

비용 최적화 원칙

AWS Well-Architected 프레임워크

2020 년 7 월



고지 사항

고객은 본 문서에 수록된 정보의 정확성을 자체적으로 평가할 책임이 있습니다. 본 문서는 (a) 정보 제공만을 위한 것이며, (b)사전 고지 없이 변경될 수 있는 현재의 AWS 제품 제공 서비스 및 사례를 보여 주며, (c) AWS 및 자회사, 공급업체 또는 라이선스 제공자로부터 어떠한 약정 또는 보증도 하지 않습니다. AWS 제품 또는 서비스는 명시적이든 묵시적이든 어떠한 종류의 보증, 진술 또는 조건 없이 "있는 그대로" 제공됩니다. 고객에 대한 AWS의 책임과 법적 책임은 AWS 계약서에 준하며 본 문서는 AWS와 고객 간의 계약에 포함되지 않고 계약을 변경하지도 않습니다.

© 2020 Amazon Web Services, Inc. 또는 자회사. All rights reserved.

목차

소개	1
비용 최적화.....	2
설계 원칙	2
정의.....	3
클라우드 재무 관리 시행	4
역할별 책임자.....	4
재무와 기술의 파트너십.....	5
클라우드 예산 및 예측	7
비용 인식 프로세스.....	7
비용 인식 문화	8
비용 최적화를 통해 제공되는 비즈니스 가치의 정량화.....	9
지출 및 사용량 인식.....	10
거버넌스.....	11
비용 및 사용량 모니터링	15
리소스 폐기.....	18
비용 효율적인 리소스.....	19
서비스를 선택할 때 비용 평가.....	20
올바른 리소스 유형, 크기 및 개수 선택.....	23
최적의 요금 모델 선택	24
데이터 전송 계획.....	30
수요 관리 및 리소스 공급.....	32
수요 관리	33

동적 공급	34
시간 경과에 따른 최적화	36
새로운 서비스 검토 및 구현	37
결론	38
기고자	38
추가 자료	39
문서 개정	39

요약

AWS(Amazon Web Services) [Well-Architected 프레임워크](#)의 비용 최적화 원칙을 중점적으로 다루는 이 백서에서는 AWS 환경 설계, 제공 및 유지 관리에 모범 사례를 적용할 때 참조할 수 있는 지침을 제공합니다.

비용 최적화 워크로드란 모든 리소스를 완전히 활용하고, 가능한 최저 가격으로 결과를 달성하며, 기능 요구 사항을 충족하는 워크로드를 말합니다. 이 백서에서는 조직 내부의 역량 강화, 워크로드 설계, 서비스 선택/구성/운영 및 비용 최적화 기법 적용을 위한 상세한 지침을 제공합니다.

소개

[AWS Well-Architected 프레임워크](#)는 AWS에서 워크로드를 구축할 때 내리는 의사 결정을 이해하는 데 도움이 됩니다. 이 프레임워크는 클라우드에서 안정적이고 안전하며 효율적이고 경제적인 워크로드를 설계하고 운영하기 위한 설계 모범 사례를 제공합니다. 또한 모범 사례를 기준으로 아키텍처를 일관적으로 측정하고 개선 영역을 식별하는 방법을 알아볼 수 있습니다. Well-Architected 워크로드를 갖추면 비즈니스 성공 가능성이 높아집니다.

이 프레임워크에는 다음의 5가지 원칙이 포함됩니다.

- 운영 우수성
- 보안
- 안정성
- 성능 효율성
- 비용 최적화

이 섹션에서는 비용 최적화 원칙에 대해 중점적으로 설명합니다. 구체적으로는 최저 비용으로 비즈니스 결과를 달성하기 위해 서비스와 리소스를 가장 효율적으로 사용하여 워크로드를 설계하는 방법을 다룹니다.

또한 조직 내부에 비용 최적화 원칙의 모범 사례를 적용하는 방법도 알아봅니다. 기존의 온프레미스 솔루션에서는 비용을 최적화하기가 어려울 수 있습니다. 복잡한 구매 프로세스를 진행하면서 향후의 용량 및 비즈니스 요구를 예측해야 하기 때문입니다. 이 섹션의 모범 사례를 채택하면 조직이 다음 목표를 달성하는 데 도움이 됩니다.

- 클라우드 재무 관리 시행
- 지출 및 사용량 인식
- 비용 효율적인 리소스
- 수요 관리 및 리소스 공급

- 시간 경과에 따른 최적화

이 백서는 CTO(최고 기술 책임자), CFO(최고 재무 책임자), 아키텍처 설계자, 개발자, 재무 관리자, 재무 계획자, 비즈니스 분석가 및 운영 팀 구성원과 같은 기술 및 재무 담당자를 위해 작성되었습니다. 구현 세부 정보나 아키텍처 패턴은 제공되지 않지만 적절한 리소스에 대한 참조가 포함되어 있습니다.

비용 최적화

비용 최적화는 워크로드의 수명 주기에 걸쳐 시스템을 개선하고 구체화하는 지속적인 프로세스입니다. 이 백서의 사례는 비용을 최소화하고 조직의 투자 수익률을 극대화하는 동시에 비즈니스 성과를 달성할 수 있도록 하는 비용 인식 워크로드를 구축 및 운영하는 데 도움이 됩니다.

설계 원칙

비용 최적화를 위해 고려할 설계 원칙은 다음과 같습니다.

클라우드 재무 관리 구현: 클라우드에서 금전적 성공을 달성하고 비즈니스 가치 실현을 앞당기려면 클라우드 재무 관리에 투자해야 합니다. 조직에서는 이 새로운 분야의 기술 및 사용량 관리를 위한 기능을 구축하는 데 필요한 시간과 리소스를 할애해야 합니다. 보안 또는 운영 역량과 마찬가지로, 비용 효율적인 조직이 되는 데 도움이 되는 지식 강화, 프로그램, 리소스 및 프로세스를 통해 역량을 구축해야 합니다.

소비 모델 도입: 사용하는 컴퓨팅 리소스에만 비용을 지불하고 비즈니스 요구 사항에 따라 사용량을 늘리거나 줄입니다. 예를 들어 개발 및 테스트 환경은 주로 주중 근무일에 하루 8시간 동안만 사용됩니다. 사용되지 않는 동안 이러한 리소스를 중단하면 75%의 비용을 절감할 수 있습니다(기존의 168 시간에 비해 40 시간으로 단축).

전반적인 효율성 측정: 워크로드의 비즈니스 결과와 워크로드 제공과 관련된 비용을 측정합니다. 이 데이터를 사용하면 개선된 결과, 향상된 기능 및 절감한 비용을 파악할 수 있습니다.

확실적인 업무 부담에 대한 비용 지출 중단: 랙 및 스택 설치와 서버 전원 공급 등 데이터 센터 운영의 힘든 작업을 AWS가 처리합니다. 또한 관리형 서비스를 통해 운영 체제 및 애플리케이션을 관리하는 운영 부담을 덜어줍니다. 따라서 IT 인프라 대신 고객과 비즈니스 프로젝트에 집중할 수 있습니다.

지출 분석 및 귀속: 클라우드에서는 워크로드의 비용 및 사용량을 정확하게 식별할 수 있으므로 IT 비용의 원인이 되는 수익 흐름 및 개별 워크로드 소유자를 투명하게 파악할 수 있습니다. 이는 ROI(투자 대비 수익률)를 측정하고 워크로드 소유자가 리소스를 최적화하고 비용을 절감하는 기회를 얻을 수 있도록 해 줍니다.

정의

클라우드의 비용 최적화에는 5 가지 중심 영역이 있습니다.

- 클라우드 재무 관리 시행
- 지출 및 사용량 인식
- 비용 효율적인 리소스
- 수요 관리 및 리소스 공급
- 시간 경과에 따른 최적화

Well-Architected 프레임워크의 다른 원칙과 마찬가지로 비용 최적화에도 트레이드-오프가 필요합니다. 예를 들어 최적화의 기준을 출시 시간에 들지, 비용에 들지 결정하는 것입니다. 경우에 따라서는 선결제 비용 최적화에 투자하는 것보다 출시 시간을 단축하거나, 새로운 기능을 배포하거나, 단순히 납기를 준수하는 등 속도를 기준으로 최적화하는 것이 가장 좋습니다.

데이터를 확인하지 않고 시간에 쫓겨 설계 의사 결정을 내리는 경우도 있으며, 가장 비용 최적화된 배포에 대한 벤치마킹에 시간을 들이는 대신 과잉 보상을 선택하기도 합니다. 과잉 보상은 최적화되지 않은 배포를 필요 이상으로 많이 프로비저닝하는 결과로 이어질 수 있습니다. 그러나 온프레미스 환경의 리소스를 클라우드로 “온전히 이전”한 후 최적화해야 하는 경우에는 이 선택이 적합할 수 있습니다.

사전에 비용 최적화 전략에 적절한 노력을 들여 모범 사례를 일관적으로 준수하고 불필요한 오버프로비저닝을 방지하면 클라우드의 경제적 이점을 더 빨리 실현할 수 있습니다. 다음 섹션에서는 클라우드 재무 관리의 초기 및 지속적인 구현과 워크로드의 비용 최적화를 위한 기술과 모범 사례를 제공합니다.

클라우드 재무 관리 시행

CFM(클라우드 재무 관리)을 시행하면 비용 및 사용량을 최적화하고 AWS 에서 규모를 확장하면서 비즈니스 가치와 금전적 성공을 실현할 수 있습니다.

다음은 클라우드 재무 관리 모범 사례입니다.

- 역할별 책임자
- 재무와 기술의 파트너십
- 클라우드 예산 및 예측
- 비용 인식 프로세스
- 비용 인식 문화
- 비용 최적화를 통해 제공되는 비즈니스 가치의 수량화

역할별 책임자

비용 최적화 역할 설정: 이 역할은 비용 인식 문화를 확립하고 유지하는 업무를 담당합니다. 기존의 개인, 조직 내부의 팀 또는 조직 전반의 핵심 재무, 기술 및 조직 이해관계자로 구성된 신규 팀이 이 역할을 맡을 수 있습니다.

이 역할(개인 또는 팀)은 업무 시간의 필요한 비율을 비용 관리 및 비용 최적화 활동에 우선적으로 할애합니다. 소규모 조직의 경우 이 역할이 소비하는 시간의 비율은 대기업의 정규직 역할보다 작을 수 있습니다.

이 역할은 프로젝트 관리, 데이터 과학, 재무 분석 및 소프트웨어/인프라 개발 기능을 사용하여 다방면에서 접근해야 합니다. 또한, 비용 최적화를 실행(중앙 집중식 접근 방식)하거나, 기술 팀에 영향력을 행사하여 최적화를 실행(분산 접근 방식)하도록 하거나, 이 두 방법을 조합(하이브리드)하여 워크로드의 효율성을 개선할 수 있습니다. 이 역할의 성과는 비용 최적화 목표를 실행하고 제공할 수 있는 능력을 기준으로 측정될 수 있습니다(예: 워크로드 효율성 지표).

조직에서는 이 역할에 대한 경영진의 후원을 확보해야 합니다. 스폰서는 비용 효율적인 클라우드 사용의 옹호자로 간주되며 이 역할에 대한 에스컬레이션 지원을 제공하여 비용 최적화 활동이 조직이 정의한 우선 순위 수준에 따라 처리될 수 있도록 합니다. 스폰서와 역할은 함께 조직에서 클라우드가 효율적으로 사용되는지 확인하고 지속적으로 비즈니스 가치를 전달합니다.

재무와 기술의 파트너십

재무와 기술 간의 파트너십 수립: 기술 팀이 클라우드에서 더 빠른 혁신을 달성할 수 있는 이유는 승인, 구매 및 인프라 배포 주기가 짧기 때문입니다. 이는 이전에 데이터 센터와 온프레미스 환경에서 자본을 조달 및 배포하고 프로젝트 승인 시에만 비용을 할당하기 위해 시간 소모적이고 리소스 집약적인 프로세스를 실행했던 재무 조직에 있어서 하나의 조정이 될 수 있습니다.

주요 재무 관계자와 기술 관계자 간의 파트너십을 수립하면 조직의 목표를 함께 이해하고 클라우드 컴퓨팅의 가변 지출 모델에서 재정적 성공을 지원하는 메커니즘을 개발할 수 있습니다. 다음을 포함한 조직 내부의 관련 팀이 클라우드 여정의 모든 단계에서 비용 및 사용 논의에 참여해야 합니다.

