AWS Well-Architected 프레임워크

비용 최적화 요소

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

비용 최적화 요소: AWS Well-Architected 프레임워크

Copyright © 2025 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

요약 및 소개 1
요약 1
소개 1
비용 최적화 3
설계 원칙 3
정의 4
클라우드 재무 관리 시행 5
COST01-BP01 비용 최적화 소유권 설정 8
구현 가이드 8
리소스 10
COST01-BP02 재무 팀과 기술 팀 간의 파트너십 수립 10
구현 가이드 8
리소스 10
COST01-BP03 클라우드 예산 및 예측 수립 14
구현 가이드 8
리소스 10
COST01-BP04 조직의 프로세스에서 비용 인식 구현 18
구현 가이드 8
리소스 10
COST01-BP05 비용 최적화 보고 및 알림 19
구현 가이드 8
리소스 10
COST01-BP06 사전 예방적 비용 모니터링 22
구현 가이드 8
리소스 10
COST01-BP07 새로운 서비스 릴리스로 최신 상태 유지 23
구현 가이드 8
리소스 10
COST01-BP08 비용 인식 문화 조성 25
구현 가이드 8
리소스 10
COST01-BP09 비용 최적화의 비즈니스 가치 정량화 26
구현 가이드 8
리소스 10

지출 및 사용량 인식 ... 29
지배구조 ... 29
COST02-BP01 조직 요구 사항에 따라 정책 개발 ... 30
COST02-BP02 목표 및 타겟 이행 ... 33
COST02-BP03 계정 구조 구현 ... 38
COST02-BP04 그룹 및 역할 구현 ... 42
COST02-BP05 비용 제어 기능 구현 ... 44
COST02-BP06 프로젝트 수명 주기 추적 ... 46
비용 및 사용량 모니터링 ... 48
COST03-BP01 세부 정보 소스 구성 ... 49
COST03-BP02 비용 및 사용량에 조직 정보 추가 ... 51
COST03-BP03 비용 귀속 범주 식별 ... 53
COST03-BP04 조직 지표 설정 ... 56
COST03-BP05 결제 및 비용 관리 도구 구성 ... 57
COST03-BP06 워크로드 지표를 기준으로 비용 할당 ... 60
리소스 폐기 ... 61
COST04-BP01 수명 주기 동안 리소스 추적 ... 61
COST04-BP02 폐기 프로세스 구현 ... 63
COST04-BP03 리소스 폐기 ... 66
COST04-BP04 리소스 자동 폐기 ... 66
COST04-BP05 데이터 보존 정책 적용 ... 68
비용 효율적인 리소스 ... 70
서비스를 선택할 때 비용 평가 ... 70
COST05-BP01 조직의 비용 요구 사항 파악 ... 70
COST05-BP02 워크로드의 모든 구성 요소 분석 ... 72
COST05-BP03 각 구성 요소의 철저한 분석 수행 ... 74
COST05-BP04 비용 효율적인 라이선스가 포함된 소프트웨어 선택 ... 76
COST05-BP05 조직의 우선순위에 따라 비용을 최적화할 이 워크로드의 구성 요소 선택 ... 78
COST05-BP06 시간별로 사용량이 달라지는 경우 비용 분석 수행 ... 80
올바른 리소스 유형, 크기 및 개수 선택 ... 81
COST06-BP01 비용 모델링 수행 ... 82
COST06-BP02 데이터를 기준으로 리소스 유형, 크기, 개수 선택 ... 84
COST06-BP03 지표를 기준으로 리소스 유형, 크기, 개수 자동 선택 ... 86
COST06-BP04 공유 리소스 사용 고려 ... 88
최적의 요금 모델 선택 ... 91
COST07-BP01 요금 모델 분석 수행 ... 96

COST07-BP02 비용을 기준으로 리전 선택 98
COST07-BP03 비용 효율적인 조건을 갖춘 서드파티 계약 선택 101
COST07-BP04 워크로드의 모든 구성 요소에 대한 요금 모델 구현 102
COST07-BP05 관리 계정 수준에서 요금 모델 분석 수행 104
데이터 전송 계획 106
COST08-BP01 데이터 전송 모델링 수행 106
COST08-BP02 데이터 전송 비용을 최적화할 구성 요소 선택 109
COST08-BP03 데이터 전송 비용을 줄이기 위한 서비스 구현 110
수요 관리 및 리소스 공급 113
COST09-BP01 워크로드 수요 분석 수행 113
구현 가이드 8
리소스 10
COST09-BP02 수요 관리를 위한 버퍼 또는 제한 구현 115
구현 가이드 8
리소스 10
COST09-BP03 동적으로 리소스 공급 118
구현 가이드 8
구현 단계 9
리소스 10
시간 경과에 따른 최적화 125
검토 프로세스 정의 및 정기적인 워크로드 분석 125
COST10-BP01 워크로드 검토 프로세스 개발 125
COST10-BP02 정기적으로 워크로드 검토 및 분석 127
운영 자동화 129
COST11-BP01 운영 자동화 실행 129
결론 133
기여자 134
참조 자료 135
문서 수정 136
고지 사항 138
AWS 용어집 139

비용 최적화 원칙 - AWS Well-Architected Framework

게시 날짜: 2024년 6월 27일(문서 수정)

요약

이 백서에서는 Amazon Web Services(AWS) Well-Architected Framework의 비용 최적화 원칙을 중점적으로 다룹니다. 또한 AWS 환경 설계, 제공 및 유지 관리에 모범 사례를 적용할 때 참조할 수 있는 지침을 제공합니다.

비용 최적화 워크로드는 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, 서비스 구성 및 운영, 비용 최적화 기법 적용을 위한 상세한 지침을 제공합니다.

소개

AWS Well-Architected Framework는 AWS에서 워크로드를 구축할 때 내리는 의사 결정을 이해하는 데 도움이 됩니다. 이 프레임워크는 클라우드에서 신뢰할 수 있고 안전하며 효율적이고 경제적이며 지속 가능한 워크로드를 설계하고 운영하기 위한 아키텍처 모범 사례를 제공합니다. 모범 사례를 기준으로 아키텍처를 일관적으로 측정하고 개선 영역을 식별하는 방법도 알아볼 수 있습니다. 워크로드를 제대로 설계하면 비즈니스 성공 가능성이 커집니다.

이 프레임워크는 다음 6가지 원칙을 기반으로 합니다.

- 운영 우수성
- 보안
- 신뢰성
- 성능 효율성
- 비용 최적화
- 지속 가능성

이 백서에서는 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 조달 프로세스를 진행하면서 향후의 용량 및 비즈니스 요구 사항을 예측해야 하기 때문입니다. 이 백서의 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 최고 기술 책임자(CTO), 최고 재무 책임자(CFO), 아키텍처 설계자, 개발자, 재무 관리자, 재무 계획자, 비즈니스 분석가 및 운영 팀원과 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.

주제

- 설계 원칙
- 정의

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 168시간에 비해 40시간으로 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.

획일적인 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 인프라가 아니라 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. 그 결과 투자 대비 수익률(ROI)을 측정할 수 있어 워크로드 소유자에게는 리소스를 최적화하고 비용을 절감하는 기회가 됩니다.

정의

클라우드의 비용 최적화에는 5가지 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 Framework의 다른 원칙과 마찬가지로 비용 최적화에도 장단점이 있습니다. 예를 들어 최적화의 기준을 출시 시간에 둘지, 비용에 둘지 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 '리프트 앤 시프트'한 후 최적화해야 하는 경우에는 이 선택이 적합할 수 있습니다.

사전에 비용 최적화 전략에 적당한 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

클라우드 재무 관리를 위해서는 기존 재무 프로세스를 발전시켜 AWS 환경에 맞는 비용 투명성, 제어, 계획 및 최적화를 수립하고 운영해야 합니다.

동적 클라우드 사용에 기존의 정적 워터폴 계획, IT 예산 책정 및 비용 평가 모델을 적용하면 리스크가 발생할 수 있으며 부정확한 계획으로 이어져 가시성이 떨어지는 결과를 낳습니다. 결국, 이것은 비용을 효과적으로 최적화하고 제어하며 장기적인 비즈니스 가치를 실현할 기회를 상실하는 결과로 이어집니다. 이와 같은 위험을 방지하려면, 클라우드에 기본적으로 애플리케이션을 구축하는지, 클라우드에 워크로드를 마이그레이션하는지 아니면 클라우드 서비스의 도입을 확대하는지 여부에 관계없이 클라우드 여정을 통해 비용을 적극적으로 관리해야 합니다.

AWS에서 사용과 규모가 증가함에 따라 재무, 제품, 기술 및 비즈니스 조직은 클라우드 재무 관리(CFM)를 통해 비용을 관리, 최적화 및 계획할 수 있습니다. CFM의 기본 목표는 고객이 가장 비용 효율적인 방식으로 비즈니스 성과를 달성하고, 경제 및 비즈니스 가치 창출을 가속화하는 동시에 민첩성과 제어 간 올바른 균형을 찾도록 하는 것입니다.

CFM 솔루션은 비용 투명성, 제어, 예측 및 최적화를 통한 비즈니스 혁신을 지원합니다. 또한 이러한 솔루션은 모든 팀과 기능에 대한 책임을 지는 비용 인식 문화를 조성할 수 있습니다. 재무 팀은 비용이 어디에서 오는지 확인하고, 예상치 못한 비용을 최소화하여 운영을 실행하며, 동적 클라우드 사용을 계획하고, 클라우드 비용을 절감하는 동시에 팀의 클라우드 도입을 확대할 수 있습니다. 엔지니어링 팀과 이를 공유하면 리소스 선택, 사용 및 최적화에 필요한 재무 컨텍스트를 확보할 수 있습니다.

