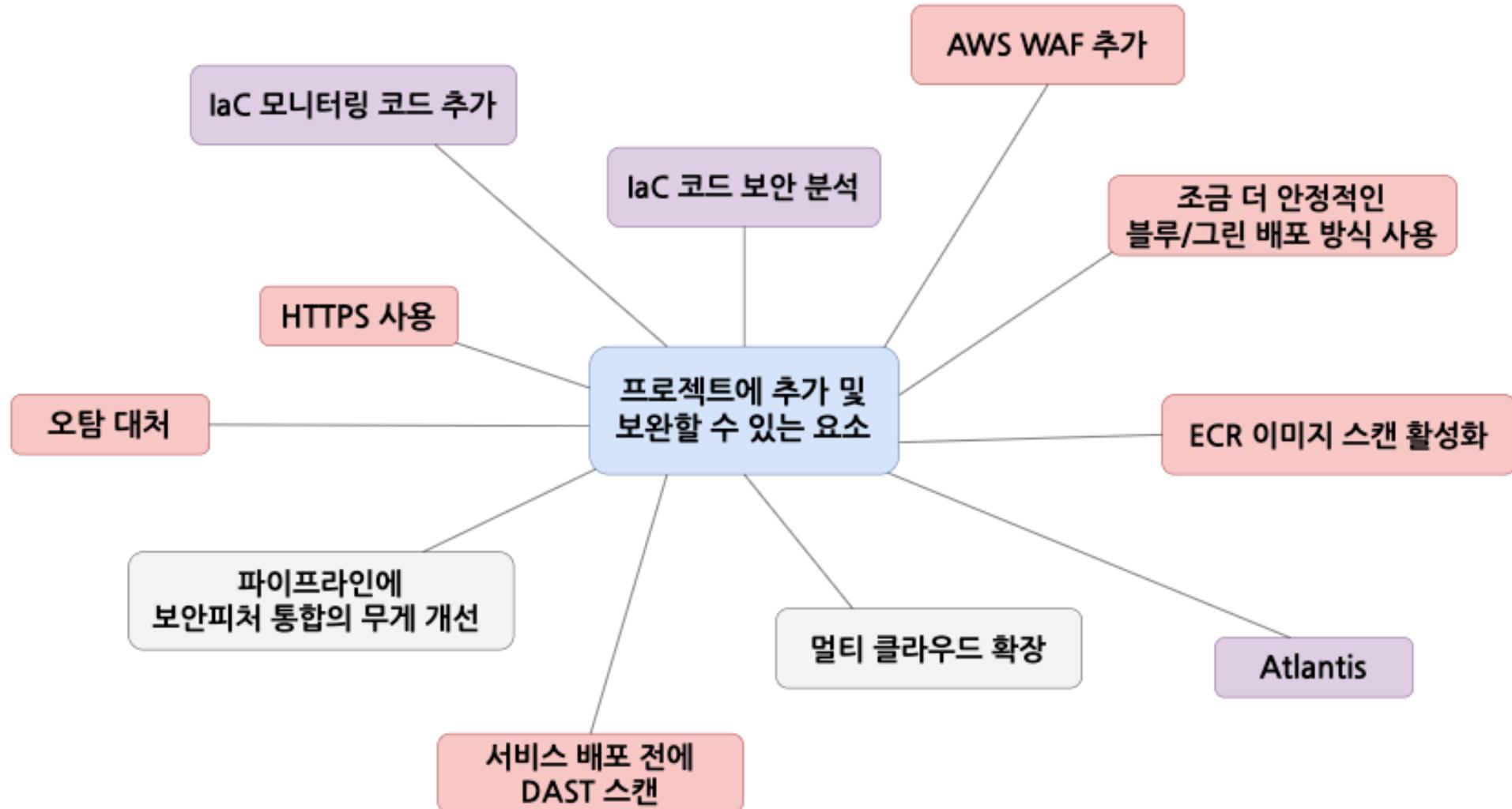
비용 및 사용량 그래프 [정보](#)

총 비용
US\$85.30

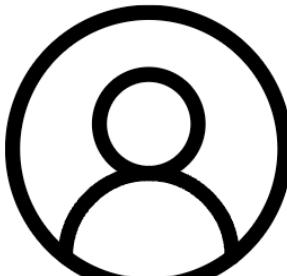
월별 평균 비용
US\$42.65



향후 발전 방향

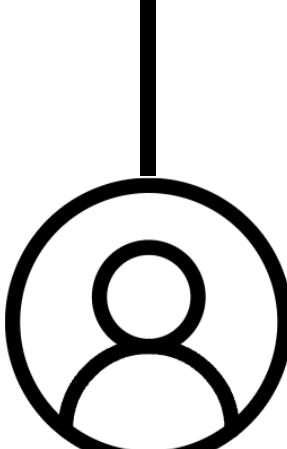


느낀 점



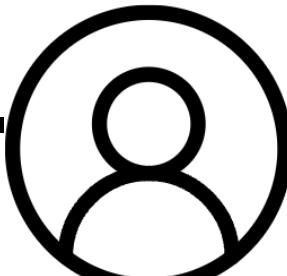
곽민경

좋은 팀원들을 만나서 완성도 높은 프로젝트로 잘 마무리할 수 있었습니다. 프로젝트를 하면서 **클라우드에 대한 지식과 서비스 다루는 방법**들을 익힐 수 있어서, 보안 공부와 클라우드 인프라 환경에 대한 이해가 잘 되었던 것 같습니다.



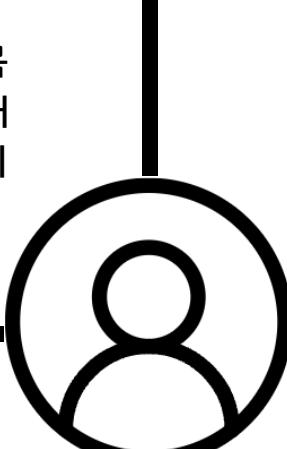
윤주원

AWS 및 관련 서비스들을 사용해 볼 수 있는 좋은 기회였습니다. 또한 한국어로 된 자료가 생각보다 적은 것을 보아 클라우드 분야가 블루오션이라는 생각이 들었습니다. 앞으로 **클라우드 분야의 취업을 목표로 더 심도 있는 공부**를 해나가고 싶습니다.



천재권

클라우드 분야에 기본적인 지식을 학습하고 관련 서비스들을 사용해 보며 좋은 경험을 했습니다. 이를 통해 앞으로 조금 **더 체계적이고 심층적인 DevSecOps 관련 프로젝트**를 진행해 보고 싶다는 생각을 했습니다.



박혜수

처음 진행하는 프로젝트에서 큰 갈등 없이 계획한 목표를 달성할 수 있어서 좋았습니다. 클라우드라는 새로 접하는 분야에 대한 **계획을 잡아 실행하는 과정**이 어려웠지만, 끝나고 보니 **많은 이론과 실습을 통해 성장한 것** 같아 뿌듯했습니다.

감사합니다.