- 재무 책임자:** CFO, 재무 관리자, 재무 계획자, 비즈니스 분석가, 구매, 소싱 및 지급 계정 부서는 클라우드 모델 소비, 구매 옵션 및 월별 인보이스 프로세스를 이해해야 합니다. 클라우드와 온프레미스 운영 사이에는 근본적인 차이(예: 사용량 변경 속도, 종량과금제, 계층화된 요금, 요금 모델, 세부 결제 및 사용량 정보)가 있으므로 재무 조직은 클라우드 사용이 구매 프로세스, 인센티브 추적, 비용 할당 및 재무제표 등의 비즈니스 측면에 미치는 영향을 이해해야 합니다.

- **기술 책임자:** 기술 책임자(제품 및 애플리케이션 소유자 포함)는 재무 요구 사항(예: 예산 제약)과 비즈니스 요구 사항(예: 서비스 수준 계약)을 알고 있어야 합니다. 그러면 적절한 조직 목표를 달성하기 위한 워크로드를 구현할 수 있습니다.

재무와 기술의 파트너십은 다음과 같은 이점을 제공합니다.

- 재무 팀과 기술 팀이 비용과 사용량을 거의 실시간으로 파악할 수 있습니다.
- 재무 팀과 기술 팀이 클라우드 지출 차이를 처리하기 위한 표준 운영 절차를 설정할 수 있습니다.
- 재무 관계자는 자본을 사용하여 약정 할인을 구매하는 방법(예: 예약 인스턴스 또는 AWS Savings Plans)과 클라우드를 사용하여 조직 성장을 지원하는 방법과 관련하여 전략적 자문 역할을 할 수 있습니다.
- 클라우드에서 기존의 지급 계정 및 구매 프로세스를 함께 사용할 수 있습니다.
- 재무 팀과 기술 팀이 공동으로 향후 AWS 비용 및 사용량을 예측하여 조직 예산을 조정 및 작성할 수 있습니다.
- 공동의 언어를 통해 조직 간 커뮤니케이션과 재무 개념에 대한 일반적인 이해를 개선할 수 있습니다.

조직 내에서 비용 및 사용량 논의에 관여해야 하는 추가 이해관계자는 다음과 같습니다.

- **사업부 책임자:** 사업부 책임자는 사업부와 전체 회사에 지침을 제공할 수 있도록 클라우드 비즈니스 모델을 파악해야 합니다. 확장 및 워크로드 사용량을 예측해야 할 때와 예약 인스턴스 또는 Savings Plans 등의 장기 구매 옵션을 평가할 때는 이러한 클라우드 관련 지식이 반드시 필요합니다.
- **타사:** 조직에서 타사(예: 컨설턴트 또는 도구)를 사용하는 경우 이러한 업체가 재무 목표에 부합하는지 확인해야 하며 참여 모델 및 ROI(투자 수익률)를 통해 이러한 일치를 증명할 수 있습니다. 타사는 일반적으로 관리 대상 워크로드 보고와 분석을 수행하며 설계 대상 워크로드의 비용을 분석합니다.

클라우드 예산 및 예측

클라우드 예산 및 예측 설정: 고객은 효율성, 속도 및 민첩성을 달성하기 위해 클라우드를 사용합니다. 이러한 특성으로 인해 클라우드의 비용 및 사용량은 매우 가변적입니다. 워크로드 효율성이 증가하거나 새 워크로드 및 기능이 배포되는 경우 비용이 증가할 수 있습니다. 또는 더 많은 고객을 지원하기 위해 워크로드를 확장할 경우에도 사용량 및 비용이 증가합니다. 따라서 이러한 가변성을 포함하도록 기존 조직의 예산 편성 프로세스를 수정해야 합니다.

트렌드 기반 알고리즘(기간별 비용을 입력으로 사용)을 사용하거나, 비즈니스 드라이버 기반 알고리즘(예: 신제품 출시 또는 지역 확장)을 사용하거나, 또는 트렌드와 비즈니스 드라이버의 조합을 사용하여 기존의 예산 편성 및 예측 프로세스를 보다 동적인 프로세스로 조정하십시오.

[AWS Cost Explorer](#)를 사용하면 기간별 비용(추세 기반)에 적용된 기계 학습 알고리즘을 기반으로 일 단위(최대 3개월) 또는 월 단위(최대 12개월) 클라우드 비용을 예측할 수 있습니다.

비용 인식 프로세스

조직의 프로세스 안에 비용 인식을 구현: 신규 및 기존 조직 프로세스 안에 비용 인식을 구현해야 합니다. 가능한 경우 기존 프로세스를 재사용하고 수정하는 것이 좋습니다. 이렇게 하면 민첩성과 속도에 대한 영향을 최소화할 수 있습니다. 다음은 워크로드에 비용 인식을 구현하는 데 도움이 되는 권장 사항입니다.

- 변경 관리에 변경의 재정적 영향을 정량화하는 비용 측정이 포함되어 있는지 확인하십시오. 비용 측정을 포함하면 비용 관련 문제를 사전에 해결하고 비용 절감을 강조할 수 있습니다.
- 비용 최적화가 운영 기능의 핵심 구성 요소인지 확인하십시오. 예를 들어 기존 인시던트 관리 프로세스를 활용하여 비용 및 사용량 이상(비용 초과)의 근본 원인을 조사하고 식별할 수 있습니다.
- 자동화 또는 도구를 사용하여 비용 절감 및 비즈니스 가치 실현을 가속화하십시오. 구현 비용을 고려할 때는 ROI 구성 요소를 포함하도록 대화를 구성하여 시간 또는 금전 투자의 당위성을 설명합니다.

- 조직 전반에 비용 인식 교육을 포함하도록 기존 교육 및 개발 프로그램을 확대하십시오. 여기에 지속적인 교육 및 자격증을 포함하는 것이 좋습니다. 이렇게 하면 조직에서 비용 및 사용량을 자체적으로 관리할 수 있는 역량을 갖출 수 있습니다.

비용 및 사용량 최적화에 대한 보고 및 알림: 조직 내의 비용 및 사용량 최적화를 정기적으로 보고해야 합니다. 비용 최적화를 위한 전용 세션을 구현하거나 워크로드에 대한 정기적인 운영 보고 주기에 비용 최적화를 포함할 수 있습니다. [AWS Cost Explorer](#) 는 대시보드 및 보고서를 제공합니다. [AWS 예산 보고서](#) 를 사용하면 구성된 예산을 기준으로 비용 및 사용량의 진행 상태를 추적할 수 있습니다.

[Amazon QuickSight](#) 와 CUR(비용 및 사용량 보고서) 데이터를 함께 사용하면 세분화된 데이터로 사용자 정의 기반의 보고를 제공할 수 있습니다.

비용 및 사용량의 변화에 신속하게 대응할 수 있도록 비용 및 사용량에 대한 알림을 구현하십시오. [AWS 예산](#) 을 사용하면 타겟을 기준으로 알림을 제공할 수 있습니다. 워크로드의 증가 및 감소와 비용 및 사용량에 모두 알림을 구성하는 것이 좋습니다.

비용 및 사용량에 대한 사전 모니터링: 예외 또는 이상이 있을 뿐만 아니라 조직 내에서 선제적으로 비용 및 사용량을 모니터링하는 것이 좋습니다. 사무실이나 업무 환경 전체에 가시성이 높은 대시보드를 구축하면 핵심 인력이 필요한 정보에 접근할 수 있을 뿐 아니라 조직이 비용 최적화에 중점을 두고 있음을 보여줄 수 있습니다. 가시성 있는 대시보드를 사용하면 성공적인 결과를 적극적으로 홍보하고 조직 전체에 이를 구현할 수 있습니다.

비용 인식 문화

비용 인식 문화 조성: 조직 전체에 비용 인식 문화를 조성하는 변화를 주도하거나 이러한 프로그램을 구현하십시오. 소규모로 시작한 후 역량이 커지고 조직의 클라우드 사용이 증가함에 따라 대규모 프로그램을 구현하는 것이 좋습니다.

비용 인식 문화가 조성되면 조직 전체에서 유기적이고 분산된 방식으로 수행되는 모범 사례를 통해 비용 최적화 및 클라우드 비용 관리를 확장할 수 있습니다. 이렇게 하면 엄격한 하향식의 중앙 집중식 접근 방식에 비해 최소한의 노력으로 조직 전체의 역량이 강화됩니다.

문화의 작은 변화는 현재 및 향후 워크로드의 효율성에 큰 영향을 미칠 수 있습니다. 예를 들면 다음과 같습니다.

- 조직 전체 비용 및 사용량의 게임화. 공개적으로 확인 가능한 대시보드 또는 팀 전체의 정규화된 비용 및 사용량을 비교하는 보고서(예: 워크로드당 비용, 트랜잭션당 비용)를 사용하여 게임화할 수 있습니다.
- 비용 효율성 인식. 자발적 또는 임의의 비용 최적화 성과를 공개적으로 또는 비공개적으로 보상하고, 실수를 통한 교훈으로 향후 재발을 방지합니다.
- 워크로드에 대한 조직의 하향식 요구 사항을 생성하여 미리 정의된 예산을 실행합니다.

새로운 서비스 릴리스로 최신 상태 유지: 새로운 AWS 서비스와 기능을 구현하여 워크로드의 비용 효율성을 높일 수 있습니다. [AWS 뉴스 블로그](#), [AWS 비용 관리 블로그](#) 및 [AWS의 새로운 소식](#)을 주기적으로 검토하여 새로운 서비스 및 기능 릴리스에 대한 정보를 확인하십시오.

비용 최적화를 통해 제공되는 비즈니스 가치의 정량화

비용 최적화의 비즈니스 가치 정량화: 비용 최적화의 절감 효과를 보고하는 것에 더해 제공된 추가 가치를 정량화 하는 것이 좋습니다. 비용 최적화의 이점은 일반적으로 각 비즈니스 결과에 대한 비용 절감이라는 측면에서 정량화 됩니다. 예를 들어 저렴한 비용으로 워크로드 결과 수준을 유지할 수 있는 Savings Plans 를 구매하는 경우 온디맨드 Amazon Elastic Compute Cloud(Amazon EC2) 비용 절감을 수량화할 수 있습니다. 유휴 Amazon EC2 인스턴스가 종료되거나 연결되지 않은 Amazon Elastic Block Store(Amazon EBS) 볼륨이 삭제될 때 AWS 지출의 비용 감소를 수량화할 수 있습니다.

비용 최적화의 비즈니스 가치를 수량화하면 조직에 대한 전체적인 이점을 이해할 수 있습니다. 비용 최적화는 필요한 투자이므로 비즈니스 가치를 수량화하면 이해관계자에게 투자 수익률을 설명할 수 있습니다. 비즈니스 가치를 수량화하면 향후 비용 최적화 투자에 대해 이해관계자로부터 더 많은 지지를 얻는 데 도움이 되며, 조직의 비용 최적화 활동에 대한 결과를 측정하는 프레임워크로 사용될 수 있습니다.

그러나 비용 최적화는 비용 절감 또는 회피 이상의 이점을 제공합니다. 효율성 개선 및 비즈니스 가치를 측정하는 추가 데이터를 캡처하는 것이 좋습니다. 개선의 예는 다음과 같습니다.

- **비용 최적화 모범 사례 실행:** 예를 들어 리소스 수명 주기를 관리하면 인프라 및 운영 비용이 절감되며 실험을 위한 시간 및 예상 밖의 예산이 창출됩니다. 이렇게 하면 조직의 민첩성이 개선되고 수익 창출을 위한 새로운 기회가 열립니다.
- **자동화 구현:** 예를 들어 Auto Scaling 은 최소한의 노력으로 탄력성을 보장하고 수동 용량 계획 작업을 제거하여 직원의 생산성을 높여줍니다. 운영 복원력에 대한 자세한 내용은 [Well-Architected 안정성 원칙 백서](#)를 참조하십시오.
- **향후 AWS 비용 예측:** 재무 관계자는 예측을 통해 다른 내부 및 외부 조직의 이해관계자의 기대치를 설정하고 조직의 재무 예측 기능을 개선할 수 있습니다. [AWS Cost Explorer](#)를 사용하여 비용 및 사용량을 예측할 수 있습니다.

리소스

클라우드 지출 예산 책정 및 예측과 관련한 AWS 모범 사례에 대해 자세히 알아보려면 다음 리소스를 참조하십시오.