AWS CFM은 클라우드 비용을 관리, 최적화 및 계획하는 동시에 비즈니스 민첩성을 유지할 수 있는 다양한 기능을 제공합니다. CFM은 비용의 효과적인 관리는 물론, 투자를 통해 예상되는 비즈니스 결과가 주도되고 있는지 확인하기 위한 가장 중요한 요소입니다. AWS 클라우드에서 클라우드 재무 관리 프레임워크의 네 가지 원칙은 확인, 저장, 계획, 실행과 같습니다. 이러한 각 원칙에는 여러 활동과 기능이 있습니다.

클라우드 재무 관리의 네 가지 원칙.

- 참조: 현재 클라우드 지출을 어떻게 측정, 모니터링 및 책정하고 있나요? AWS가 처음이거나 AWS 사용을 계획 중인 경우 비용 및 사용 가시성을 확립할 계획이 있으신가요?

  AWS 비용을 이해하고 지출을 최적화하려면 해당 비용이 어디에서 오는지 파악해야 합니다. 이를 위해서는 계정과 리소스에 대한 신중한 구조를 통해 재무 조직이 지출 흐름을 추적하고 팀이 수익에서 차지하는 부분에 대해 책임을 지도록 해야 합니다.

  AWS 서비스: AWS Control Tower, AWS Organizations, 비용 할당 태그, 태그 정책, AWS Resource Groups, AWS 비용 범주, AWS Cost Explorer, AWS Cost and Usage Report, RI 및 SP

  리소스: AWS 태그 지정 모범 사례, AWS 비용 범주

- 절감: 지출을 최적화하기 위해 현재 어떤 비용 최적화 도구를 사용하고 있나요? AWS를 사용하지 않는 경우 일반적인 사용량 기반 및 요금 모델 기반 최적화에 익숙하신가요?

  절감 원칙을 위해 우리는 요금 및 리소스 권장 사항으로 비용을 최적화합니다. 비용 최적화는 새로운 클라우드 운영 모델에 대한 잘 정의된 전략을 보유하는 것으로 시작합니다. 이상적으로는 클라우드 여정에서 가능한 한 조기에 시작하여 올바른 프로세스와 행동으로 강화된 비용 인식 문화를 위한 발판을 마련해야 합니다.

  클라우드 비용을 최적화할 수 있는 방법은 다양합니다. 그 중 하나는 올바른 구매 모델(RI 및 SP)을 선택하거나, 워크로드가 변경되지 않고 컨테이너화되어 Amazon EC2 스팟 인스턴스를 도입할지 여부를 지정하는 것입니다. 또한 Amazon EC2 Auto Scaling 그룹을 사용하여 워크로드 규모를 조정합니다.

AWS 서비스: RI 및 SP, Amazon EC2 Auto Scaling 그룹, 스팟 인스턴스

리소스: 예약형 인스턴스, 절감형 플랜, Amazon EC2 처리 모범 사례

- 계획: 현재 향후 클라우드 사용 및 지출을 어떻게 계획하고 있나요? 새로운 마이그레이션을 위한 가치 창출을 정량화할 수 있는 방법론이 있으신가요? 클라우드의 다양한 사용을 도입하기 위해 현재의 예산 책정 및 예측 프로세스를 발전시키셨나요?

  계획 원칙은 유연한 예산 책정과 예측을 통해 계획을 개선하는 것을 의미합니다. 가시성 및 비용 제어를 설정한 후에는 클라우드 프로젝트에 대한 지출을 계획하고 기대치를 설정하는 것이 좋습니다. AWS를 통해 동적 예측 및 예산 책정 프로세스를 구축할 수 있는 유연성을 확보할 수 있습니다. 따라서 비용이 예산 한도를 초과하지 않는지 아니면 초과하는지에 관한 정보를 파악할 수 있습니다.

  AWS 서비스: AWS Cost Explorer, AWS Cost and Usage Report, AWS Budgets

  리소스: 사용량 기반 예측, AWS 예산 보고서 및 알림

- 실행: 현재 클라우드 지출을 관리하기 위해 어떤 운영 프로세스와 도구를 사용하고 있으며, 해당 작업의 책임자는 누구인가요? AWS 사용을 시작한 후로 일상적인 운영 관점에서 일이 어떻게 진행될지 생각해 보셨나요?

  실행 원칙은 청구 및 비용 제어를 실제 관리하는 것입니다. 가드레일을 구축하고 거버넌스를 설정하여 비용이 예산에 맞게 유지되도록 보장할 수 있습니다. AWS는 이를 시작할 수 있는 여러 도구를 제공합니다.

  AWS 서비스: AWS 결제 및 비용 관리 콘솔, AWS Identity and Access Management, 서비스 제어 정책(SCP), AWS Service Catalog, AWS Cost Anomaly Detection, AWS Budgets

  리소스: AWS Billing 콘솔 시작하기

다음은 클라우드 재무 관리 모범 사례입니다.

모범 사례

- COST01-BP01 비용 최적화 소유권 설정
- COST01-BP02 재무 팀과 기술 팀 간의 파트너십 수립
- COST01-BP03 클라우드 예산 및 예측 수립
- COST01-BP04 조직의 프로세스에서 비용 인식 구현
- COST01-BP05 비용 최적화 보고 및 알림
- COST01-BP06 사전 예방적 비용 모니터링

- COST01-BP07 새로운 서비스 릴리스로 최신 상태 유지
- COST01-BP08 비용 인식 문화 조성
- COST01-BP09 비용 최적화의 비즈니스 가치 정량화

COST01-BP01 비용 최적화 소유권 설정

조직 전체에서 비용 인식의 확립과 유지를 담당하는 팀(클라우드 비즈니스 오피스 또는 클라우드 혁신 센터 또는 FinOps 팀)을 구성합니다. 비용 최적화 책임자는 전체 조직 및 클라우드 관련 재무를 이해하는 개인 또는 팀(재무, 기술 및 비즈니스 팀 인력 필요)이 될 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

여기에서는 클라우드 컴퓨팅에서 비용 인식 문화의 확립과 유지를 담당하는 클라우드 비즈니스 오피스(CBO) 또는 클라우드 혁신 센터(CCoE) 부서 또는 팀을 소개합니다. 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 또는 인프라 개발 역량을 바탕으로 다방면에 걸친 접근을 필요로 합니다. 이 역할은 다음 세 가지 부문 내에서 비용 최적화를 실행하여 워크로드의 효율성을 개선할 수 있습니다.

- 중앙화: 고객은 FinOps 팀, 클라우드 재무 관리(CFM) 팀, 클라우드 비즈니스 오피스(CBO) 또는 클라우드 혁신 센터(CCoE)와 같은 지정된 팀을 통해 거버넌스 메커니즘을 설계 및 구현하고 전사적으로 모범 사례를 추진할 수 있습니다.
- 탈집중화: 기술 팀이 비용 최적화를 실행하도록 영향을 줍니다.
- 하이브리드: 중앙화 팀과 탈중앙화 팀이 함께 비용 최적화를 실행하기 위해 협력할 수 있습니다.

이 역할의 성과는 비용 최적화 목표(예: 워크로드 효율성 지표)를 실행하고 제공할 수 있는 능력을 기준으로 측정할 수 있습니다.

이 역할에 대한 경영진 차원의 지원을 아끼지 않는 것이 성공의 핵심 요소입니다. 스폰서는 비용 효율적인 클라우드 사용의 옹호자로 간주되며, 팀에 에스컬레이션 지원을 제공하여 비용 최적화 활동이 조

직이 정의한 우선순위 수준에 따라 처리될 수 있도록 합니다. 그렇지 않으면 지침이 무시되고 비용 절감 기회를 우선시하지 않게 됩니다. 스폰서와 팀은 조직이 클라우드를 효율적으로 사용하고 비즈니스 가치를 전달할 수 있도록 함께 돕습니다.

Business, Enterprise-On-Ramp 또는 Enterprise 지원 플랜을 사용 중이며 이러한 팀 또는 역할을 구축하는 데 도움이 필요한 경우 계정 팀을 통해 클라우드 재무 관리(CFM) 전문가에게 문의하세요.

구현 단계

- 주요 구성원 정의: 조직의 모든 관련 부서가 비용 관리에 기여하고 관심을 가져야 합니다. 조직 내의 공통 팀으로는 일반적으로 재무, 애플리케이션 또는 제품 책임자, 관리, 기술 팀(DevOps) 등이 있습니다. 일부(재무 또는 기술)는 정규직이며 일부는 필요에 따라 주기적으로 근무합니다. CFM을 수행하는 개인 또는 팀에는 다음과 같은 기술이 필요합니다.
  - 소프트웨어 개발: 스크립트 및 자동화가 구축되는 경우에 해당합니다.
  - 인프라 엔지니어링: 스크립트를 배포하고 프로세스를 자동화하고 서비스 또는 리소스 프로비저닝 방법을 파악하려는 경우에 해당합니다.
  - 운영 감각: CFM은 클라우드의 효율적인 사용을 측정, 모니터링, 수정, 계획 및 조정하여 클라우드를 효율적으로 운영할 수 있는 솔루션입니다.
- 목표 및 지표 정의: 이 역할은 조직에 가치를 다양한 방법으로 전달해야 합니다. 역할 목표는 정의되며 조직이 발전함에 따라 계속해서 변화합니다. 공통 활동으로는 조직 전체의 비용 최적화에 대한 교육 프로그램 생성 및 실행, 비용 최적화를 위한 모니터링 및 보고와 같은 조직 차원의 표준 개발, 최적화에 대한 워크로드 목표 설정 등이 있습니다. 또한 이 역할은 조직의 비용 최적화 역량에 대해 조직에 정기적으로 보고해야 합니다.

