



아키텍처 다이어그램

Amazon ECS의 SQL 기반 데이터 처리



Amazon ECS의 SQL 기반 데이터 처리: 아키텍처 다이어그램

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

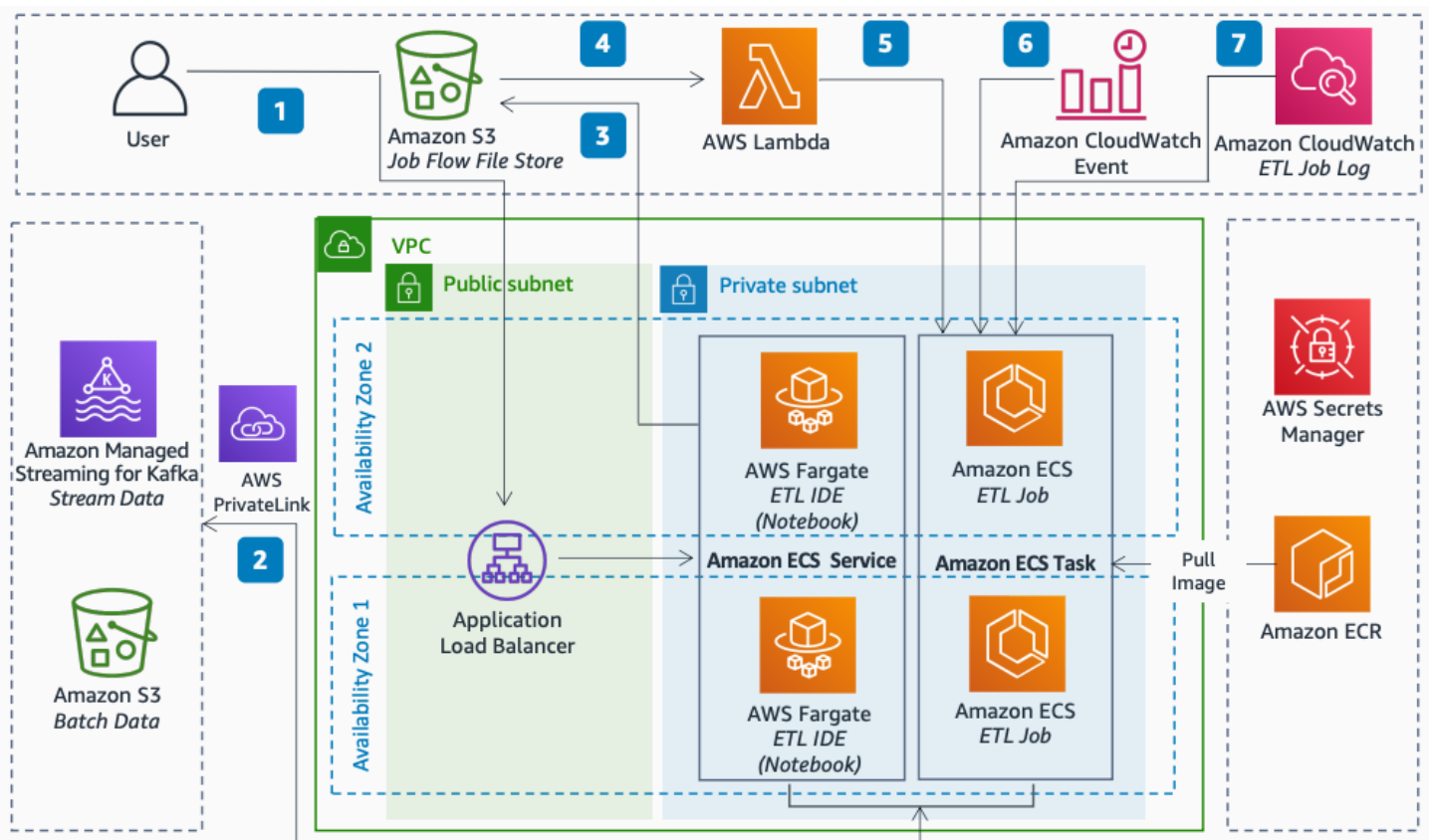
SQL 데이터 처리 ECS	i
Amazon ECS의 SQL 기반 데이터 처리	1
참조 자료	2
다이어그램 기록	2
.....	iii

Amazon ECS의 SQL 기반 데이터 처리

게시일: 2021년 3월 8일([다이어그램 기록](#))

이 아키텍처를 사용하여 데이터 처리를 간소화하고 가속화하는 컨테이너식 ETL 프레임워크([ARC](#))를 사용하여 구성 기반 코드 없는 extract-transform-load(ETL) 대안을 구축하는 방법을 보여줍니다. Apache Spark. [AWS Fargate](#)를 사용하여 [Amazon Elastic Container Service](#)에서 프레임워크를 실행합니다.

Amazon ECS의 SQL 기반 데이터 처리



다음 단계에서는 아키텍처를 설명합니다.

1. 사용자는 대화형 ARC Jupyter Notebook에서 ARC 프레임워크 및 SQL 스크립트를 기반으로 ETL 데이터 파이프라인을 생성합니다. 노트북은 AWS Fargate를 사용하는 Amazon ECS에서 실행됩니다.
2. 노트북 및 ETL 작업은 AWS PrivateLink를 통해 배치 및 스트림 데이터를 처리합니다. ETL 프로세스와 데이터 스토어 간의 트래픽은 Amazon 네트워크를 벗어나지 않습니다.

3. ARC Jupyter 노트북은 작업 흐름 구성 JSON 파일을 생성합니다. 사용자는 CI/CD 자동 배포 프로세스를 통해 또는 수동으로 파일 및 SQL 스크립트를 [Amazon Simple Storage Service](#)에 업로드합니다.
4. Amazon S3 파일 도착 이벤트는 [AWS Lambda](#) 함수를 트리거합니다.
5. Lambda 함수는 Amazon ECS 작업을 실행하여 배치 데이터를 일시적으로 처리하거나 장기 실행 컨테이너에서 스트림 데이터를 지속적으로 처리합니다. 각 작업은 격리된 컴퓨팅 리소스를 사용합니다.
6. [Amazon CloudWatch](#) Events는 AWS Fargate 또는 Amazon EC2 시작 유형을 사용하여 정규 ARC ETL 작업과 Amazon ECS 작업을 예약하고 오케스트레이션합니다.
7. ARC ETL 작업은 세분화된 수준에서 각 데이터 처리 단계에 대한 애플리케이션 로그를 생성합니다. CloudWatch는 모니터링 및 알림 기능을 제공합니다.

참조 자료

자세한 내용은 다음 리소스를 참조하세요.

- [AWS 아키텍처 아이콘](#)
- [AWS 아키텍처 센터](#)
- [AWS Well-Architected](#)

다이어그램 기록

이 참조 아키텍처 다이어그램의 업데이트에 대한 알림을 받으려면 RSS 피드를 구독하세요.

변경 사항	설명	날짜
최초 게시	참조 아키텍처 다이어그램이 처음 게시되었습니다.	2021년 3월 8일

Note

RSS 업데이트를 구독하려면 사용 중인 브라우저에 대해 RSS 플러그인이 활성화되어 있어야 합니다.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.