- [예산 보고서로 예산 지표 보고](#)
- [AWS Cost Explorer를 통해 예측](#)
- [AWS 교육](#)
- [AWS Certification](#)
- [AWS 클라우드 관리 도구 파트너](#)

지출 및 사용량 인식

비용 및 사용량을 효과적으로 관리하고 비용 절감 기회를 식별하려면 조직의 비용 및 동인을 파악하는 것이 중요합니다. 조직에서는 일반적으로 여러 팀이 운영하는 여러 워크로드를

운영합니다. 이러한 팀은 매출원이 각기 다른 개별 조직 단위 소속일 수 있습니다. 워크로드, 개별 조직 또는 제품 책임자에게 리소스 비용을 귀속하는 기능은 효율적인 사용 행동 양식으로 이어지고 낭비되는 요소를 줄여줍니다. 비용 및 사용량을 정확하게 모니터링하면 각 조직 단위와 제품의 수익성 수준을 이해할 수 있으며, 관련 정보를 근거로 하여 조직 내에서 리소스를 할당할 위치를 더 적절하게 결정할 수 있습니다. 비용은 사용량에 따라 변경되므로 비용 변경을 주도하려면 조직의 모든 수준에서 사용량을 인식하는 것이 필수적입니다.

사용량 및 지출을 적절하게 파악하려면 다각적 방식을 사용하는 것이 좋습니다. 팀에서는 데이터를 수집하여 분석한 다음 보고해야 합니다. 고려해야 할 주요 요소는 다음과 같습니다.

- 거버넌스
- 비용 및 사용량 모니터링
- 폐기

거버넌스

클라우드에서 비용을 관리하려면 아래 거버넌스 영역을 통해 사용량을 관리해야 합니다.

조직 정책 개발: 거버넌스를 수행하는 첫 번째 단계는 조직의 요구 사항을 활용하여 클라우드 사용에 대한 정책을 개발하는 것입니다. 이러한 정책은 조직에서 클라우드를 사용하는 방법과 리소스를 관리하는 방법을 정의합니다. 정책에는 리소스 수명 주기 동안의 리소스 생성, 수정, 폐기를 비롯하여 비용 또는 사용량과 관련된 워크로드의 모든 측면이 포함되어야 합니다.

정책은 조직 전체에서 쉽게 이해할 수 있고 효과적으로 구현할 수 있을 만큼 단순해야 합니다. 허용되는 지리적 리전 사용량 또는 리소스가 실행되어야 하는 시간 등 넓은 범위의 상위 수준 정책부터 시작합니다. 그런 다음 다양한 조직 단위 및 워크로드에 대한 정책을 점진적으로 구체화합니다. 일반적인 정책에는 사용할 수 있는 서비스 및 기능(예: 테스트/개발 환경의 저성능 스토리지)과 다른 그룹에서 사용할 수 있는 리소스 유형(예: 개발 계정에서 사용할 수 있는 가장 큰 리소스 크기는 중간)이 포함됩니다.

목표 및 타겟 개발: 조직의 비용과 사용량에 대한 목표와 타겟을 개발합니다. 목표는 예상 결과에 대한 지침과 방향을 조직에 제공합니다. 타겟은 달성해야 할 구체적이고 측정 가능한 결과를 제공합니다. 예를 들어 목표는 비용을 조금(비선형) 늘리면서 플랫폼 사용량을 크게

늘리는 것입니다. 타겟은 플랫폼 사용량을 20% 늘리고 비용 증가를 5% 미만으로 유지하는 것입니다. 워크로드 효율성을 6개월마다 개선하는 것은 일반적인 목표의 또 다른 예입니다. 여기에 수반되는 타겟은 워크로드의 결과당 비용을 6개월마다 5% 줄이는 것이 될 수 있습니다.

클라우드 워크로드의 일반적인 목표는 워크로드 효율성을 높이고 시간이 지남에 따라 워크로드의 비즈니스 결과당 비용을 줄이는 것입니다. 모든 워크로드에 대해 이 목표를 구현하고 6~12개월마다 효율성 5% 증가와 같은 목표를 설정하는 것이 좋습니다. 클라우드에서는 비용 최적화 기능의 구축과 새로운 서비스 및 서비스 기능의 릴리스를 통해 이러한 목표를 달성할 수 있습니다.

계정 구조: AWS에는 마스터(상위, 이전의 지급 계정) 계정과 멤버(하위, 이전의 연결된 계정) 계정이라고 하는 일대다의 상위 및 하위 계정 구조가 존재합니다. 모범 사례는 조직의 규모나 사용량에 관계없이 하나의 멤버 계정이 포함된 하나 이상의 마스터 계정을 보유하는 것입니다. 모든 워크로드 리소스는 멤버 계정 내에만 존재해야 합니다.

필요한 AWS 계정 수는 경우에 따라 다릅니다. 현재 및 향후의 운영 모델과 비용 모델을 평가하여 AWS 계정 구조가 조직의 목표를 반영하는지 확인해야 합니다. 일부 회사에서는 비즈니스 이유로 다음과 같은 여러 AWS 계정을 생성합니다.

- 조직 단위, 비용 센터 또는 특정 워크로드 간에 관리 및/또는 회계 및 결제 작업을 분리해야 하는 경우
- 특정 워크로드별로 AWS Service Limits가 설정되어 있는 경우
- 워크로드와 리소스 간에 격리 및 분리에 대한 요구 사항이 있는 경우

[AWS Organizations](#)에서 [통합 결제](#)를 사용하는 경우 하나 이상의 멤버 계정과 마스터 계정 간의 구조가 생성됩니다. 멤버 계정을 사용하면 비용과 사용량을 그룹으로 분리하고 구별할 수 있습니다. 각 조직 단위(재무, 마케팅, 영업 등) 또는 각 환경 수명 주기(개발, 테스트, 프로덕션 등) 또는 각 워크로드(워크로드 a, b, c)용으로 별도의 멤버 계정을 생성한 다음 통합 결제를 사용해 이러한 연결 계정을 집계하는 방식이 흔히 사용됩니다.

통합 결제에서는 여러 멤버 AWS 계정의 결제를 단일 마스터 계정에 통합할 수 있으며, 각 연결 계정의 활동은 계속 확인할 수 있습니다. 마스터 계정에 비용 및 사용량이 집계되므로 서비스

대량 구매 할인율을 극대화하고 약정 할인(Savings Plans 및 예약 인스턴스)을 최대한 활용하여 가장 높은 할인을 받을 수 있습니다.

[AWS Control Tower](#) 를 사용하면 여러 AWS 계정을 빠르게 설정하고 구성하여 조직의 요구 사항에 부합하는 거버넌스를 시행할 수 있습니다.

조직 그룹 및 역할: 정책을 개발한 후에는 조직 내부 사용자의 논리적 그룹 및 역할을 생성하여 권한을 할당하고 사용량을 제어할 수 있습니다. 하이레벨 인력 그룹부터 시작하여, 일반적으로 조직 단위 및 직무 역할(예: IT 부서의 시스템 관리자 또는 재무 관리자)과 일치합니다. 그룹에는 유사한 작업을 수행하고 유사한 접근 권한이 필요한 사람이 포함됩니다. 역할은 그룹이 수행해야 할 작업을 정의합니다. 예를 들어 IT 의 시스템 관리자는 모든 리소스를 생성할 수 있는 접근 권한이 필요하지만 분석 팀원은 분석 리소스만 생성하면 됩니다.

제어 - 알림: 비용 제어를 구현하기 위한 일반적인 첫 번째 단계는 비용 또는 사용량 이벤트가 정책 범위를 벗어날 때 알림을 설정하는 것입니다. 이렇게 하면 워크로드 또는 새로운 활동에 대한 제한 또는 부정적인 영향 없이 신속하게 조치를 취하고 교정 조치가 필요한지 여부를 확인할 수 있습니다. 워크로드 및 환경 제한을 파악한 후에는 거버넌스를 시행할 수 있습니다. AWS 에서 알림은 AWS 비용, 사용량 및 약정 할인(Savings Plans 및 예약 인스턴스)에 대한 월 예산을 정의할 수 있는 [AWS 예산](#)을 통해 시행됩니다. 집계 비용 수준(예: 모든 비용)에서 예산을 생성하거나 연결 계정, 서비스, 태그 또는 가용 영역과 같은 특정 차원만 포함하는 보다 세분화된 수준에서 예산을 생성할 수 있습니다. 예산에 이메일 알림을 연결할 수도 있습니다. 그러면 현재 또는 예상 비용 또는 사용량이 정의된 백분율 임계값을 초과할 때 해당 알림이 트리거됩니다.

제어 - 적용: 두 번째 단계로, AWS 에서 [AWS Identity and Access Management\(IAM\)](#) 및 [AWS Organizations SCP\(서비스 제어 정책\)](#)를 사용하여 거버넌스 정책을 적용할 수 있습니다. IAM 은 AWS 서비스 및 리소스에 대한 접근을 안전하게 관리할 수 있는 기능을 제공합니다. IAM 을 사용하면 AWS 리소스를 생성 및 관리할 수 있는 사용자, 생성할 수 있는 리소스 유형 및 리소스 생성 위치를 제어할 수 있습니다. 이렇게 하면 필요하지 않은 리소스의 생성이 최소화됩니다. 이전에 생성한 역할 및 그룹을 사용하고 [IAM 정책](#)을 할당하여 올바른 사용을 시행할 수 있습니다. SCP 를 사용하면 조직 내 모든 계정에 대해 사용 가능한 최대 권한을 중앙에서 제어하여 계정이 접근 제어 지침을 준수하는지 확인할 수 있습니다. SCP 는 모든 기능이 활성화된 조직에서만 사용할 수 있으며, 기본적으로 멤버 계정에 대한 작업을

거부하거나 허용하도록 SCP 를 구성할 수 있습니다. 접근 관리 구현에 대한 자세한 내용은 [Well-Architected 보안 원칙 백서](#)를 참조하십시오.

제어 - 서비스 할당량: 서비스 할당량 관리를 통해 거버넌스를 구현할 수도 있습니다. 오버헤드를 최소화하는 방식으로 서비스 할당량을 설정하고 정확하게 유지 관리하면 조직의 요구 사항에 포함되지 않는 리소스 생성을 최소화할 수 있습니다. 이렇게 하려면 요구 사항 변경 속도와 진행 중인 프로젝트(리소스 생성 및 폐기)를 파악하고 할당량 변경을 구현할 수 있는 속도를 고려해야 합니다. 필요한 경우 [서비스 할당량](#)을 사용하여 할당량을 늘릴 수 있습니다.

[AWS 비용 관리 서비스](#)는 AWS Identity and Access Management (IAM) 서비스와 통합됩니다. IAM 서비스와 비용 관리 서비스를 함께 사용하면 결제 콘솔의 AWS 도구 및 재무 데이터 접근을 제어할 수 있습니다.

워크로드 수명 주기 추적: 워크로드의 전체 수명 주기를 추적해야 합니다. 수명 주기를 추적하면 워크로드 또는 워크로드 구성 요소가 더 이상 필요하지 않을 때 이를 폐기하거나 수정할 수 있습니다. 이 기능은 새로운 서비스 또는 기능을 릴리스할 때 특히 유용합니다. 사용 중인 것처럼 보일 수 있는 기존 워크로드 및 구성 요소를 폐기하여 고객을 새 서비스로 리디렉션해야 합니다. 워크로드의 이전 단계를 확인하십시오. 워크로드가 프로덕션 단계로 전환된 후에는 이전 환경을 폐기하거나 다시 필요할 때까지 용량을 크게 줄일 수 있습니다.

AWS 는 엔터티 수명 주기 추적에 사용할 수 있는 다양한 관리 및 거버넌스 서비스를 제공합니다. [AWS Config](#) 또는 [AWS Systems Manager](#) 를 사용하여 AWS 리소스 및 구성에 대한 상세한 인벤토리를 제공할 수 있습니다. 추적 기능을 기존 프로젝트 또는 자산 관리 시스템에 통합하여 조직 내의 진행 중인 프로젝트와 제품을 추적하는 것이 좋습니다. AWS 에서 제공하는 다양한 이벤트 및 지표 집합과 현재 시스템을 통합하면 중요한 수명 주기 이벤트 뷰(view)를 작성하고 리소스를 사전에 관리하여 불필요한 비용을 줄일 수 있습니다.

엔터티 수명 주기 추적에 대한 자세한 내용은 [Well-Architected 운영 우수성 원칙 백서](#)를 참조하십시오.

비용 및 사용량 모니터링

워크로드를 자세히 확인할 수 있게 되면 각 팀에서 비용 및 사용량에 대해 조치를 취할 수 있습니다. 비용 최적화는 비용 및 사용량 내역을 세부적으로 이해하고, 향후 지출, 사용량 및 기능을 모델링하고 예측하며, 비용 및 사용량을 조직의 목표에 맞게 조정할 수 있는 충분한 메커니즘을 구현하는 것에서 시작됩니다. 다음은 비용 및 사용량의 모니터링에 필요한 영역입니다.