  가치 또는 비용 기반 핵심 성과 지표(KPI)를 정의할 수 있습니다. KPI를 정의할 때 효율성과 예상되는 비즈니스의 성과 측면에서 예상 비용을 계산할 수 있습니다. 가치 기반 KPI는 비용 및 사용량 지표를 비즈니스 가치 동인과 연계하고 AWS 지출 변화를 합리화하는 데 도움이 됩니다. 가치 기반 KPI를 이끌어 내기 위한 첫 번째 단계는 표준 KPI 세트를 선정하고 합의하기 위해 조직 전체에서 협력하는 것입니다.
- 정기적인 주기 설정: 그룹(재무, 기술 및 비즈니스 팀)은 정기적으로 함께 모여 목표와 지표를 검토해야 합니다. 일반적인 주기는 조직의 상태를 검토하고 현재 실행 중인 프로그램을 검토한 후 전반적인 재무 및 최적화 지표를 검토하는 것입니다. 그 후에는 주요 워크로드가 보다 자세히 보고됩니다.

  이러한 정기 검토 중에 워크로드 효율성(비용) 및 비즈니스 성과를 검토할 수 있습니다. 예를 들어, 워크로드에 대한 20%의 비용 증가는 고객 사용량 증가와 일치할 수 있습니다. 이 경우 20%라는 비용 증가는 투자로 해석할 수 있습니다. 이처럼 정기적인 주기의 회의는 팀이 전체 조직에 의미를 줄 수 있는 가치 KPI를 식별하는 데 도움이 될 수 있습니다.

리소스

관련 문서:

- AWS CCOE 블로그
- Creating Cloud Business Office
- CCOE - Cloud Center of Excellence

관련 비디오:

- Vanguard CCOE Success Story

관련 예제:

- Using a Cloud Center of Excellence (CCOE) to Transform the Entire Enterprise
- Building a CCOE to transform the entire enterprise
- 7 Pitfalls to Avoid When Building CCOE

COST01-BP02 재무 팀과 기술 팀 간의 파트너십 수립

클라우드 여정의 모든 단계에서 비용 및 사용량 논의에 재무 및 기술 팀을 참여시킵니다. 팀은 정기적으로 만나 조직의 목적과 목표, 비용 및 사용량의 현재 상태, 재무 및 회계 실무와 같은 주제에 대해 논의합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

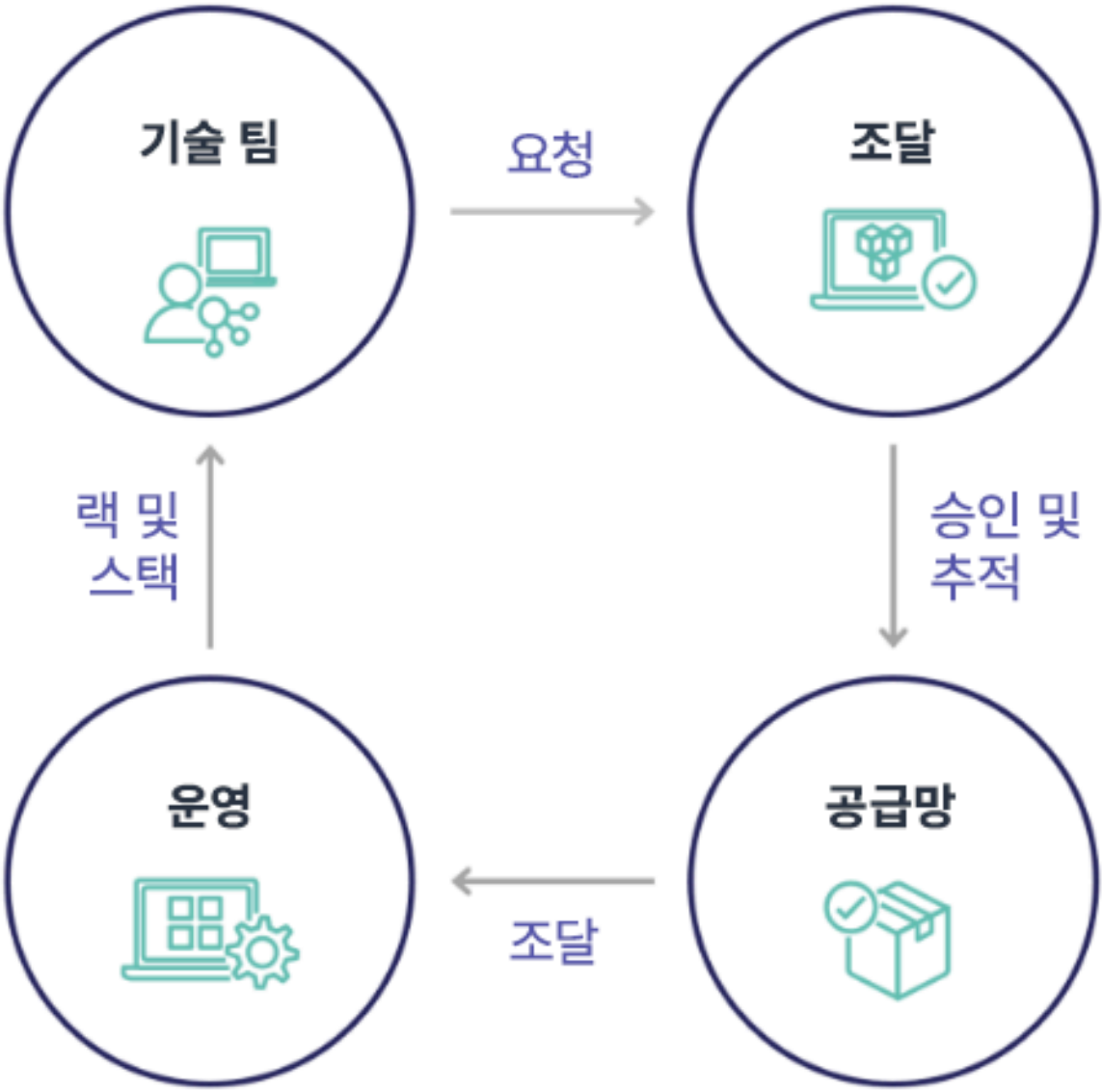
구현 가이드

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.

재무 및 조달 조직 관점에서 자본 예산 책정, 자본 요청, 승인, 조달 및 물리적 인프라 설치 프로세스는 수십 년 동안 학습하고 표준화하고 있는 분야입니다.

- 엔지니어링 또는 IT 팀은 일반적으로 요청자입니다.

- 다양한 재무 팀은 승인자 및 조달자의 역할을 합니다.
- 운영 팀은 바로 사용할 수 있는 인프라를 래킹, 스태킹 및 전달합니다.



클라우드를 도입하면 인프라 조달 및 사용 시 더 이상 종속성의 사슬에 얽매이지 않아도 됩니다. 클라우드 모델에서 기술 팀과 제품 팀은 구축자가 아니라 제품의 운영자와 소유자가 되며, 조달과 배포를 비롯하여 과거에는 재무 팀과 운영 팀과 관련이 있던 작업 대부분을 담당합니다.

클라우드 리소스를 프로비저닝하는 데는 계정과 올바른 권한 집합만 있으면 됩니다. 또한 IT 및 재무 위험을 줄여줍니다. 즉, 팀에서 몇 번만 클릭하거나 API를 호출하기만 하면 유휴 또는 필요 없는 클라우드 리소스를 종료할 수 있습니다. 게다가 기술 팀이 더 빠르게 혁신할 수 있도록 하여, 민첩하게 실험

을 시작하고 분석하는 능력이 향상됩니다. 클라우드 사용 시 다양한 측면이 자본 예산과 예측 관점에서 예측 가능성에 영향을 미칠 수 있지만 클라우드를 이용하여 조직은 과다 프로비저닝 비용을 줄이고 보수적인 과소 프로비저닝과 관련된 기회 비용을 절감할 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. 재무 팀은 IT 가치 스토리를 생성하고 널리 공유하기 위해 기술 팀과 협력해 어떻게 기술 비용이 비즈니스 성과와 연결되는지 비즈니스 팀이 이해할 수 있도록 도와야 합니다. 이런 관점에서 기술에 대한 지출은 비용이 아니라 투자로 인식됩니다. 클라우드와 온프레미스 운영 사이에는 근본적인 차이(예: 사용량 변경 속도, 사용량에 따른 요금, 계층화된 요금, 요금 모델, 세부 결제 및 사용량 정보)가 있으므로 재무 조직은 클

라우드 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 수준에 관한 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 투입하여 구매 시 약정을 통해 할인을 받는 방법(예: 예약형 인스턴스 또는 AWS 절감형 플랜)과 클라우드를 사용하여 조직 성장을 지원하는 방법에 대한 전략적 자문 역할을 할 수 있습니다.
- 클라우드에서 기존의 지급 계정 및 구매 프로세스를 함께 사용할 수 있습니다.
- 재무 팀과 기술 팀이 공동으로 향후 AWS 비용과 사용량을 예측하여 조직 예산을 조정하고 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 예약형 인스턴스 또는 절감형 플랜 등의 장기 구매 옵션을 평가할 때는 이러한 클라우드 관련 지식이 반드시 필요합니다.
- 엔지니어링 팀: 재무 팀과 기술 팀 간의 파트너십 확립은 엔지니어가 클라우드 재무 관리(CFM)에 대한 조치를 취할 수 있도록 권장하는 비용 인식 문화를 구축하는 데 반드시 필요합니다. CFM 또는 재무 운영 실무자와 재무 팀에 발생하는 일반적인 문제 중 하나는 엔지니어가 클라우드의 전체 비즈니스를 이해하고 모범 사례를 따르며 권장 조치를 취하도록 하는 것입니다.
- 서드파티: 조직에서 서드파티(예: 컨설턴트 또는 도구)를 사용하는 경우 이러한 업체가 재무 목표에 부합하는지 확인해야 하며 참여 모델 및 투자 수익률(ROI)을 통해 이러한 일치를 증명할 수 있습니다. 서드파티는 일반적으로 관리 대상 워크로드 보고와 분석을 수행하며 설계 대상 워크로드의 비용을 분석합니다.

CFM을 구현하고 성공을 달성하려면 재무, 기술 및 비즈니스 팀 간의 협업과 조직 전반에서 클라우드 비용이 전달되고 평가되는 방식의 변화가 필요합니다. 엔지니어링 팀을 포함시켜 이들이 모든 단계에서 이러한 비용과 사용 논의에 참여하고 모범 사례를 따라 합의된 조치를 취할 수 있도록 장려합니다.

구현 단계

- 주요 구성원 정의: 재무 팀과 기술 팀의 모든 관련 구성원이 파트너십에 참여하는지 확인하세요. 재무 팀의 관련 구성원은 클라우드 청구서와 관련하여 업무를 수행하는 사람들입니다. 대개 CFO, 재무 관리자, 재무 계획자, 비즈니스 분석가, 구매, 소싱 담당자가 여기에 해당합니다. 기술 구성원은 대개 제품 및 애플리케이션 소유자, 기술 관리자 및 클라우드에 구축된 모든 팀의 담당자가 됩니다. 다른 구성원으로는 제품 사용에 영향을 주는 마케팅과 같은 사업부 소유자와 목표 및 메커니즘에 부합하도록 하고 보고를 지원하는 컨설턴트와 같은 서드파티 등이 있을 수 있습니다.
- 토롤 주제 정의: 팀 간에 공통된 주제나 공동의 이해가 필요한 주제를 정의합니다. 발생한 시점부터 청구서가 지불될 때까지 비용을 추적합니다. 관련된 모든 구성원과 적용해야 하는 조직의 프로세스를 기록합니다. 각 단계 또는 이들이 거치는 프로세스와 사용 가능한 요금 모델, 계층화된 요금, 할인 모델, 예산 책정, 재무 요구 사항 등의 관련 정보를 파악합니다.
- 정기적인 주기 설정: 재무 팀과 기술 팀 간의 파트너십을 구축하려면 일치된 방향을 수립하고 유지하기 위해 정기적인 소통 주기를 확립해야 합니다. 그룹은 목표 및 지표에 따라 정기적으로 모여야 합니다. 일반적인 주기는 조직의 상태를 검토하고 현재 실행 중인 프로그램을 검토한 후 전반적인 재무 및 최적화 지표를 검토하는 것입니다. 그런 다음 주요 워크로드가 더 자세히 보고됩니다.

리소스

관련 문서:

- AWS 뉴스 블로그

COST01-BP03 클라우드 예산 및 예측 수립

클라우드 비용 및 사용량의 매우 가변적인 특성에 맞게 기존 조직의 예산 책정 및 예측 프로세스를 조정합니다. 프로세스는 추세나 비즈니스 동인을 기반으로 하는 알고리즘 또는 둘의 조합을 사용하여 동적으로 수행되어야 합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

기존의 온프레미스 IT 환경에서 고객은 고정 비용을 계획하는 데 어려움을 겪는 경우가 많습니다. 고정 비용은 드물게 변동하며 보통은 피크 수요를 충족하기 위해 새 IT 하드웨어와 서비스를 구입하는 경우입니다. 이와 대조적으로, AWS 클라우드는 고객이 실제 IT 및 비즈니스 요구 사항에 따라 사용하는 리

소스에 대해 비용을 지불하는 다른 접근 방식을 채택합니다. 클라우드 환경에서 수요는 월별, 일별 또는 시간별로 변동될 수 있습니다.

클라우드를 사용하면 효율성, 속도, 민첩성이 향상되어 비용 및 사용 패턴의 변동성이 커집니다. 워크로드 효율성 향상이나 새로운 워크로드 및 기능의 배포에 따라 비용이 감소하거나 때때로 증가할 수 있습니다. 확장되는 고객 기반에 맞춰 워크로드가 확장됨에 따라 리소스 접근성이 증대되어 클라우드 사용량과 비용도 함께 증가합니다. 클라우드 서비스의 이러한 유연성은 비용 및 예측으로 확장되어 어느 정도의 탄력성을 제공합니다.

이처럼 변화하는 비즈니스 요구 사항 및 수요 동인과 긴밀하게 연계하고 가능한 가장 정확한 계획을 세우는 것이 중요합니다. 기존의 조직 예산 프로세스는 이러한 변동성을 수용하도록 조정되어야 합니다.

새 워크로드의 비용을 예측하면서 비용 모델링을 고려하세요. 비용 모델링은 예상 클라우드 비용에 대한 기본 이해를 생성하여 총 소유 비용(TCO), 투자 수익률(ROI) 및 기타 재무 분석을 수행하고, 이해관계자와 함께 타겟 및 기대치를 설정하고, 비용 최적화 기회를 식별하는 데 도움이 됩니다.

조직은 비용 정의와 수락된 그룹화를 이해해야 합니다. 얼마나 세부적으로 예측할 것인지는 조직의 구조와 내부 워크플로에 따라 달라질 수 있습니다. 특정 요구 사항 및 조직 설정에 맞는 세부 수준을 선택하세요. 예측이 어느 수준에서 수행되는지 이해하는 것이 중요합니다.

- 관리 계정 또는 AWS Organizations 수준: 관리 계정은 AWS Organizations를 생성할 때 사용한 계정을 말합니다. 조직에는 기본적으로 관리 계정이 하나만 있습니다.
- 연결 계정 또는 구성원 계정: Organizations의 계정은 AWS 리소스를 포함하는 표준 AWS 계정으로, 이러한 리소스에 액세스할 수 있는 자격 증명입니다.
- 환경: 환경은 애플리케이션 버전을 실행 중인 AWS 리소스 모음입니다. 연결된 계정 또는 구성원 계정을 여러 개 사용하여 하나의 환경을 만들 수 있습니다.
- 프로젝트: 프로젝트는 정해진 기간에 달성해야 하는 정해진 목표 또는 작업의 조합입니다. 예측 과정에서 프로젝트 수명 주기를 고려하는 것이 중요합니다.
- AWS 서비스: 예측을 위해 AWS 서비스를 그룹화할 수 있는 컴퓨팅 또는 스토리지 서비스와 같은 그룹 또는 범주입니다.
- 사용자 지정 그룹화: 사업부, 비용 센터, 팀, 비용 할당 태그, 비용 범주, 연결 계정 또는 이들의 조합 등 조직의 요구 사항에 따라 사용자 지정 그룹을 생성할 수 있습니다.

사용 비용에 영향을 미칠 수 있는 비즈니스 동인을 식별하고 각 동인을 개별적으로 예측하여 예상 사용량을 미리 계산하세요. 일부 동인은 조직 내 IT 및 제품 팀과 연관된 것일 수 있습니다. 영업, 마케팅 및 비즈니스 리더는 마케팅 이벤트, 프로모션, 지리적 확장, 합병 및 인수 등의 다른 비즈니스 동인을 잘 알고 있으므로 이들과 협업하고 이러한 수요 동인을 모두 고려하는 것도 중요합니다.

AWS Cost Explorer를 사용하면 정의된 미래의 시간 범위에서 과거 지출을 기준으로 추세 기반 예측을 수행할 수 있습니다. AWS Cost Explorer의 예측 엔진은 비용 유형(예: 예약형 인스턴스)을 기준으로 과거 데이터를 구분하고 기계 학습과 규칙 기반 모델을 결합하여 모든 비용 유형 전반에서 비용을 개별적으로 예측합니다.

예측 프로세스를 수립하고 모델을 구축한 후에는 AWS Budgets를 사용하여 기간, 반복 주기 또는 금액(고정 또는 가변)을 지정하고 서비스, AWS 리전 및 태그 등과 같은 필터를 추가해 사용자 지정 예산을 세부 수준에서 설정할 수 있습니다. 예산은 일반적으로 1년 단위로 준비되고 고정되어 있으며, 관련된 모든 사람이 엄격하게 준수해야 합니다. 반면 예측은 더 유연하여 연중 재조정이 가능하며 1년, 2년 또는 3년 기간에 걸쳐 동적으로 추정할 수 있습니다. 예산과 예측은 모두 다양한 기술 및 비즈니스 이해관계자 간의 재정적 기대치를 설정하는 데 중요한 역할을 합니다. 정확한 예측 및 구현을 통해 처음부터 비용 프로비저닝을 직접 담당하는 이해관계자가 책임을 지게 되며 아울러 전반적인 비용 인식도 높일 수 있습니다.

기존 예산의 성과에 대한 최신 정보를 확보하기 위해 AWS Budgets 보고서를 생성하고 자신과 이해관계자에게 해당 보고서가 이메일로 정기적으로 전송되도록 예약할 수 있습니다. 또한 사후 대응적인 실제 비용 또는 예상 비용을 기반으로 AWS Budgets 알림을 생성하여 잠재적 비용 초과에 대한 완화 조치를 구현할 시간을 확보할 수 있습니다. 비용이나 사용량이 실제로 특정 수준을 초과하거나 예산 금액을 초과할 것으로 예상되는 경우 알림을 받을 수 있습니다.

추세 기반 알고리즘(예: 기간별 비용을 입력으로 사용)을 사용하거나, 동적이고 가변적인 지출 환경에 적합한 동인 기반 알고리즘(예: 신제품 출시, 리전별 확장 또는 워크로드를 위한 새로운 환경)을 사용하여 기존의 예산 및 예측 프로세스를 보다 동적으로 조정하세요. Cost Explorer 또는 다른 도구를 사용하여 추세 기반 예측을 결정한 후에는 AWS Pricing Calculator를 사용하여 예상 사용량(트래픽, 초당 요청 또는 필요한 Amazon EC2 인스턴스)을 기반으로 AWS 사용 사례와 향후 비용을 추정할 수 있습니다.