세부적인 데이터 원본 구성: Cost Explorer 에서 시간 단위 세분화를 활성화하고 [CUR\(비용 및 사용량 보고서\)](#)을 생성합니다. 이러한 데이터 원본은 전체 조직의 비용 및 사용량을 가장 정확하게 보여줍니다. CUR 은 모든 청구 가능한 AWS 서비스에 대해 일 또는 시간 단위 사용량 세분화, 요금, 비용 및 사용량 속성을 제공합니다. 태그 지정, 위치, 리소스 속성 및 계정 ID 를 포함하여 가능한 모든 차원을 CUR 에서 사용할 수 있습니다.

다음과 같은 사용자 지정을 사용하여 CUR 을 구성합니다.

- 리소스 ID 포함
- CUR 자동 새로 고침
- 시간 단위 세분화
- 버전 관리: 기존 보고서 덮어쓰기
- 데이터 통합: Athena(Parquet 형식 및 압축)

[AWS Glue](#) 를 사용하여 분석에 사용할 데이터를 준비하고 [Amazon Athena](#) 를 사용하여 데이터 분석을 수행하며 SQL 을 사용하여 데이터를 쿼리할 수 있습니다. [Amazon QuickSight](#) 를 사용하여 사용자 지정된 복합적인 시각화를 작성하고 조직 전체에 배포할 수도 있습니다.

비용 귀속 범주 식별: 재무 팀 및 기타 이해관계자와 협력하여 조직 내의 비용 할당 방식에 관한 요구 사항을 파악합니다. 워크로드 비용은 개발, 테스트, 프로덕션 및 폐기를 포함한 전체 수명 주기에 걸쳐 할당되어야 합니다. 조직에서 학습, 직원 개발 및 아이디어 창출에 대해 발생한 비용의 귀속 방법을 파악합니다. 그러면 이 목적으로 사용되는 계정을 일반적인 IT 비용 예산 대신 교육 및 개발 예산에 올바르게 할당할 수 있습니다.

워크로드 지표 설정: 비즈니스 성공을 기준으로 워크로드의 결과를 측정하는 방법을 파악합니다. 일반적으로 각 워크로드에는 성능을 나타내는 소규모의 주요 결과 세트가 있습니다. 구성 요소가 많은 복잡한 워크로드가 있는 경우 목록의 우선 순위를 지정하거나 각 구성 요소에 대한 지표를 정의하고 추적할 수 있습니다. 팀과 협력하여 사용할 지표를 파악하십시오. 이 단위는 워크로드의 효율성 또는 각 비즈니스 결과의 비용을 파악하는 데 사용됩니다.

비용 및 사용량에 조직의 의미 지정: AWS에서는 [태그 지정](#)을 구현하여 조직 정보를 리소스에 추가할 수 있습니다. 이러한 조직 정보는 다시 비용 및 사용량 정보에 추가됩니다. 태그는 키-값 페어입니다. 키는 정의되며 조직 전체에서 고유해야 하고 값은 리소스 그룹에 고유해야 합니다. 예를 들어 키가 Environment 이고 값은 Production 인 키-값 페어가 있을 수 있습니다. 프로덕션 환경의 모든 리소스에는 이 키-값 페어가 있습니다. 태그 지정을 통해 의미가 있는 관련된 조직 정보를 사용하여 비용을 분류하고 추적할 수 있습니다. 조직 범주(예: 원가 중심점, 애플리케이션 이름, 프로젝트 또는 소유자)를 나타내는 태그를 적용하고 워크로드 및 워크로드의 특성(예: 테스트 또는 프로덕션)을 식별하여 조직 전체의 비용 및 사용량을 해당하는 개체에 귀속할 수 있습니다.

EC2 인스턴스 또는 Amazon S3 버킷과 같은 AWS 리소스에 태그를 적용하고 활성화하면 AWS가 비용 및 사용 보고서에 이 정보를 추가합니다. 태그가 지정된 리소스와 지정되지 않은 리소스에 대해 보고서를 실행하고 분석을 수행하면 내부 비용 관리 정책에 대한 규정 준수를 개선하고 정확한 귀속을 보장할 수 있습니다.

조직 전체 계정에 적용되는 AWS 태그 지정 표준을 생성하고 구현하면 통일성 있는 일관된 방식으로 AWS 환경을 관리하고 제어할 수 있습니다. AWS Organizations에서는 [태그 정책](#)을 사용하여 AWS Organizations의 계정에 포함된 AWS 리소스에 태그를 사용하는 방법에 대한 규칙을 정의할 수 있습니다. 태그 정책을 사용하면 AWS 리소스 태그 지정에 대한 표준화된 접근 방식을 쉽게 도입할 수 있습니다.

[AWS Tag Editor](#)를 사용하면 여러 리소스의 태그를 추가, 삭제 및 관리할 수 있습니다.

[AWS Cost Categories](#)를 사용하면 리소스에 태그를 지정할 필요 없이 조직의 의미를 비용에 지정할 수 있습니다. 비용 및 사용량 정보를 고유한 내부 조직 구조에 매핑할 수도 있습니다. 계정 및 태그와 같은 결제 차원을 사용하여 비용을 매핑하고 분류하는 범주 규칙을 정의하면

태그 지정에 더해 또 다른 수준의 관리 기능을 제공할 수 있습니다. 특정 계정과 태그를 여러 프로젝트에 매핑할 수도 있습니다.

결제 및 비용 최적화 도구 구성: 사용량을 수정하고 비용을 조정하려면 조직의 각 사용자가 비용 및 사용량 정보에 접근 할 수 있어야 합니다. 클라우드를 사용하는 모든 워크로드 및 팀을 위해 다음과 같은 도구를 구성하는 것이 좋습니다.

- **보고서:** 모든 비용 및 사용량 정보를 요약합니다.
- **알림:** 비용 또는 사용량이 정의된 제한을 벗어날 때 알림을 제공합니다.
- **현재 상태:** 현재 비용 및 사용량 수준을 보여주는 대시보드를 구성합니다. 대시보드는 작업 환경 내의 가시성이 높은 위치에서 사용할 수 있어야 합니다(작업 대시보드와 유사함).
- **추세:** 필요한 기간의 비용 및 사용량 변동을 보여주는 기능을 필요한 세부 수준으로 제공합니다.
- **예측:** 향후 예상 비용을 보여주는 기능을 제공합니다.
- **추적:** 구성된 목표 또는 타겟을 기준으로 현재 비용 및 사용량을 보여줍니다.
- **분석:** 팀원이 가능한 모든 차원에서 시간 단위 세부 수준까지 사용자 지정 심층 분석을 수행할 수 있는 기능을 제공합니다.

[AWS Cost Explorer](#), [AWS 예산](#) 및 [Amazon Athena](#) 와 같은 AWS 기본 도구를 [QuickSight](#) 와 함께 사용하여 이 기능을 제공할 수 있습니다. 타사 도구를 사용할 수도 있지만, 해당하는 타사 도구의 비용이 조직에 가치를 제공하는지 확인해야 합니다.

워크로드 지표를 기준으로 비용 할당: 비용 최적화는 최저 가격으로 비즈니스 성과를 제공하는 것입니다. 이를 위해서는 워크로드 지표(워크로드 효율성으로 측정됨)를 기준으로 워크로드 비용을 할당해야 합니다. 로그 파일 또는 기타 애플리케이션 모니터링을 통해 정의된 워크로드 지표를 모니터링하십시오. 이 데이터를 워크로드 비용과 결합합니다. 워크로드 비용은 특정 태그 값 또는 계정 ID 에 연결된 비용을 조회하여 확인할 수 있습니다. 이 분석은 시간 단위 수준으로 수행하는 것이 좋습니다. 정적 비용 구성 요소(예: 상시 실행되는 백엔드 데이터베이스)가 일부 있고 요청 비율이 가변적인 경우(오전 9시부터 오후 5시까지 사용량이

많고 야간에는 요청 수가 적음)에는 효율성이 변경되는 것이 일반적입니다. 정적 비용과 가변 비용 간의 관계를 이해하면 최적화 활동에 집중하는 데 도움이 됩니다.

리소스 폐기

프로젝트, 직원 및 기술 리소스 목록을 장기적으로 관리하면 더 이상 사용되지 않는 리소스와 더 이상 소유자가 없는 프로젝트를 확인할 수 있습니다.

수명 주기에 걸친 리소스 추적: 더 이상 필요하지 않은 워크로드 리소스를 폐기합니다.

일반적으로 테스트에 사용되는 리소스가 그 예입니다. 테스트가 완료된 후에는 리소스를 제거해도 됩니다. 태그를 사용하여 리소스를 추적하고 해당 태그에 대한 보고서를 실행하면 폐기할 자산을 식별하는 데 도움이 됩니다. 태그를 사용하면 리소스에 기능별 레이블을 지정하거나 폐기할 수 있는 알려진 날짜를 지정하여 리소스를 효과적으로 추적할 수 있습니다. 그런 다음 이러한 태그에 대해 보고를 실행할 수 있습니다. 예를 들어 기능 태그를 지정할 때는 워크로드 수명 주기 측면에서 리소스의 목적을 식별하는 "featureX testing"과 같은 값을 지정할 수 있습니다.

폐기 프로세스 구현: 조직 전체에서 미사용 리소스를 식별하고 제거하는 표준화된 프로세스를 구현합니다. 이 프로세스에서는 검색 수행 빈도와 리소스를 제거하여 모든 조직 요구 사항이 충족되는지 확인하는 프로세스를 정의해야 합니다.

리소스 폐기: 사용되지 않는 리소스를 검색하는 빈도 및 작업은 잠재적 절감을 고려하여 결정해야 합니다. 비용이 높은 계정은 비용이 낮은 계정보다 더 자주 분석되어야 합니다. 워크로드의 상태 변경(예: 제품 EOL 또는 교체)을 통해 검색 및 폐기 이벤트를 트리거할 수 있습니다. 검색 및 폐기 이벤트는 시장 상황의 변화나 제품 종료와 같은 외부 이벤트에 의해 트리거될 수도 있습니다.

리소스 자동 폐기: 자동화를 사용하여 폐기 프로세스와 관련된 비용을 줄이거나 제거합니다. 자동 폐기를 수행하도록 워크로드를 설계하면 수명 주기 동안 전체 워크로드 비용을 절감할 수 있습니다. [AWS Auto Scaling](#) 을 사용하여 폐기 프로세스를 수행할 수 있습니다. [API 또는 SDK](#) 를 사용하여 워크로드 리소스를 자동으로 폐기하는 사용자 지정 코드를 구현할 수도 있습니다.

리소스

지출 인식을 위한 AWS 모범 사례에 대해 자세히 알아보려면 다음 리소스를 참조하십시오.

- [AWS 태그 지정 전략](#)
- [사용자 정의 비용 할당 태그 활성화](#)
- [AWS Billing and Cost Management](#)
- [비용 관리 블로그](#)
- [다중 계정 결제 전략](#)
- [AWS SDK 및 도구](#)
- [태그 지정 모범 사례](#)
- [Well-Architected 실습 - 비용 기초](#)
- [Well-Architected 실습 - 비용 인식](#)

비용 효율적인 리소스

비용을 절약하려면 워크로드에 적절한 서비스, 리소스 및 구성을 사용해야 합니다. 비용 효율적인 리소스를 생성할 때는 다음을 고려하십시오.

- 서비스를 선택할 때 비용 평가
- 올바른 리소스 유형, 크기 및 개수 선택
- 최적의 요금 모델 선택
- 데이터 전송 계획

AWS 솔루션 아키텍트, AWS 솔루션, AWS 참조 아키텍처 및 APN 파트너를 활용하면 확인된 정보를 토대로 하여 아키텍처를 선택할 수 있습니다.

서비스를 선택할 때 비용 평가

조직 요구 사항 식별: 워크로드에 사용할 서비스를 선택할 때는 조직의 우선 순위를 이해하는 것이 중요합니다. 비용 및 기타 Well-Architected 원칙(예: 성능 및 안정성) 간에 균형을 유지해야 합니다. 완전한 비용 최적화 워크로드는 조직의 요구 사항과 가장 일치하는 솔루션을 의미하며 꼭 비용이 가장 낮은 솔루션이 아닐 수 있습니다. 조직 내 모든 팀과 회의를 통해 제품, 비즈니스, 기술, 재무 등의 정보를 수집하십시오.

모든 워크로드 구성 요소 분석: 워크로드의 모든 구성 요소에 대해 철저한 분석을 수행합니다. 분석 비용과 수명 주기 동안 워크로드의 잠재적 절감액이 균형을 이루는지 확인하십시오. 구성 요소의 현재 영향과 향후 잠재적 영향을 파악해야 합니다. 예를 들어 제안된 리소스의 비용이 월 10 USD 이고 예상 로드가 월 15 USD 를 초과하지 않는 경우, 시스템 수명 전체에서 얻을 수 있는 잠재적 이점보다 비용을 50%(월 5 USD) 절감하기 위한 하루 동안의 노력이 더 들 수 있습니다. 더 빠르고 효율적인 데이터 기반 추정을 사용하면 이 구성 요소에 대해 전반적으로 가장 좋은 결과를 얻을 수 있습니다.