이러한 예측 계산 및 추정을 기반으로 예산을 수립해야 하므로 해당 예측의 정확성을 추적합니다. 통합 클라우드 비용 예측의 정확성과 효과를 모니터링합니다. 예측과 비교하여 실제 지출을 정기적으로 검토하고 필요에 따라 조정하여 예측의 정확도를 개선하세요. 예측 편차를 추적하고 보고된 변동에 대한 근본 원인 분석을 수행하여 예측을 실행하고 조정합니다.

COST01-BP02 재무 팀과 기술 팀 간의 파트너십 수립에 언급된 것처럼 일관성을 위해 모두 동일한 도구 또는 프로세스를 사용하도록 하려면 IT 팀, 재무 팀 및 기타 이해관계자 간에 파트너십을 확립하고 주기적으로 소통하는 것이 중요합니다. 예산을 변경해야 하는 경우 소통 주기를 늘려 보다 신속하게 변화에 대응할 수 있습니다.

구현 단계

- 조직 내 비용 언어 정의: 조직 내에서 다차원 및 그룹화를 사용하여 공통 AWS 비용 언어를 생성합니다. 이해관계자가 예측 세부 수준, 가격 책정 모델, 비용 예측 수준을 이해하도록 하세요.
- 추세 기반 예측 분석: AWS Cost Explorer 및 Amazon Forecast와 같은 추세 기반 예측 도구를 사용합니다. 서비스, 계정, 태그 및 비용 범주와 같은 여러 차원에서 사용 비용을 분석할 수 있습니다.
- 동인 기반 예측 분석: 비즈니스 동인이 클라우드 사용량에 미치는 영향을 파악하고 각 동인에 대해 개별적으로 예측하여 예상 사용 비용을 미리 계산합니다. 사업부 책임자 및 이해관계자와 긴밀히 협력하여 새로운 동인에 미치는 영향을 이해하고 예상되는 비용 변경을 계산하여 정확한 예산을 정의하세요.
- 기존 예측 및 예산 프로세스 업데이트: 추세 기반, 비즈니스 동인 기반 또는 이 두 예측 방법의 조합 등 채택된 예측 방법을 기반으로 예측 및 예산 프로세스를 정의합니다. 예산은 현실적이고 예측에 기반하여 계산되어야 합니다.
- 경고 및 알림 구성: AWS Budgets 알림 및 비용 이상 탐지를 사용하여 경고 및 알림을 받을 수 있습니다.
- 주요 이해관계자와 함께 정기 검토 수행: 예를 들어 IT, 재무, 플랫폼 팀 및 기타 비즈니스 영역의 이해관계자와 함께 비즈니스 방향과 사용량의 변화에 맞춰 조정합니다.

리소스

관련 문서:

- AWS Cost Explorer
- AWS Cost and Usage Report
- Forecasting with Cost Explorer
- QuickSight Forecasting
- AWS Budgets

관련 비디오:

- How can I use AWS Budgets to track my spending and usage
- AWS Cost Optimization Series: AWS Budgets

관련 예제:

- Understand and build driver-based forecasting
- How to establish and drive a forecasting culture
- How to improve your cloud cost forecasting
- Using the right tools for your cloud cost forecasting

COST01-BP04 조직의 프로세스에서 비용 인식 구현

비용 인식을 구현하고 사용량에 영향을 미치는 신규 또는 기존 프로세스에 투명성과 책임을 더하며 비용 인식에 기존 프로세스를 활용합니다. 직원 교육에 비용 인식을 구현합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

신규 및 기존 조직 프로세스 안에 비용 인식을 구현해야 합니다. 이는 다른 모범 사례를 구현하기 위한 기본 전제 조건 중 하나입니다. 가능한 경우 기존 프로세스를 재사용하고 수정하는 것이 좋습니다. 이렇게 하면 민첩성과 속도에 미치는 영향을 최소화할 수 있습니다. 재무 및 비즈니스 이해관계자를 위해 비용 인식을 높이고 효율성 핵심 성과 지표(KPI)를 정립하기 위해 비즈니스 및 재무 팀의 의사 결정권자와 기술 팀에 클라우드 비용을 보고합니다. 다음은 워크로드에 비용 인식을 구현하는 데 도움이 되는 권장 사항입니다.

- 변경 관리에 변경이 재정에 미치는 영향을 정량화하는 비용 측정이 포함되어 있는지 확인하세요. 비용 측정을 포함하면 비용 관련 문제를 사전에 해결하고 비용 절감을 강조할 수 있습니다.
- 비용 최적화가 운영 역량의 핵심 구성 요소인지 확인하세요. 예를 들어 기존 인시던트 관리 프로세스를 활용하여 비용 및 사용량 이상 또는 비용 초과의 근본 원인을 조사하고 식별할 수 있습니다.
- 자동화 또는 도구를 사용하여 비용 절감 및 비즈니스 가치 실현을 가속화합니다. 구현 비용을 고려할 때는 투자 수익(ROI) 구성 요소를 포함하도록 대화를 구성하여 시간 또는 금전 투자의 당위성을 설명합니다.
- 약정 기반 구입 옵션, 공유 서비스 및 마켓플레이스 구입 비용을 비롯하여 클라우드 비용에 대한 내역 확인(Showback) 또는 결제 처리를 구현하여 클라우드 비용을 할당해 비용 인식이 최대한 반영된 클라우드 사용을 촉진합니다.
- 조직 전체에서 비용 인식 교육을 포함하도록 기존 교육 및 개발 프로그램을 확대하세요. 여기에 지속적인 교육 및 자격증을 포함하는 것이 좋습니다. 이렇게 하면 조직에서 비용 및 사용량을 자체적으로 관리할 수 있는 역량을 갖출 수 있습니다.
- 무료로 제공되는 AWS 기본 도구(예: AWS Cost Anomaly Detection, AWS Budgets, AWS Budgets 보고서)를 활용합니다.

조직에서 일관되게 클라우드 재무 관리(CFM) 사례를 도입하면 이러한 행동이 작업 및 의사 결정 과정에 정착하게 됩니다. 그 결과, 새로운 클라우드 애플리케이션을 설계하는 개발자부터 새로운 클라우드 투자에 대한 ROI를 분석하는 재무 관리자까지 비용 인식 문화가 더 잘 자리 잡게 됩니다.

구현 단계

- 관련 조직 프로세스 파악: 각 조직 단위에서 프로세스를 검토하고 비용 및 사용량에 영향을 미치는 프로세스를 파악합니다. 리소스가 생성 또는 종료되게 하는 모든 프로세스는 검토를 위해 포함해야 합니다. 인시던트 관리 및 교육과 같이 비즈니스에서 비용 인식을 지원할 수 있는 프로세스를 찾습니다.
- 스스로 지속 가능한 비용 인식 문화 확립: 관련된 모든 이해관계자가 변경의 원인 및 영향을 비용으로 인식하도록 하여 클라우드 비용을 이해할 수 있도록 합니다. 그러면 조직에서는 혁신을 위해 스스로 지속 가능한 비용 인식 문화를 확립할 수 있습니다.
- 비용 인식으로 프로세스 업데이트: 각 프로세스는 비용을 인식하도록 수정됩니다. 이 프로세스에서는 비용이 미치는 영향의 평가와 같은 추가 사전 점검이나 비용과 사용량에서 예상한 변화가 발생했는지 검증하는 사후 점검이 필요할 수 있습니다. 비용 및 사용량에 대한 항목을 포함하도록 교육 및 인시던트 관리와 같은 지원 프로세스를 확장할 수 있습니다.

도움을 받으려면 계정 팀을 통해 CFM 전문가에게 문의하거나 아래 리소스 및 관련 문서를 살펴보세요.

리소스

관련 문서:

- AWS 클라우드 금융 관리

관련 예제:

- Strategy for Efficient Cloud Cost Management
- Cost Control Blog Series #3: How to Handle Cost Shock
- A Beginner’s Guide to AWS Cost Management

COST01-BP05 비용 최적화 보고 및 알림

클라우드 예산을 설정하고 사용 중 이상을 감지하는 메커니즘을 구성하세요. 미리 정의된 목표에 대해 비용 및 사용량 알림을 받을 수 있도록 관련 도구를 구성하고 사용량이 해당 목표를 초과할 경우 알림

을 받을 수 있습니다. 정기 회의를 통해 워크로드의 비용 효율성을 분석하고 비용 인식을 제고할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 지침

조직 내의 비용 및 사용량 최적화를 정기적으로 보고해야 합니다. 비용 성과를 논의하기 위한 전용 세션을 구현하거나 워크로드에 대한 정기적인 운영 보고 주기에 비용 최적화를 포함할 수 있습니다. 서비스와 도구를 사용하여 비용 성과를 정기적으로 모니터링하고 비용 절감 기회를 모색할 수 있습니다.

AWS Cost Explorer를 사용하여 여러 필터 및 세분화로 비용 및 사용량을 볼 수 있습니다. 여기에서는 서비스별 또는 계정별 비용, 일일 비용 또는 마켓플레이스 비용과 같은 대시보드 및 보고서를 제공합니다. AWS Budgets 보고서를 통해 구성된 예산을 기준으로 비용 및 사용량의 진행 상태를 추적합니다.

AWS Budgets를 통해 사용자 지정 예산을 설정하여 임곗값을 초과한 경우 비용 및 사용량을 추적하고 이메일로 받은 알림 또는 Amazon Simple Notification Service(SNS) 알림에 신속하게 대응할 수 있습니다. 기본 예산 기간을 매일, 매월, 매분기 또는 매년으로 설정하고 특정 예산 한도를 생성하여 실제 또는 예측 비용 및 사용량이 예산 임곗값을 향해 어떻게 진행되고 있는지 알림을 받을 수 있습니다. 또한 알림 및 해당 알림에 대한 작업이 자동으로 실행되거나 예산 목표가 초과된 경우에는 승인 프로세스를 통해 실행되도록 설정할 수도 있습니다.

예기치 못한 비용 및 사용량의 변화에 신속하게 대응할 수 있도록 비용 및 사용량에 대한 알림을 구현합니다. AWS Cost Anomaly Detection를 사용하면 갑작스럽게 발생하는 비용을 줄이고 혁신 속도를 늦추지 않고 제어 기능을 개선할 수 있습니다. AWS Cost Anomaly Detection에서는 비정상적인 비용 및 근본 원인을 파악하여 갑작스러운 비용 청구 위험을 줄여줍니다. 간단한 3단계를 수행하면 맥락에 맞는 고유한 모니터링 시스템을 구축하고 이례적인 비용이 감지된 경우 알림을 받을 수 있습니다.

QuickSight와 AWS Cost and Usage Report(CUR) 데이터를 함께 사용하면 세분화된 데이터로 사용자 정의 기반의 보고를 제공할 수 있습니다. QuickSight에서는 보고서 일정을 수립하고 과거 비용 및 사용량 또는 비용 절감 기회에 대한 비용 보고서를 이메일을 통해 정기적으로 받을 수 있습니다. QuickSight에 구축된 Cost Intelligence Dashboard(CID) 솔루션을 확인해 보세요. 이 솔루션은 고급 가시성을 제공합니다.

AWS Trusted Advisor를 사용합니다. 이는 프로비저닝된 리소스가 비용 최적화를 위한 AWS 모범 사례에 부합하는지 여부를 확인하는 지침을 제공합니다.

세부적인 비용 및 사용량과 비교한 시각적 그래프를 통해 절감형 플랜 권장 사항을 확인해 보세요. 시간 단위 그래프는 온디맨드 지출을 권장 절감형 플랜 약정과 함께 보여주며 예상 절감액, 절감형 플랜

적용 범위 및 절감형 플랜 사용률에 대한 인사이트를 제공합니다. 이를 통해 조직은 지출을 분석하기 위한 모델을 구축하는 데 시간과 리소스를 투자하지 않고도 절감형 플랜이 각 지출 시간에 적용되는 방식을 이해할 수 있습니다.

안정 상태 워크로드, 유휴 리소스 및 사용률이 낮은 리소스와 관련된 비용의 절감을 위해 AWS Cost Explorer에서 절감형 플랜, 예약형 인스턴스 및 Amazon EC2의 적정 크기 조정 권장 사항이 포함된 보고서를 주기적으로 생성합니다. 배포되는 리소스에 대한 클라우드 낭비와 관련된 비용을 파악하여 회수합니다. 크기가 잘못 지정된 리소스를 생성하거나 예상한 패턴 대신 다른 사용 패턴이 관찰되는 경우 클라우드 낭비가 발생합니다. AWS 모범 사례에 따라 낭비를 줄이거나 계정 팀 및 파트너에게 도움을 요청하여 클라우드 비용을 최적화 및 절약합니다.

더 나은 리소스 구매 옵션을 선택할 수 있도록 정기적으로 보고서를 생성하여 워크로드에 대한 단위 비용을 절감하세요. 절감형 플랜, 예약형 인스턴스 또는 Amazon EC2 스팟 인스턴스 등과 같은 구입 옵션은 내결함성 워크로드에서 가장 크게 비용을 절감할 수 있고 이해관계자(비즈니스 소유자, 재무 팀 및 기술 팀)가 이러한 약정 논의에 참여할 수 있습니다.

클라우드의 총 소유 비용(TCO)을 절감하는 데 도움이 될 수 있는 기회 또는 새로운 릴리스 발표 내용이 포함된 보고서를 제공합니다. 추가 비용 절감을 위해 새로운 서비스, 리전, 기능, 솔루션 또는 새로운 방법을 채택합니다.

구현 단계

- AWS Budgets 구성: 워크로드의 모든 계정에서 AWS Budgets를 구성합니다. 태그를 사용하여 전체 계정 지출에 대한 예산과 워크로드에 대한 예산을 설정합니다.
  - Well-Architected Labs: Cost and Governance Usage
- 비용 최적화 보고: 워크로드의 효율성을 논의하고 분석하기 위한 규칙적인 주기를 설정합니다. 설정된 지표를 사용하여 달성된 지표와 지표 달성에 든 비용을 보고합니다. 부정적인 추세가 있다면 찾아서 수정하고 조직 전체에 장려할 수 있는 긍정적인 추세를 찾습니다. 보고에는 클라우드 지출과 관련된 애플리케이션 팀 및 책임자, 재무 책임자 및 주요 의사 결정권자가 포함되어야 합니다.

리소스

관련 문서:

- AWS Cost Explorer
- AWS Trusted Advisor
- AWS Budgets

- AWS Cost and Usage Report
- AWS Budgets 모범 사례
- Amazon S3 분석

관련 예제:

- Key ways to start optimizing your AWS cloud costs

COST01-BP06 사전 예방적 비용 모니터링

도구 및 대시보드를 구현하여 워크로드에 대한 비용을 사전에 모니터링합니다. 알림을 받을 때만 비용 및 범주를 살펴보지 말고 구성된 도구 또는 바로 사용 가능한 도구와 관련된 비용을 정기적으로 검토합니다. 비용을 사전에 모니터링하고 분석하면 긍정적인 추세를 파악하여 조직 전체에서 해당 추세를 촉진할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 지침

예외 또는 이상이 있을 뿐만이 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. 가시성이 높은 대시보드를 사용하면 성공적인 결과를 적극적으로 홍보하고 조직 전체에 이를 구현할 수 있습니다.

AWS Cost Explorer 또는 기타 대시보드(예: Amazon QuickSight)를 사용하는 매일 또는 잦은 루틴을 만들어 사전에 비용을 살펴보고 분석할 수 있습니다. AWS 계정 수준, 워크로드 수준 또는 특정 AWS 서비스 수준에서 그룹화와 필터링을 사용하여 AWS 서비스 사용량과 비용을 분석하고 해당 사용량과 비용이 예상된 것인지 확인합니다. 시간 수준 및 리소스 수준 세분화와 태그를 사용하여 가장 많이 사용하는 리소스에서 발생하는 비용을 필터링하고 식별할 수 있습니다. 또한 Cost Intelligence Dashboard(AWS Solutions Architects가 구축한 Amazon QuickSight 솔루션)를 사용하여 고유한 보고서를 작성하고 실제 비용 및 사용량과 예산을 비교할 수 있습니다.

구현 단계

- 비용 최적화 보고: 워크로드의 효율성을 논의하고 분석하기 위한 규칙적인 주기를 설정합니다. 설정된 지표를 사용하여 달성된 지표와 지표 달성에 든 비용을 보고합니다. 부정적인 추세가 있다면 찾아서 수정하고 조직 전체에 장려할 수 있는 긍정적인 추세를 찾습니다. 보고에는 애플리케이션 팀 및 소유자, 재무 및 경영진의 담당자가 포함되어야 합니다.

- 비용 초과 가능성을 예방하기 위해 적시에 조치를 취할 수 있도록 비용과 사용량에 대해 일 단위 AWS Budgets 생성 및 활성화: AWS Budgets를 사용하면 알림을 구성할 수 있어 예산 유형이 미리 구성된 임곗값을 벗어난 경우 알림을 받을 수 있습니다. AWS Budgets을 제대로 활용하려면 기대 비용과 사용량을 한도로 설정해 두는 것이 좋습니다. 그러면 예산을 초과하는 모든 항목을 초과 지출로 간주할 수 있습니다.
- 비용 모니터링을 위한 AWS Cost Anomaly Detection 생성: AWS Cost Anomaly Detection에서는 고급 기계 학습 기술을 사용하여 비정상적인 지출과 근본 원인을 식별하므로 빠르게 조치를 취할 수 있습니다. 또한 평가하고 싶은 지출 세그먼트(예: 개별 AWS 서비스, 구성원 계정, 비용 할당 태그 및 비용 범주)를 정의하는 비용 모니터링을 구성할 수 있고 알림을 받는 경우와 위치 그리고 방법을 설정할 수 있습니다. 각 모니터링에 이름, 비용 영향 임곗값, 각 구독에 대한 알림 빈도(개별 알림, 일일 요약, 주별 요약)를 비롯하여 비즈니스 소유자와 기술 팀에 대한 여러 알림 구독을 첨부할 수 있습니다.
- AWS Cost Explorer를 사용하거나 AWS Cost and Usage Report(CUR) 데이터를 Amazon QuickSight 대시보드와 통합하여 조직의 비용 시각화: AWS Cost Explorer에는 시간에 따라 AWS 비용과 사용량을 시각화하고 이해하며 관리할 수 있는 사용하기 쉬운 인터페이스가 있습니다. Cost Intelligence Dashboard는 사용자 지정 및 액세스 가능한 대시보드로, 고유한 비용 관리 및 최적화 도구의 기반을 만드는 데 유용합니다.