워크로드는 시간이 지남에 따라 변경될 수 있으며 워크로드 아키텍처 또는 사용량이 변경되면 워크로드에 가장 적합한 서비스 세트도 변경될 수 있습니다. 서비스 선택을 분석할 때는 현재 및 향후 워크로드 상태 및 사용량 수준을 포함해야 합니다. 향후 워크로드 상태 또는 사용량에 대한 서비스를 구현하면 향후 변경에 필요한 노력을 줄이거나 제거하여 전반적인 비용을 절감할 수 있습니다.

[AWS Cost Explorer](#)와 [CUR](#)을 사용하여 PoC(개념 증명) 또는 환경 실행 비용을 분석할 수 있습니다. [AWS 월 사용량 계산기](#) 또는 [AWS 요금 계산기](#)로 워크로드 비용을 추정할 수도 있습니다.

관리형 서비스: 관리형 서비스를 사용하면 서비스 유지 관리를 위한 운영 및 관리 부담이 제거되므로 혁신에 집중할 수 있습니다. 또한 관리형 서비스는 클라우드 규모로 운영되기 때문에 트랜잭션 또는 서비스당 비용을 줄일 수 있습니다.

팀은 이렇게 절약한 시간을 기술적 부채 청산, 혁신 및 부가 가치 기능에 집중할 수 있습니다. 예를 들어 온프레미스 환경을 최대한 빠르게 클라우드로 "Lift and shift"하고 최적화는 나중에

수행하는 경우입니다. 라이선스 비용이 거의 발생하지 않거나, 전혀 발생하지 않는 관리형 서비스를 사용하여 실현할 수 있는 비용 절감 이점을 파악하는 것이 좋습니다.

관리형 서비스에는 대개 충분한 용량을 보장하기 위해 설정할 수 있는 속성이 있습니다. 초과 용량을 최소한으로 유지하면서 성능은 극대화할 수 있도록 이러한 속성을 설정하고 모니터링해야 합니다. AWS Management Console 또는 AWS API 및 SDK 를 사용해 AWS Managed Services 의 속성을 수정하여 변화하는 수요에 맞게 리소스 요구를 조정할 수 있습니다. 예를 들어, Amazon EMR 클러스터나 Amazon Redshift 클러스터의 노드 수를 늘리거나 줄여서 클러스터 규모를 확장하거나 감축할 수 있습니다.

또한 AWS 리소스의 여러 인스턴스를 압축하여 리소스 사용 밀도를 높일 수도 있습니다. 예를 들어 단일 Amazon Relational Database Service(Amazon RDS) DB 인스턴스에 소형 데이터베이스 여러 개를 프로비저닝할 수 있습니다. 그리고 사용량이 증가하면 스냅샷 및 복원 프로세스를 사용하여 전용 RDS DB 인스턴스에 데이터베이스 중 하나를 마이그레이션할 수 있습니다.

관리형 서비스에서 워크로드를 프로비저닝할 때는 서비스 용량 조정 요구 사항을 파악해야 합니다. 일반적으로 이러한 요구 사항은 시간, 작업량 및 정상 워크로드 작동에 미치는 영향을 의미합니다. 리소스를 프로비저닝할 때는 변경이 발생하기까지 소요되는 시간을 고려하여 이 시간을 허용하는 데 필요한 오버헤드를 프로비저닝해야 합니다. Amazon CloudWatch 등의 시스템 및 모니터링 도구와 통합된 API 와 SDK 를 사용하면 서비스를 수정하는 데 필요한 지속적인 작업을 사실상 수행하지 않아도 됩니다.

[Amazon Relational Database Service\(RDS\)](#), [Amazon Redshift](#) 및 [Amazon ElastiCache](#) 는 관리형 데이터베이스 서비스를 제공합니다. [Amazon Athena](#), [Amazon Elastic Map Reduce\(EMR\)](#) 및 [Amazon Elasticsearch](#) 는 관리형 분석 서비스를 제공합니다.

[AWS Managed Services\(AMS\)](#)는 엔터프라이즈 고객 및 파트너를 대신하여 AWS 인프라를 운영하는 서비스입니다. 이 서비스를 사용하면 규정을 준수하는 안전한 환경에 워크로드를 배포할 수 있습니다. AMS 는 자동화가 포함된 엔터프라이즈 클라우드 운영 모델을 사용하므로 고객은 조직 요구 사항을 충족하면서 클라우드로 더 빠르게 이전하고 지속적인 관리 비용을 절감할 수 있습니다.

서버리스 또는 애플리케이션 수준 서비스: [AWS Lambda](#), [Amazon Simple Queue Service\(Amazon SQS\)](#), [Amazon Simple Notification Service\(Amazon SNS\)](#), [Amazon Simple Email Service\(Amazon SES\)](#) 등의 서버리스 또는 애플리케이션 수준 서비스를 사용할 수 있습니다. 이러한 서비스를 사용하면 리소스를 관리할 필요가 없으며 코드 실행, 대기열 서비스 및 메시지 전송 기능을 제공합니다. 또 다른 이점은 사용량에 따라 성능과 비용이 확장되므로 효율적인 비용 할당 및 귀속이 가능하다는 것입니다.

서버리스에 대한 자세한 내용은 [Well-Architected 서버리스 애플리케이션 렌즈 백서를 참조하십시오.](#)

워크로드 시간대별 사용량 분석: AWS에서 새로운 서비스와 기능이 릴리스되면 워크로드에 대한 최적의 서비스가 변경될 수 있습니다. 필요한 노력에는 잠재적 이점이 반영되어야 합니다. 워크로드 검토 빈도는 조직의 요구 사항에 따라 다릅니다. 비용이 높은 워크로드인 경우 새로운 서비스를 빨리 구현할수록 비용 절감이 극대화되므로 검토를 자주 수행하는 것이 좋습니다. 사용량 패턴이 변경되는 경우에도 검토를 시행해야 합니다. 사용량의 큰 변화는 대체 서비스가 더 적합하다는 의미일 수 있습니다. 예를 들어 데이터 전송 속도가 더 높으면 직접 연결 서비스를 통해 VPN보다 저렴한 비용으로 필요한 연결을 제공할 수 있습니다. 서비스 변경의 잠재적인 영향을 예측하여 이러한 사용량 수준 트리거를 모니터링하고 가장 비용 효율적인 서비스를 더 빨리 구현할 수 있도록 하십시오.

라이선스 비용: 오픈 소스 소프트웨어를 사용하면 소프트웨어 라이선스 비용을 줄일 수 있습니다. 라이선스 비용은 워크로드 규모가 확장됨에 따라 워크로드 비용에 상당한 영향을 미칠 수 있습니다. 총 비용을 기준으로 라이선스가 부여된 소프트웨어의 이점을 측정하면 워크로드 효율성을 최대한 최적화할 수 있습니다. 라이선스 변경 사항과 이러한 변경 사항이 워크로드 비용에 미치는 영향을 모델링하십시오. 공급업체가 데이터베이스 라이선스 비용을 변경하는 경우 이 비용이 워크로드의 전반적인 효율성에 어떤 영향을 미치는지 조사하십시오. 공급업체의 기간별 요금 발표를 고려하여 제품 전반의 라이선스 변경 추세를 파악하십시오. 또한 라이선스 비용은 하드웨어에 따라 확장되는 라이선스(CPU 바인딩 라이선스)와 같이 처리량이나 사용량에 관계없이 확장될 수 있습니다. 이러한 라이선스는 해당하는 결과 없이 비용이 빠르게 증가할 수 있으므로 피해야 합니다.

[AWS License Manager](#)를 사용하면 워크로드의 소프트웨어 라이선스를 관리할 수 있습니다. 라이선스 규칙을 구성하고 필요한 조건을 적용하여 라이선스 위반을 방지하고 라이선스 초과로 인한 비용을 절감할 수 있습니다.

올바른 리소스 유형, 크기 및 개수 선택

최적의 리소스 유형, 크기 및 리소스 수를 선택하면 가장 낮은 리소스 비용으로 기술 요구 사항을 충족할 수 있습니다. 규모 조정 활동을 수행할 때는 워크로드의 모든 리소스, 각 개별 리소스의 모든 속성, 적절한 크기 조정 작업에 수반되는 노력을 고려해야 합니다. 규모 조정 활동은 AWS 가격 인하, 새 AWS 리소스 유형 등의 외부 요인과 사용 패턴의 변화로 인해 트리거되는 반복적인 프로세스일 수 있습니다. 규모 조정에 들어가는 노력의 비용이 워크로드 수명에 걸친 잠재적 절감액보다 높다면 이 작업이 단 한 번만 수행될 수도 있습니다.

AWS에서는 다음과 같은 다양한 방식이 사용됩니다.

- 비용 모델링 수행
- 지표 또는 데이터를 기준으로 크기 선택
- 자동으로 크기 선택(지표 기준)

비용 모델링: 워크로드와 각 구성 요소에 대한 비용 모델링을 수행하여 리소스 간의 균형을 파악하고 특정 성능 수준을 고려하여 워크로드의 각 리소스에 대해 적합한 크기를 찾습니다. 그리고 예상되는 다양한 부하에서 워크로드의 벤치마크 활동을 수행하여 비용을 비교합니다. 모델링 작업은 소요되는 시간 대비 구성 요소 비용 또는 예상 절감액 등의 제공될 수 있는 이점을 반영해야 합니다. 모범 사례는 [AWS Well-Architected 프레임워크의 성능 효율성 원칙](#) 백서의 검토 섹션을 참조하십시오.

[AWS Compute Optimizer](#)는 실행 중인 워크로드에 대한 비용 모델링에 도움이 될 수 있습니다. 이 서비스는 사용량 기록을 기준으로 컴퓨팅 리소스에 적합한 크기 권장 사항을 제공합니다. 무료 서비스이고 기계 학습을 활용하여 위험 수준에 따라 여러 권장 사항을 제시하므로 컴퓨팅 리소스에 사용하기에 적합한 데이터 소스입니다. [Amazon CloudWatch](#) 및 [CloudWatch Logs](#)에서 사용자 지정 로그를 데이터 원본으로 사용하여 다른 서비스 및 워크로드 구성 요소의 규모 조정 작업을 수행할 수도 있습니다.

다음은 비용 모델링 데이터 및 지표에 대한 권장 사항입니다.

- 모니터링은 최종 사용자 환경을 정확하게 반영해야 합니다. 기간의 정확한 세부 수준을 선택하고, 평균이 아닌 최대값이나 99 번째 백분위수를 적절하게 선택합니다.
- 모든 워크로드 주기를 포함하는 데 필요한 분석 기간의 정확한 세부 수준을 선택합니다. 예를 들어 분석을 2 주 동안 수행하는 경우 사용률이 높은 월 단위 주기를 분석하지 못하여 리소스가 너무 적게 프로비저닝될 수 있습니다.

지표 또는 데이터 기반 선택: 워크로드 및 리소스 특성(예: 컴퓨팅, 메모리, 처리량, 쓰기 집약형)을 기준으로 리소스 크기나 유형을 선택합니다. 일반적으로는 비용 모델링, 온프레미스 버전과 같은 워크로드의 이전 버전, 설명서 또는 워크로드와 관련된 기타 정보 출처(백서, 게시된 솔루션)를 사용하여 선택합니다.

지표 기반 자동 선택: 실행 중인 워크로드에서 나오는 활성 지표를 사용하여 해당 워크로드를 변경하는 피드백 루프를 워크로드 안에 생성합니다. [AWS Auto Scaling](#) 과 같은 관리형 서비스를 사용하여 규모 조정 작업을 대신 수행하도록 서비스를 구성할 수 있습니다. 또한 AWS 는 최소한의 노력으로 리소스를 수정할 수 있는 [API, SDK](#) 및 기능을 제공합니다. EC2 인스턴스를 중지하고 시작하도록 워크로드를 프로그래밍하면 인스턴스 크기 또는 인스턴스 유형을 변경할 수 있습니다. 이렇게 하면 변경에 필요한 거의 모든 운영 비용을 절감하면서 규모 조정의 이점을 실현할 수 있습니다.

일부 AWS 서비스에는 [S3 Intelligent-Tiering](#) 과 같은 자동 유형 또는 크기 선택 기능이 기본적으로 포함되어 있습니다. S3 Intelligent-Tiering 은 사용 패턴에 따라 자주 접근하고 자주 접근하지 않는 두 개의 접근 계층 간에 자동으로 데이터를 이동합니다

최적의 요금 모델 선택

워크로드 비용 모델링 수행: 워크로드 구성 요소의 요구 사항을 고려하고 가능한 요금 모델을 파악합니다. 구성 요소의 가용성 요구 사항을 정의합니다. 워크로드의 기능이 다수의 독립된 리소스를 통해 수행되는지 여부와 워크로드의 시간대별 요구 사항을 파악합니다. 기본 온디맨드 요금 모델 및 기타 적용 가능한 모델을 사용하여 리소스 비용을 비교합니다. 리소스 또는 워크로드 구성 요소의 잠재적 변경을 고려합니다.

정기적인 계정 수준 분석 수행: 정기적인 비용 모델링을 수행하면 여러 워크로드에 걸친 최적화 기회를 구현할 수 있습니다. 예를 들어 여러 워크로드에 온디맨드 모델을 사용하는 경우 집계 수준에서 변경 위험이 낮으므로 약정 기반 할인을 구현하여 전반적인 비용을 절감할 수 있습니다. 2 주에서 1 개월의 정기적인 주기로 분석을 수행하는 것이 좋습니다. 이렇게 하면 구매를 조금씩 조정할 수 있으므로 워크로드 및 워크로드 구성 요소의 변경에 따라 요금 모델의 적용 범위를 점진적으로 변경할 수 있습니다.

[AWS Cost Explorer](#) 권장 사항 도구를 사용하여 약정 할인의 기회를 찾을 수 있습니다.

스팟 워크로드 기회를 찾으려면 전체 사용량을 시간대별로 확인하여 사용량 또는 탄력성이 주기적으로 변경되는 기간을 찾아보십시오.

요금 모델: AWS 는 조직의 요구 사항에 적합한 가장 비용 효율적인 방법으로 리소스 요금을 지불할 수 있는 다수의 [요금 모델](#)을 제공합니다. 다음 섹션에서는 아래의 각 구매 모델에 대해 설명합니다.

- 온디맨드
- 스팟
- 약정 할인 - Savings Plans
- 약정 할인 - 예약 인스턴스/용량
- 지리적 선택
- 타사 계약 및 요금

온디맨드: 사용한 만큼 지불하는 기본 요금 모델입니다. EC2 인스턴스 등의 리소스나 DynamoDB 등의 서비스를 온디맨드 방식으로 사용할 때는 정액 요금을 결제하며 장기 약정은 없습니다. 애플리케이션의 수요에 따라 리소스나 서비스의 용량을 늘리거나 줄일 수 있습니다. 온디맨드에는 시간당 요금이 청구되지만 서비스에 따라 1 초 단위로 청구될 수 있습니다(예: AWS Lambda 또는 Linux EC2 인스턴스). 주기적으로 리소스/서비스 사용량이 급증하는 단기 워크로드(예: 4 개월 동안 진행되는 프로젝트)나 예측과 중단이 불가능한 워크로드에는 온디맨드 모델을 사용하는 것이 좋습니다. 온디맨드는 무중단 런타임이 요구되지만 약정

할인(Savings Plans 또는 예약 인스턴스)을 받을 수 있을 만큼 오래 실행되지 않는 사전 프로덕션 환경과 같은 워크로드에도 적합합니다.

스팟: [스팟 인스턴스](#)는 온디맨드 요금에 비해 최대 90% 할인된 요금으로 장기 약정 없이 제공되는 예비 EC2 컴퓨팅 용량입니다. 스팟 인스턴스를 사용하면 애플리케이션 실행 비용을 대폭 절감하거나 동일한 예산으로 애플리케이션의 컴퓨팅 파워를 확장할 수 있습니다. 온디맨드와 달리, 스팟 인스턴스는 EC2 에서 용량을 다시 확보해야 하거나 스팟 인스턴스 가격이 구성된 가격을 초과하는 경우 2 분 전 경고와 함께 인스턴스가 중단될 수 있습니다. 평균적으로 스팟 인스턴스가 중단되는 비율은 해당 시간의 5% 미만입니다.

스팟은 대기열 또는 버퍼가 있거나 요청을 처리하기 위해 독립적으로 작동하는 다수의 리소스가 있는 경우에 이상적입니다(예: 하둡 데이터 처리). 일반적으로 이러한 워크로드는 내결함성이 있고 상태 비저장(Stateless)이며 유연합니다(예: 배치 처리, 빅 데이터 및 분석, 컨테이너식 환경, HPC(고성능 컴퓨팅)). 테스트 및 개발 환경과 같은 중요하지 않은 워크로드 또한 스팟 인스턴스를 사용하기에 적합합니다.

또한 스팟은 EC2 Auto Scaling 그룹(ASG), Elastic MapReduce(EMR), Elastic Container Service(ECS) 및 AWS Batch 와 같은 여러 AWS 서비스와 통합됩니다.

스팟 인스턴스를 회수해야 하는 경우 EC2 는 CloudWatch Events 와 인스턴스 메타데이터를 통해 전달되는 스팟 인스턴스 중단 공지를 통해 2 분 전에 경고를 보냅니다. 애플리케이션은 이 2 분의 시간을 사용하여 상태를 저장하거나, 실행 중인 컨테이너를 비우거나, 최종 로그 파일을 업로드하거나, 로드 밸런서에서 애플리케이션 자체를 제거할 수 있습니다. 2 분 후에는 스팟 인스턴스를 최대 절전 모드로 전환하거나, 중지하거나, 종료할 수 있습니다.

워크로드에 스팟 인스턴스를 채택할 때는 다음 모범 사례를 고려하십시오.

- **최고 가격을 온디맨드 요금으로 설정:** 이렇게 하면 현재 스팟 요금(가장 저렴한 요금)이 부과되고 온디맨드 요금보다 높은 요금이 부과되지 않습니다. 현재 및 기간별 요금은 콘솔 및 API 를 통해 확인할 수 있습니다.
- **가능한 한 많은 인스턴스 유형을 유연하게 사용:** 인스턴스 유형의 패밀리와 크기를 유연하게 사용하여 타겟 용량 요구 사항을 충족할 가능성을 높이고, 가능한 최저 비용을 확보하며, 중단 영향 최소화합니다.

- **워크로드의 실행 위치를 유연하게 활용:** 가용 용량은 가용 영역에 따라 다를 수 있습니다. 실행 위치를 유연하게 활용하면 여러 예비 용량 풀을 사용하여 타겟 용량을 충족할 가능성을 높이고 최대한 낮은 비용을 제공할 수 있습니다.
- **연속성을 고려한 설계:** EC2 용량 중 일부가 중단되더라도 워크로드의 가용성이나 성능에 영향을 미치지 않도록 상태 비저장 및 내결함성을 고려하여 워크로드를 설계합니다.
- 성능을 유지하면서 워크로드 비용 최적화를 극대화하려면 스팟 인스턴스를 온디맨드 및 Savings Plans/예약 인스턴스와 함께 사용하는 것이 좋습니다.

약정 할인 – Savings Plans: AWS 는 특정 양의 리소스 사용을 예약하거나 약정하고 리소스에 대한 할인된 요금을 받아 비용을 절감할 수 있는 다양한 방법을 제공합니다. [Savings Plan](#) 을 사용하면 1 년 또는 3 년에 대한 시간당 지출 약정을 체결하고 리소스 전체에 할인된 요금을 적용할 수 있습니다. Savings Plans 는 EC2, Fargate 및 Lambda 와 같은 AWS 컴퓨팅 서비스에 대한 할인을 제공합니다. 약정을 체결하는 경우 시간당 약정 금액을 지불하게 되며, 이 약정 금액은 할인율로 온디맨드 사용량에서 차감됩니다. 예를 들어 시간당 50 USD 를 약정하고 온디맨드 사용량이 시간당 150 USD 인 경우 Savings Plans 요금을 고려하면 특정 사용량에 대해 50% 할인율이 적용됩니다. 따라서 50 USD 의 약정에 100 USD 의 온디맨드 사용량이 포함됩니다. 결과적으로는 50 USD(약정)와 나머지 온디맨드 사용량에 대한 50 USD 를 지불하게 됩니다.

[Compute Savings Plans](#) 는 가장 유연한 요금제로, 최대 66%의 할인을 제공합니다. 가용 영역, 인스턴스 크기, 인스턴스 패밀리, 운영 체제, 테넌시, 리전 및 컴퓨팅 서비스에 자동으로 적용됩니다.

[Instance Savings Plans](#) 는 덜 유연하지만 할인율은 더 높습니다(최대 72%). 가용 영역, 인스턴스 크기, 인스턴스 패밀리, 운영 체제 및 테넌시에 자동으로 적용됩니다.

세 가지 결제 옵션이 있습니다.

- **선결제 없음:** 선결제 금액은 없으며 매월 총 사용 시간에 해당하는 시간당 요금(할인 적용)을 결제하는 방식입니다.

- **부분 선결제:** 선결제 없음보다 더 높은 할인 요금을 제공합니다. 사용량에 해당하는 금액 중 일부를 선결제하며, 매월 총 사용 시간에 해당하는 더 저렴한 시간당 요금(할인 적용)을 결제하는 방식입니다.
- **전체 선결제:** 전체 기간의 사용량에 해당하는 금액을 선결제하며, 잔여 기간 동안 약정에 포함되는 사용량에 대한 추가 요금이 발생하지 않습니다.

이러한 세 가지 구매 옵션을 원하는 대로 조합하여 워크로드 전체에 적용할 수 있습니다.

Savings Plans 는 할인이 높은 순서에서 낮은 순서로 구매 계정의 사용량에 먼저 적용된 후 다른 모든 계정의 통합 사용량에 할인이 높은 순서에서 낮은 순서로 적용됩니다.

모든 Savings Plans 는 마스터 계정과 같은 사용량이나 리소스가 없는 계정에서 구입하는 것이 좋습니다. 이렇게 하면 Savings Plans 는 모든 사용량에 걸쳐 가장 높은 할인이 적용되므로 할인 금액을 최대화할 수 있습니다.

워크로드와 사용량은 시간 경과에 따라 변경되는 것이 일반적입니다. 소액의 Savings Plans 를 지속적으로 구매하는 것이 좋습니다. 이렇게 하면 적용 범위를 높은 수준으로 유지하여 할인을 극대화하고, 항상 워크로드 및 조직 요구 사항에 일치하는 요금제를 사용할 수 있습니다.

가능한 할인은 변동될 수 있으므로 계정에 타겟 적용 범위를 설정하지 마십시오. 적용 범위가 낮다고 해서 잠재적인 비용 절감이 꼭 높아지는 것은 아닙니다. 계정의 적용 범위가 낮더라도 사용량이 스몰 인스턴스와 라이선스가 있는 운영 체제로 구성된 경우에는 잠재적 절감액이 단 몇 %의 수준으로 낮을 수 있습니다. 따라서 Savings Plan 권장 사항 도구에서 가능한 잠재적 절감액을 추적하고 모니터링하는 것이 좋습니다. Cost Explorer 에서 Savings Plans 권장 사항을 자주 검토하고(정기적인 분석 수행) 예상 절감액이 조직에 필요한 할인 미만으로 떨어지기 전까지 약정을 계속해서 구매합니다. 예를 들어 잠재적 할인이 20% 미만으로 유지되는지 추적 및 모니터링하고, 이 비율보다 높아지면 구매합니다.

사용률과 적용 범위를 모니터링하되 변경 사항을 감지하는 용도로만 모니터링합니다. 특정 사용률(%) 또는 적용 범위(%)를 목표로 하지 마십시오. 이 비율이 늘어난다고 해서 절감액도 커지는 것은 아닙니다. Savings Plans 를 구매할 경우 적용 범위가 증가하는지 확인하십시오. 적용 범위 또는 사용률이 감소하면 이를 수량화하고 숙지합니다. 예를 들어 워크로드 리소스를

새로운 인스턴스 유형으로 마이그레이션할 경우 기존 요금제의 사용률은 감소하지만 성능상의 이점이 절감액을 증가합니다.

약정 할인 - 예약 인스턴스/약정: Savings Plans 와 마찬가지로, [예약 인스턴스](#)는 최소량의 리소스 실행을 약정할 경우 최대 72%의 할인을 제공합니다. 예약 인스턴스는 RDS, Elasticsearch, ElastiCache, Amazon Redshift 및 DynamoDB 에서 사용할 수 있습니다. Amazon CloudFront 와 AWS Elemental MediaConvert 는 최소 사용량 약정을 체결할 경우 할인을 제공합니다. 예약 인스턴스는 현재 EC2 에서 사용할 수 있지만, Savings Plans 는 향상된 유연성과 관리 오버헤드 없이 동일한 수준의 할인을 제공합니다.