리소스

관련 문서:

- AWS Budgets
- AWS Cost Explorer
- Daily Cost and Usage Budgets
- AWS Cost Anomaly Detection

관련 예제:

- AWS Cost Anomaly Detection Alert with Slack

COST01-BP07 새로운 서비스 릴리스로 최신 상태 유지

정기적으로 전문가 또는 AWS 파트너의 상담을 받아 더 저렴한 가격을 제공하는 서비스와 기능을 고려합니다. AWS 블로그와 기타 정보 출처를 검토합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

AWS는 새로운 기능을 지속적으로 추가하고 있습니다. 따라서 최신 기술을 활용하여 더욱 신속하게 실험하고 혁신할 수 있습니다. 새로운 AWS 서비스와 기능을 구현하여 워크로드의 비용 효율성을 높일 수 있습니다. AWS 비용 관리, AWS 뉴스 블로그, AWS 비용 관리 블로그 및 AWS의 새로운 소식을 주기적으로 검토하여 새로운 서비스 및 기능 릴리스에 대한 정보를 확인하세요. 새로운 소식 게시물은 모든 AWS 서비스, 기능 및 리전 확장이 발표되면 해당 내용을 간략하게 안내합니다.

구현 단계

- 블로그 구독: AWS 블로그 페이지를 방문하고 새로운 소식 블로그와 기타 관련 블로그를 구독합니다. 이메일 주소를 사용하여 커뮤니케이션 기본 설정 페이지에 가입할 수 있습니다.
- AWS 뉴스 구독: 정기적으로 AWS 뉴스 블로그와 AWS의 새로운 소식을 검토하여 새로운 서비스 및 기능 출시에 대한 정보를 확인하세요. RSS 피드를 구독하거나 이메일을 사용하여 발표 및 릴리스 소식을 팔로우할 수 있습니다.
- AWS 요금 인하 준수: 스케일 조정에서 얻은 경제적 효율성을 고객에게 되돌려 드리기 위해 AWS에서 실시하는 모든 서비스의 정기적인 가격 인하는 표준으로 자리를 잡았습니다. 2023년 9월 20일 기준으로 AWS는 2006년 이후 134회 가격을 인하했습니다. 가격 때문에 결정을 보류하고 있다면 가격 인하와 새로운 서비스 통합을 적용한 후 다시 검토할 수 있습니다. Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스를 포함한 이전의 가격 인하 노력에 대한 내용은 price-reduction category of the AWS News Blog에서 확인할 수 있습니다.
- AWS 이벤트 및 회의: 현지 AWS 서밋과 해당 지역의 다른 조직과의 모든 현지 회의에 참석합니다. 직접 참석할 수 없는 경우 가상 이벤트에 참석하여 AWS 전문가와 기타 고객의 비즈니스 사례에 대해 자세히 들어보세요.
- 계정 팀과 회의: 계정 팀과 정기적인 회의를 예약하고 이들과 만나 업계 동향과 AWS 서비스에 대해 논의합니다. 계정 관리자, 솔루션 아키텍트 및 지원 팀과 이야기합니다.

리소스

관련 문서:

- AWS 비용 관리
- AWS의 새로운 소식
- AWS 뉴스 블로그

관련 예제:

- Amazon EC2 – 15 Years of Optimizing and Saving Your IT Costs
- AWS 뉴스 블로그 - 가격 인하

COST01-BP08 비용 인식 문화 조성

조직 전체에 비용 인식 문화를 조성하기 위한 변화를 주도하거나 이러한 프로그램을 구현합니다. 소규모로 시작한 후 역량이 커지고 조직의 클라우드 사용이 증가함에 따라 대규모 프로그램을 구현하는 것이 좋습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 지침

비용 인식 문화가 조성되면 조직 전체에서 유기적이고 분산된 방식으로 수행되는 모범 사례를 통해 비용 최적화 및 클라우드 비용 관리(재무 운영, 클라우드 혁신 센터, 클라우드 운영 팀 등)를 조정할 수 있습니다. 비용 인식을 통해 엄격한 하향식의 중앙 집중식 접근 방식에 비해 최소한의 노력으로 조직 전체의 역량을 강화할 수 있습니다.

클라우드 컴퓨팅에서 특히, 클라우드 컴퓨팅의 1차 비용 요인에 대한 비용 인식을 통해 팀에서는 모든 변경의 예상 결과를 비용 관점에서 이해할 수 있습니다. 클라우드 환경에 액세스하는 팀은 가격 모델과 전통적인 온프레미스 데이터 센터와 클라우드 컴퓨팅 간의 차이를 알고 있어야 합니다.

비용 인식 문화의 주요 이점은 기술 팀이 상황이 발생한 후 필요에 따라 비용 최적화를 대응적으로 수행하는 대신 지속적으로 사전에 비용을 최적화한다는 점입니다(예: 비용은 새로운 워크로드를 설계하거나 기존 워크로드를 변경할 때 기술적 요구 사항이 아닌 것으로 간주됨).

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

- 비용 관점에서 역할과 미치는 영향을 파악할 수 있도록 엔지니어링 팀 내 가시성 확보 및 인식 확립.
- 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.
- 필요한 만큼만 지불할 수 있도록 변경에 대한 비즈니스 요구 사항과 아키텍처 인프라 또는 워크로드 구성에 대해 요청된 변경 사항이 미치는 비용 영향에 대해 질문하세요.

- 변경 계획자가 비용에 영향을 미치는 예상 변경 사항을 파악하고 비즈니스 결과를 비용 효율적으로 전달할 수 있도록 이해관계자가 확인하도록 하세요.

구현 단계

- 기술 팀에 클라우드 비용 보고: 재무 및 비즈니스 이해관계자를 위해 비용 인식을 높이고 효율성 KPI를 확립하기 위함입니다.
- 이해관계자 또는 팀원에게 계획된 변경에 대해 알림: 주간 변경 회의 중 계획된 변경 및 워크로드에 대한 비용 이점에 대해 논의할 수 있도록 소개 항목을 생성합니다.
- 계정 팀과의 만남 계정 팀과 정기적인 회의 주기를 확립하고 업계 동향과 AWS 서비스에 대해 논의합니다. 계정 관리자, 아키텍트 및 지원 팀과 이야기합니다.
- 성공 스토리 공유: 긍정적인 태도 및 비용 최적화 권장을 위해 워크로드, AWS 계정 또는 조직에 대한 비용 절감 성공 사례를 공유합니다.
- 교육: 기술 팀 또는 팀원이 AWS 클라우드에 대한 리소스 비용 인식의 교육을 받도록 합니다.
- AWS 이벤트 및 회의: 현지 AWS 서밋과 해당 지역의 다른 조직과의 모든 현지 회의에 참석합니다.
- 블로그 구독: AWS 블로그 페이지를 방문하여 새로운 소식 블로그 및 기타 관련 블로그를 구독하여 AWS에서 공유한 새로운 릴리스, 구현, 예제, 변경 사항을 팔로우해 보세요.

리소스

관련 문서:

- AWS 블로그
- AWS 비용 관리
- AWS 뉴스 블로그

관련 예제:

- AWS 클라우드 금융 관리

COST01-BP09 비용 최적화의 비즈니스 가치 정량화

비용 최적화의 비즈니스 가치를 정량화하면 조직에 대한 전체적인 이점을 이해할 수 있습니다. 비용 최적화는 필요한 투자이므로 비즈니스 가치를 정량화하면 이해관계자에게 투자 수익률을 설명할 수 있습니다. 비즈니스 가치를 정량화하면 향후 비용 최적화 투자에 대해 이해관계자로부터 더 많은 지지를

얻는 데 도움이 되며, 조직의 비용 최적화 활동에 대한 결과를 측정하는 프레임워크로 사용될 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

비즈니스 가치를 정량화한다는 것은 기업이 취하는 조치와 결정을 통해 얻는 이점을 측정하는 것을 의미합니다. 비즈니스 가치는 유형(예: 비용 절감 또는 수익 증대)일 수도 있고 무형(예: 브랜드 평판 향상 또는 고객 만족도 향상)일 수도 있습니다.

비용 최적화를 통해 비즈니스 가치를 정량화한다는 것은 더 효율적으로 지출하기 위한 노력을 통해 얻을 수 있는 가치나 혜택을 파악하는 것을 의미합니다. 예를 들어 회사에서 AWS에 워크로드를 배포하는 데 10만 USD를 지출하고 나중에 최적화한다면 품질이나 출력에 영향을 주지 않으면서 새로 드는 비용이 8만 USD에 불과합니다. 이 시나리오에서는 비용 최적화를 통해 비즈니스 가치를 정량화하면 2만 USD를 절감할 수 있습니다. 그러나 비용 절감뿐만 아니라 납품 시간 단축, 고객 만족도 향상 또는 비용 최적화 노력으로 인한 기타 지표 측면에서 가치를 정량화할 수도 있습니다. 이해관계자들은 비용 최적화의 잠재적 가치, 워크로드 최적화 비용, 수익 가치를 판단해야 합니다.

비용 최적화의 절감 효과를 보고하는 것에 더해 제공된 추가 가치를 정량화하는 것이 좋습니다. 비용 최적화의 이점은 일반적으로 각 비즈니스 결과에 대한 비용 절감이라는 측면에서 정량화됩니다. 예를 들어 비용을 줄이고 워크로드 결과 수준을 유지할 수 있는 절감형 플랜을 구매하는 경우 Amazon Elastic Compute Cloud(Amazon EC2) 비용 절감을 정량화할 수 있습니다. 유휴 Amazon EC2 인스턴스가 종료되거나 연결되지 않은 Amazon Elastic Block Store(Amazon EBS) 볼륨이 제거될 때 AWS에서 지출의 비용 감소를 정량화할 수 있습니다.