예약 인스턴스는 선결제 없음, 부분 선결제 및 전체 선결제 금액과 동일한 요금 옵션을 제공하며 1 년 또는 3 년 약정으로 사용할 수 있습니다.

예약 인스턴스는 리전 또는 특정 가용 영역에서 구매할 수 있습니다. 가용 영역에서 구매할 때는 용량을 예약할 수 있습니다.

EC2 에서는 컨버터블 RI 를 사용할 수 있지만 모든 EC2 인스턴스에는 향상된 유연성과 운영 비용 절감을 제공하는 Savings Plans 를 사용해야 합니다.

예약 인스턴스를 추적하고 구매할 때도 동일한 프로세스와 지표를 사용해야 합니다. 계정 전체에서 RI 적용 범위를 추적하는 것은 권장되지 않습니다. 또한 사용률 %를 모니터링하거나 추적하지 않는 것이 좋습니다. 대신 Cost Explorer 에서 사용률 보고서를 확인하고 테이블의 순 절감액 열을 사용하십시오. 순 절감액이 아주 큰 음수인 경우 미사용 RI 를 개선하기 위한 조치를 취해야 합니다.

EC2 플릿: [EC2 플릿](#)을 사용하면 타겟 컴퓨팅 파워를 정의한 다음 해당 플릿용으로 적절하게 절충한 온디맨드 및 스팟 옵션과 인스턴스 유형을 지정할 수 있습니다. 그러면 정의된 용량을 충족할 수 있는 최저 가격의 리소스 조합이 EC2 플릿에서 자동으로 시작됩니다.

지리적 선택: 솔루션 설계 시의 모범 사례는 사용자와 더 가까운 위치에 컴퓨팅 리소스를 배치하여 지연 시간을 줄이고 데이터 주권을 강화하는 것입니다. 대상 고객이 전 세계 여러 지역에 분산되어 있는 경우에는 이러한 요구를 충족하기 위해 여러 위치를 사용해야 합니다. 비용을 최소화할 수 있는 지리적 위치를 선택해야 합니다.

AWS 클라우드 인프라는 [리전 및 가용 영역](#)을 중심으로 구축됩니다. 리전은 다수의 가용 영역이 포함된 전 세계의 물리적 위치입니다. 가용 영역은 하나 이상의 개별 데이터 센터로 구성됩니다. 각 데이터 센터는 분리된 시설에 구축되며 이중화된 전력, 네트워킹 및 연결 기능을 갖추고 있습니다.

각 AWS 리전은 지역 시장의 조건에 따라 운영되며 리소스 가격은 각 리전별로 다릅니다. 전 세계에서 사용 가능한 최저 가격으로 리소스를 실행할 수 있도록 솔루션의 한 구성 요소나 전체 솔루션을 운영할 특정 리전을 선택합니다. AWS 월 사용량 계산기를 사용하여 다양한 리전의 워크로드 비용을 추정할 수 있습니다.

타사 계약 및 요금: 클라우드에서 타사 솔루션 또는 서비스를 활용하는 경우 요금 구조는 비용 최적화 결과에 부합해야 합니다. 요금은 제공하는 결과와 가치에 따라 조정되어야 합니다. 예를 들어 제공하는 절감액의 일정 비율을 가져와서 절감액(결과)이 커질수록 더 많은 요금을 부과하는 소프트웨어가 있습니다. 청구 금액에 따라 규모가 조정되는 계약은 특정 청구 금액의 일정 부분에 대해 결과를 제공하지 않는 한 일반적으로 비용 최적화에 맞지 않습니다. 예를 들어 EC2에 대한 권장 사항을 제공하고 전체 청구 금액의 일정 비율을 부과하는 솔루션을 사용하는 경우 이 권장 사항이 어떤 이득도 되지 않는 다른 서비스를 함께 사용하면 솔루션 요금이 증가하게 됩니다. 또 다른 예로는 관리되는 리소스 비용의 일정 비율로 요금을 부과하는 관리형 서비스가 있습니다. 인스턴스 크기가 크다고 해서 반드시 더 많은 관리 노력이 필요한 것은 아니지만 부과되는 요금은 증가합니다. 따라서 이러한 서비스 요금 방식에 서비스의 효율성을 증진하는 비용 최적화 프로그램 또는 기능이 포함되어 있는지 확인해야 합니다.

데이터 전송 계획

클라우드의 장점 중 하나는 관리형 네트워크 서비스라는 것입니다. 더 이상 스위치, 라우터 및 기타 관련 네트워크 장비를 관리하고 운영할 필요가 없습니다. 클라우드의 네트워킹 리소스는 CPU 및 스토리지와 동일한 방식으로 사용되며 사용한 만큼만 요금을 지불합니다.

클라우드에서 비용 최적화를 달성하려면 네트워킹 리소스를 효율적으로 사용해야 합니다.

데이터 전송 모델링 수행: 워크로드에서 데이터 전송이 발생하는 위치, 전송 비용 및 관련된 이점을 파악하십시오. 그러면 정보를 바탕으로 결정을 내리고 아키텍처 의사 결정을 수정하거나 수락할 수 있습니다. 예를 들어 가용 영역 간에 데이터를 복제하는 다중 가용 영역

구성이 있는 경우 구조의 비용을 모델링하고 필요한 안정성 및 복원력을 달성하기에 허용 가능한 비용(두 가용 영역의 컴퓨팅 및 스토리지 비용을 지불하는 것과 유사함)인지 결정합니다.

다양한 사용 수준에 걸쳐 비용을 모델링합니다. 워크로드 사용량은 시간이 지남에 따라 변경될 수 있으며 여러 수준에서 다양한 서비스를 사용하는 것이 더 비용 효율적일 수 있습니다.

[AWS Cost Explorer](#) 또는 [CUR\(비용 및 사용량 보고서\)](#)를 사용하여 데이터 전송 비용을 파악하고 모델링하십시오. PoC(개념 증명)를 구성하거나 워크로드를 테스트하고 사실적으로 시뮬레이션된 로드로 테스트를 실행합니다. 다양한 워크로드 수요에서 비용을 모델링할 수 있습니다.

데이터 전송 최적화: 데이터 전송을 고려하여 설계를 수행하면 데이터 전송 비용을 최소화할 수 있습니다. 이 설계 과정에서는 콘텐츠 전송 네트워크를 사용해 사용자와 더 가까운 위치에 데이터를 배치하거나, 온프레미스에서 AWS 로의 전용 네트워크 링크를 사용합니다. 또한 WAN 최적화 및 애플리케이션 최적화를 사용하여 구성 요소 간에 전송되는 데이터의 양을 줄일 수도 있습니다.

데이터 전송 비용을 줄이는 서비스 선택: 지연 시간이 짧고 전송 속도가 빠른 데이터 전송용 글로벌 콘텐츠 전송 네트워크인 [Amazon CloudFront](#)는 전 세계의 엣지 로케이션에서 데이터를 캐시하므로 리소스에 대한 로드가 감소합니다. CloudFront를 사용하면 전 세계의 많은 사용자에게 콘텐츠를 전송하는 과정의 관리 작업을 줄이고 지연 시간을 최소화할 수 있습니다.

[AWS Direct Connect](#)를 사용하면 AWS에 대한 전용 네트워크 연결을 설정할 수 있습니다. 이렇게 하면 네트워크 비용을 줄이고 대역폭을 늘릴 수 있으며 인터넷 기반 연결에 비해 더 일관성이 높은 네트워크 환경을 제공할 수 있습니다.

[AWS VPN](#)을 사용하면 프라이빗 네트워크와 AWS 글로벌 네트워크 간에 안전한 프라이빗 연결을 설정할 수 있습니다. 탄력적인 완전관리형 서비스로서 빠르고 간편한 연결을 제공하므로 소규모 사무실 또는 비즈니스 파트너에 적합합니다.

[VPC 엔드포인트](#)를 사용하면 프라이빗 네트워킹을 통해 여러 AWS 서비스를 연결할 수 있으며, 퍼블릭 데이터 전송 및 [NAT 게이트웨이](#) 비용을 줄일 수 있습니다. [게이트웨이 VPC 엔드포인트](#)는 시간당 비용이 없으며 Amazon S3 및 Amazon DynamoDB를 지원합니다.

[인터페이스 VPC](#) 엔드포인트는 AWS PrivateLink 를 통해 제공되며 시간당 요금과 GB 당 사용 비용이 있습니다.

리소스

비용 효율적인 리소스에 대한 AWS 모범 사례에 대해 자세히 알아보려면 다음 리소스를 참조하십시오.

- [AWS Managed Services: Enterprise Transformation Journey 동영상](#)
- [Cost Explorer 를 사용한 비용 분석](#)
- [예약 인스턴스 권장 사항에 접근](#)
- [규모 조정 권장 사항 시작하기](#)
- [스팟 인스턴스 모범 사례](#)
- [스팟 플릿](#)
- [예약 인스턴스의 작동 방식](#)
- [AWS 글로벌 인프라](#)
- [스팟 인스턴스 어드바이저](#)
- [Well-Architected 실습 - 비용 효율적인 리소스](#)

수요 관리 및 리소스 공급

클라우드로 이전하면 필요한 용량에 대한 비용만 지불하면 됩니다. 필요할 때 워크로드 수요에 맞춰 리소스를 제공할 수 있으므로 비용이 많이 들고 낭비되는 오버프로비저닝이 필요하지 않습니다. 또한 조절, 버퍼 또는 대기열을 통해 수요를 수정하여 수요를 원활하게 하고 더 적은 리소스로 수요를 처리할 수 있습니다.

리소스 장애, 고가용성 및 프로비저닝 시간을 고려한 요구 사항에 따라 서비스를 적시에 공급하는 방식을 통해서 비용 측면의 이점을 적절하게 절충해야 합니다. 그리고 수요가

고정적이든 가변적이든 규모에 관계없이 관리 작업을 최소화할 수 있는 지표와 자동화 기능 생성을 계획해야 합니다. 수요를 수정할 때는 워크로드가 수용할 수 있는 허용 가능한 지연 및 최대 지연을 알아야 합니다.

AWS에서는 다양한 접근 방식으로 수요를 관리하고 리소스를 공급할 수 있습니다. 다음 섹션에서는 아래와 같은 방식을 사용하는 방법을 설명합니다.

- 워크로드 분석
- 수요 관리
- 수요 기반 공급
- 시간 기반 공급

워크로드 분석: 워크로드의 요구 사항을 파악합니다. 조직의 요구 사항에는 요청에 대한 워크로드 응답 시간이 나타나야 합니다. 응답 시간을 사용하면 수요가 관리되는지 여부 또는 리소스 공급이 수요에 따라 변경되는지 여부를 확인할 수 있습니다.

분석에는 수요의 예측 가능성 및 반복 가능성, 수요 변경의 속도 및 수요 변경의 규모가 포함되어야 합니다. 분석은 월말 처리 또는 휴가철 피크와 같은 계절적 변동을 포함하기에 충분히 긴 기간에 걸쳐 수행되어야 합니다.

분석 작업에 확장 구현의 잠재적 이점이 반영되는지 확인하십시오. 구성 요소의 예상 총 비용을 찾아보고 워크로드 수명에 걸쳐 사용량 및 비용이 증가하거나 감소하는지 살펴봅니다.

[AWS Cost Explorer](#) 또는 [Amazon QuickSight](#)에서 CUR 또는 애플리케이션 로그를 사용하여 워크로드 수요를 시각적으로 분석할 수 있습니다.

수요 관리

수요 관리 – 조절: 수요의 소스에 재시도 기능이 있는 경우 조절을 구현할 수 있습니다. 조절은 소스에 현재 요청을 처리할 수 없는 경우 나중에 다시 시도해야 함을 알려줍니다. 소스는 일정 시간 동안 기다린 후 요청을 다시 시도합니다. 조절을 구현하면 워크로드의 최대 리소스 양과 비용을 제한할 수 있는 장점이 있습니다. AWS에서는 [Amazon API Gateway](#)를 사용하여

조절을 구현할 수 있습니다. 조절 구현에 대한 자세한 내용은 [Well-Architected 안정성 원칙 백서](#)를 참조하십시오.

수요 관리 - 버퍼 기반: 조절과 비슷하게, 버퍼는 서로 다른 속도로 실행되는 애플리케이션이 효과적으로 통신할 수 있도록 요청 처리를 연기합니다. 버퍼링 기반 접근 방식에서는 대기열을 사용해 생산자의 메시지(작업 단위)를 수락합니다. 메시지는 소비자가 읽은 후 처리되므로 소비자의 비즈니스 요구 사항을 충족하는 속도로 메시지를 실행할 수 있습니다. 생산자는 데이터 내구성 및 백 프레셔(소비자 실행 속도가 느려서 생산자의 속도도 느려지는 현상)와 같은 조절 관련 문제를 처리할 필요가 없습니다.

AWS에서는 여러 서비스 중에서 버퍼링 방식을 구현하는 데 적합한 서비스를 선택할 수 있습니다. [Amazon SQS](#)는 단일 소비자가 개별 메시지를 읽을 수 있는 대기열을 제공하는 관리형 서비스입니다. [Amazon Kinesis](#)에서는 여러 소비자가 같은 메시지를 읽을 수 있는 스트림을 제공합니다.

버퍼 기반 접근 방식으로 아키텍처를 설계할 때는 필요한 시간에 요청을 처리하도록 워크로드를 설계해야 하며 작업에 대한 중복 요청을 처리할 수 있는 기능이 있어야 합니다.

동적 공급

수요 기반 공급: 클라우드의 탄력성을 활용하여 수요 변경에 맞춰 리소스를 공급합니다. API 또는 서비스 기능을 활용하면 프로그래밍 방식을 통해 아키텍처의 클라우드 리소스 양을 동적으로 변경할 수 있습니다. 이렇게 하면 아키텍처의 구성 요소를 확장할 수 있으며, 수요 급증 기간 동안에는 리소스 수를 자동으로 늘려 성능을 유지하고 수요 감소 기간에는 용량을 줄여 비용을 절감할 수 있습니다.

[Auto Scaling](#)은 용량을 조정하여 최대한 저렴한 비용으로 안정적이고 예측 가능한 성능을 유지하는 데 도움이 됩니다. Amazon EC2 인스턴스 및 스팟 플릿, Amazon ECS, Amazon DynamoDB 및 Amazon Aurora와 통합되는 완전관리형의 무료 서비스입니다.

Auto Scaling이 제공하는 자동 리소스 검색 기능을 사용하면 워크로드에서 구성 가능한 리소스를 쉽게 찾을 수 있습니다. 또한 기본적으로 포함되어 있는 확장 전략을 통해 성능, 비용 또는 둘 사이의 균형을 최적화할 수 있으며 예측 조정 기능을 통해 주기적으로 발생하는 스파이크를 지원할 수 있습니다.

Auto Scaling 은 수동, 예약 또는 수요 기반 조정을 구현할 수 있습니다. [Amazon CloudWatch](#) 의 지표 및 경보를 사용하여 워크로드에 대한 조정 이벤트를 트리거할 수도 있습니다. 일반적인 지표로는 CPU 사용률, 네트워크 처리량 및 ELB 에서 관찰된 요청/응답 지연 시간과 같은 표준 Amazon EC2 지표가 있습니다. 가능한 경우 고객 경험을 나타내는 지표를 사용해야 합니다. 일반적으로 이 지표는 워크로드 내 애플리케이션 코드에서 생성될 수 있는 사용자 지정 지표입니다.

수요 기반 방식을 사용하여 설계할 때는 두 가지 주요 사항을 고려해야 합니다. 먼저 새 리소스를 프로비저닝해야 하는 속도를 파악해야 합니다. 그리고 수요와 공급 간의 차이 규모는 변화한다는 점을 이해해야 합니다. 따라서 수요 변화 속도에 맞게 공급 속도를 변경할 수 있도록 준비하는 동시에 리소스 장애에도 대비해야 합니다.

[Elastic Load Balancing](#)(ELB)은 여러 리소스에 걸쳐 수요를 분산하여 확장하는 데 도움이 됩니다. 추가 리소스를 구현할 때는 로드 밸런서에 리소스를 추가하여 수요를 처리합니다. AWS ELB 는 EC2 인스턴스, 컨테이너, IP 주소 및 Lambda 함수를 지원합니다.

시간 기반 공급: 시간 기반 방식에서는 시간별로 예측 가능하거나 적절하게 정의되는 수요에 맞게 리소스 용량을 조정합니다. 이 방식에서는 일반적으로 리소스 용량이 리소스 사용률 수준에 따라 달라지지 않습니다. 시간 기반 방식을 사용하면 필요한 특정 시간에 리소스를 사용할 수 있으며, 시작 절차 및 시스템 또는 일관성 검사로 인한 지연 없이 리소스를 제공할 수 있습니다. 또한 사용량이 많은 기간 동안 추가 리소스를 제공하거나 용량을 늘릴 수 있습니다.

예약된 Auto Scaling 을 사용하면 시간 기반 접근 방식을 구현할 수 있습니다. 사용자 또는 수요 도달 시 리소스를 사용할 수 있도록 정의된 시간(예: 업무 시간 시작 시)에 워크로드 규모 확장/감축을 예약할 수 있습니다.

또한 [AWS API 및 SDK](#) 와 [AWS CloudFormation](#) 을 활용하면 전체 환경을 필요할 때 자동으로 프로비저닝하고 폐기할 수 있습니다. 이 방식은 정의된 업무 시간이나 일정 기간 동안에만 실행되는 개발 또는 테스트 환경에 적합합니다.

API 를 사용해 환경 내에서 리소스 규모를 조정할 수 있습니다(수직 확장). 예를 들어 인스턴스 크기나 클래스를 변경하여 프로덕션 워크로드의 규모를 확장할 수 있습니다. 이렇게 하려면 인스턴스를 중지했다가 시작한 후 다른 인스턴스 크기나 클래스를 선택합니다. 크기를

늘리거나, 성능(IOPS)을 조정하거나, 사용 중에 볼륨 유형을 변경하기 위해 수정할 수 있는 EBS 탄력적 볼륨 등의 다른 리소스에도 이 기술을 적용할 수 있습니다.

시간 기반 방식을 사용하여 설계할 때는 두 가지 주요 사항을 고려해야 합니다. 먼저 사용 패턴의 일관성 정도를 파악해야 합니다. 그리고 패턴 변경 시의 영향을 고려해야 합니다. 워크로드를 모니터링하고 비즈니스 인텔리전스를 사용하면 예측 정확도를 높일 수 있습니다. 사용 패턴이 크게 변경되는 경우에는 패턴이 변경된 기간이 포함되도록 시간을 조정할 수 있습니다.

동적 공급: [AWS Auto Scaling](#) 을 사용하거나 코드에 크기 조정을 포함할 수 있습니다. 코드에 포함하려는 경우에는 [AWS API 또는 SDK](#) 를 사용하면 됩니다. 이렇게 하면 환경을 수동으로 변경하는 데 따른 운영 비용이 제거되므로 전반적인 워크로드 비용이 절감되며 변경을 훨씬 더 빠르게 수행할 수 있습니다. 또한 항상 수요와 가장 일치하는 양의 워크로드 리소스를 공급할 수 있습니다.

리소스

수요 관리 및 리소스 공급에 대한 AWS 모범 사례에 대해 자세히 알아보려면 다음 리소스를 참조하십시오.

- [API Gateway 조절](#)
- [Amazon SQS 시작하기](#)
- [Amazon EC2 Auto Scaling 시작하기](#)

시간 경과에 따른 최적화

AWS에서는 새로운 서비스를 검토하고 워크로드에 구현함으로써 시간이 지남에 따라 최적화를 수행할 수 있습니다.

새로운 서비스 검토 및 구현

AWS 에서 신규 서비스와 기능이 릴리즈 되면 기존에 결정한 아키텍처 관련 사항을 검토하여 해당 사항이 비용 측면에서 여전히 효율적으로 유지되는지 확인하는 것이 좋습니다. 요구 사항이 변경되면 더 이상 필요하지 않은 리소스, 구성 요소 및 워크로드는 과감하게 폐기하십시오. 다음을 고려하여 시간 경과에 따른 최적화를 지원하십시오.

- 워크로드 검토 프로세스 개발
- 서비스 검토 및 구현

워크로드 검토 프로세스 개발: 항상 가장 비용 효율적인 워크로드를 유지하려면 워크로드를 정기적으로 검토하여 새로운 서비스, 기능 및 구성 요소를 구현할 기회가 있는지 확인해야 합니다. 전반적인 비용 절감을 실현하려면 검토 프로세스가 잠재적인 절감액에 비례해야 합니다. 예를 들어 전체 지출의 50%를 차지하는 워크로드는 전체 지출의 5%를 차지하는 워크로드보다 더 정기적으로, 더 철저히 검토되어야 합니다. 외부 요인 또는 변동성 고려. 워크로드가 특정 지역 또는 시장 부문에 서비스를 제공하고 있고 해당 영역의 변화가 예측되는 경우에는 자주 검토하여 비용을 절감할 수 있습니다. 검토에서 고려할 또 다른 요인은 변경 구현에 들어가는 노력입니다. 변경 사항을 테스트하고 검증하는 데 상당한 비용이 발생한다면 검토 빈도를 줄여야 합니다.

오래된 레거시 구성 요소 및 리소스를 유지 관리하는 데 드는 장기적인 비용과 여기에 새로운 기능을 구현할 수 없다는 점을 고려하십시오. 현재의 테스트 및 검증 비용이 제안된 이점을 상회할 수 있습니다. 그러나 시간이 지남에 따라 워크로드와 현재 기술 간의 격차가 증가하면 변경 비용이 크게 증가하여 비용이 훨씬 높아질 수 있습니다. 예를 들어 새로운 프로그래밍 언어로 이동하는 비용은 현재로서 비용 효율적이지 않을 수 있습니다. 하지만 5년 안에는 해당 언어에 숙련된 인력을 구하는 비용이 증가할 수 있으며, 워크로드 증가로 인해 더 큰 시스템을 새로운 언어로 전환하는 데 이전보다 더 많은 노력이 필요하게 될 것입니다.

워크로드를 구성 요소로 나누고 구성 요소의 비용을 할당한 다음(추정치로 충분함) 각 구성 요소 옆에 요인(예: 작업에 들어가는 노력 및 외부 시장 요인)을 나열하십시오. 이러한 지표를 사용하여 각 워크로드에 대한 검토 빈도를 결정합니다. 예를 들어 높은 비용, 낮은 변경 노력 및 높은 외부 요인으로 분류되는 웹 서버의 경우 검토 빈도가 높을 수 있습니다. 중앙

데이터베이스는 중간 비용, 높은 변화 노력, 낮은 외부 요인으로 분류되므로 검토 빈도는 중간일 수 있습니다.

워크로드 검토 및 서비스 구현: 새로운 AWS 서비스 및 기능의 이점을 실현하려면 워크로드에 대한 검토 프로세스를 실행하고 필요에 따라 새로운 서비스와 기능을 구현해야 합니다. 예를 들어 워크로드를 검토한 후 메시징 구성 요소를 Amazon Simple Email Service(SES)로 대체할 수 있습니다. 이렇게 하면 인스턴스 플릿의 운영 및 유지 관리 비용을 절감하면서 모든 기능을 제공할 수 있습니다.

결론

비용 최적화 및 클라우드 재무 관리는 지속적인 노력입니다. 따라서 정기적으로 재무 팀 및 기술 팀과 협력하고, 아키텍처 접근 방식을 검토하고, 구성 요소 선택을 업데이트해야 합니다.

AWS 를 활용하면 복원력과 응답성이 뛰어난 적응형 배포를 구축하면서 비용을 최소화할 수 있습니다. 이 백서에서 설명한 도구, 기술 및 모범 사례를 활용하면 배포의 비용을 최대한 최적화할 수 있습니다.

기고자

이 문서를 작성하는 데 도움을 주신 분들입니다.

- Philip Fitzsimons, Well-Architected 선임 관리자, Amazon Web Services
- Nathan Besh, Well-Architected 비용 부문 담당자, Amazon Web Services
- Levon Stepanian, Amazon Web Services
- Keith Jarrett, 비즈니스 개발 부문 담당자 – 비용 최적화
- PT Ng, 커머셜 솔루션 아키텍트, Amazon Web Services
- Arthur Basbaum, 비즈니스 개발자/관리자, Amazon Web Service
- Jarman Hauser, 커머셜 솔루션 아키텍트, Amazon Web Services

추가 자료

다음에서 추가 정보를 참조하십시오.

- [AWS Well-Architected 프레임워크](#)

문서 개정

날짜	설명
2020 년 4 월	CFM, 새로운 서비스 및 Well-Architected 와의 통합을 포함하도록 업데이트되었습니다.
2018 년 7 월	AWS 에 대한 변경 사항을 반영하고 고객 검토에서 파악한 내용을 포함하도록 업데이트되었습니다.
2017 년 11 월	AWS 에 대한 변경 사항을 반영하고 고객 검토에서 파악한 내용을 포함하도록 업데이트되었습니다.
2016 년 11 월	초판