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

구현 단계

- 비즈니스 이점 평가: 지출 비용의 1 USD마다 얻을 수 있는 이점을 극대화하는 방식으로 AWS 클라우드 클라우드 비용을 분석하고 조정하는 프로세스입니다. 비즈니스 가치에 대한 고려 없이 비용을 절감하는 데에만 초점을 맞추지 말고 비용 최적화의 비즈니스 혜택과 투자 수익률을 고려하세요. 그러면 지출한 비용으로 더 많은 가치를 창출할 수 있습니다. 현명하게 지출하고 수익률이 가장 높은 분야에 투자와 지출을 하는 것이 관건입니다.
- 예상되는 AWS 비용 분석: 재무 관계자는 예측을 통해 다른 내부 및 외부 조직의 이해관계자의 기대치를 설정하고 조직의 재무 예측 기능을 개선할 수 있습니다. AWS Cost Explorer를 사용하여 비용 및 사용량을 예측할 수 있습니다.

리소스

관련 문서:

- AWS 클라우드 경제학
- AWS 블로그
- AWS 비용 관리
- AWS 뉴스 블로그
- Well-Architected 신뢰성 원칙 백서
- AWS Cost Explorer

관련 비디오:

- Unlock Business Value with Windows on AWS

관련 예제:

- Measuring and Maximizing the Business Value of Customer 360
- The Business Value of Adopting Amazon Web Services Managed Databases
- The Business Value of Amazon Web Services for Independent Software Vendors
- Business Value of Cloud Modernization
- The Business Value of Migration to Amazon Web Services

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.

주제

- 지배구조
- 비용 및 사용량 모니터링
- 리소스 폐기

지배구조

클라우드에서 비용을 관리하려면 다음과 같은 거버넌스 영역을 통해 사용량을 관리해야 합니다.

모범 사례

- COST02-BP01 조직 요구 사항에 따라 정책 개발
- COST02-BP02 목표 및 타겟 이행
- COST02-BP03 계정 구조 구현
- COST02-BP04 그룹 및 역할 구현
- COST02-BP05 비용 제어 기능 구현
- COST02-BP06 프로젝트 수명 주기 추적

COST02-BP01 조직 요구 사항에 따라 정책 개발

조직에서 리소스를 관리하는 방법을 정의하는 정책을 개발하고 정기적으로 검사합니다. 정책에서는 리소스 수명 주기 동안의 생성, 수정, 폐기를 포함한 리소스 및 워크로드의 비용 측면을 다루어야 합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

비용 및 사용량을 효과적으로 관리하고 비용 절감 기회를 파악하려면 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 행동으로 이어지고 낭비되는 요소를 줄여줍니다. 정확한 비용 및 사용량 모니터링을 통해 워크로드가 얼마나 최적화되었는지, 조직 단위 및 제품의 수익성이 어느 정도인지 이해할 수 있습니다. 이를 통해 조직 내에서 리소스를 할당할 위치에 대해 더 많은 정보를 활용하여 의사 결정을 내릴 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.

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. 정책에서는 리소스 수명 주기 동안의 생성, 수정, 폐기를 비롯하여 비용 또는 사용량과 관련된 워크로드의 모든 측면을 다루어야 합니다. 정책과 절차를 준수하고 클라우드 환경의 모든 변화에 대비하여 구현되었는지 확인합니다. IT 변경 관리 회의에서 계획된 변경 사항의 비용 영향, 즉 증가 및 감소 여부, 비즈니스 타당성, 예상 결과에 대해 질문합니다.

정책은 조직 전체에서 쉽게 이해할 수 있고 효과적으로 구현할 수 있을 만큼 단순해야 합니다. 또한 정책은 준수 및 해석하기 쉬워야 하고(그래야 정책이 사용될 수 있음) 구체적이어야 합니다(그래야 팀 간에 오해가 없음). 또한 당사의 메커니즘과 같이 정기적으로 점검하고 고객의 비즈니스 상황 또는 우선 순위가 바뀌면 정책이 상황에 맞지 않을 수 있으므로 업데이트해야 합니다.

사용할 지리적 리전, 리소스를 실행해야 하는 하루 중 시간 등과 같이 넓은 범위의 상위 수준 정책부터 시작합니다. 그런 다음 다양한 조직 단위 및 워크로드에 대한 정책을 점진적으로 구체화합니다. 가장 많이 사용하는 정책에는 사용할 수 있는 서비스와 기능(예: 테스트 및 개발 환경의 비교적 성능이 낮은 스토리지), 각 그룹에서 사용할 수 있는 리소스 유형(예: 개발 계정에서 사용할 수 있는 가장 큰 리소스 크기는 중간 규모), 리소스 사용 시간(임시, 단기 또는 특정 기간)이 있습니다.

정책 예제

다음은 비용 최적화에 중점을 둔 자체 클라우드 거버넌스 정책을 작성하기 위해 검토할 수 있는 샘플 정책입니다. 조직의 요구 사항과 이해관계자의 요청에 따라 정책을 조정해야 합니다.

- 정책 이름: 리소스 최적화 및 비용 절감 정책과 같은 명확한 정책 이름을 정의합니다.
- 목적: 이 정책을 사용해야 하는 이유와 예상되는 결과를 설명합니다. 이 정책의 목적은 비즈니스 요구 사항을 충족하기 위해 원하는 워크로드를 배포하고 실행하는 데 필요한 최소 비용이 있는지 확인하는 것입니다.
- 범위: 이 정책을 누가 언제 사용해야 하는지 명확하게 정의합니다. 예를 들어 DevOps X Team은 X 환경(프로덕션 또는 비프로덕션)에서 미국 동부 고객에게 이 정책을 사용해야 합니다.

정책 문

1. 워크로드의 환경 및 비즈니스 요구 사항(개발, 사용자 승인 테스트, 사전 프로덕션 또는 프로덕션)에 따라 us-east-1 또는 여러 개의 미국 동부 리전을 선택합니다.
2. 오전 6시~오후 8시(동부 표준시(EST)) 사이에 실행되도록 Amazon EC2 및 Amazon RDS 인스턴스를 예약합니다.
3. 8시간 동안 활동이 없으면 사용하지 않는 모든 Amazon EC2 인스턴스를 중지하고 24시간 동안 활동이 없으면 사용하지 않는 Amazon RDS 인스턴스를 중지합니다.
4. 비프로덕션 환경에서는 24시간 동안 활동이 없으면 사용하지 않는 모든 Amazon EC2 인스턴스를 종료합니다. Amazon EC2 인스턴스 소유자(태그 기준)에게 프로덕션 환경에서 중지된 Amazon EC2 인스턴스를 검토하도록 요청하고 사용하지 않을 경우 72시간 이내에 해당 Amazon EC2 인스턴스가 종료될 것임을 알립니다.
5. 일반 인스턴스 패밀리 및 크기(예: m5.large)를 사용한 다음 AWS Compute Optimizer를 사용하여 CPU 및 메모리 사용률에 따라 인스턴스 크기를 조정합니다.
6. Auto Scaling을 사용하여 우선순위를 지정해 트래픽에 따라 실행 중인 인스턴스 수를 동적으로 조정합니다.
7. 중요하지 않은 워크로드에는 스팟 인스턴스를 사용합니다.
8. 용량 요구 사항을 검토하여 예측 가능한 워크로드를 위한 절감형 플랜 또는 예약형 인스턴스를 약정하고 클라우드 재무 관리 팀에 알립니다.
9. Amazon S3 수명 주기 정책을 사용하여 자주 액세스하지 않는 데이터를 저렴한 스토리지 계층으로 이동합니다. 보존 정책이 정의되지 않은 경우 Amazon S3 Intelligent Tiering을 사용하여 객체를 자동으로 아카이브 계층으로 이동합니다.
10. Amazon CloudWatch를 사용하여 리소스 사용률을 모니터링하고 경보를 설정하여 규모 조정 이벤트를 트리거합니다.

11. 각 AWS 계정에 대해 AWS Budgets를 사용하여 비용 센터 및 사업부를 기준으로 계정의 비용 및 사용 예산을 설정합니다.
12. 계정의 비용 및 사용 예산을 설정하는 데 AWS Budgets를 사용하면 지출을 확실히 파악하고 예상치 못한 청구서를 피할 수 있으므로 비용을 더 잘 관리할 수 있습니다.

절차: 이 정책을 구현하기 위한 세부 절차를 제공하거나 각 정책 문을 구현하는 방법을 설명하는 다른 문서를 소개합니다. 이 섹션에서는 정책 요구 사항을 수행하기 위한 단계별 지침을 제공해야 합니다.

이 정책을 구현하려면 다양한 서드파티 도구 또는 AWS Config 규칙을 사용하여 정책 문 준수 여부를 확인하고 AWS Lambda 함수를 사용하여 자동화된 수정 작업을 트리거할 수 있습니다. AWS Organizations를 사용하여 정책을 적용할 수도 있습니다. 또한 정기적으로 리소스 사용량을 검토하고 필요에 따라 정책을 조정하여 비즈니스 요구 사항을 계속 충족하는지 확인해야 합니다.

구현 단계

- 이해관계자와 만남: 정책을 개발하려면 조직 내 이해관계자(클라우드 비즈니스 오피스, 엔지니어 또는 정책 시행을 위한 기능적 의사 결정권자)에게 요구 사항을 지정하고 문서화하도록 요청합니다. 광범위하게 시작하여 반복적인 접근 방식을 취하고 각 단계에서 가장 작은 단위까지 계속 세분화합니다. 팀원에는 조직 단위 또는 애플리케이션 소유자와 같이 워크로드에 직접적인 관심이 있는 구성원뿐만 아니라 보안 및 재무 팀과 같은 지원 그룹이 포함됩니다.
- 확인 받기: 각 팀이 AWS 클라우드에 액세스하고 배포할 수 있는 정책에 동의하는지 확인합니다. 각 팀이 조직 정책을 준수하고, 리소스 생성이 합의된 정책 및 절차에 적합한지 확인합니다.
- 온보딩 교육 세션 생성: 새로운 조직 구성원에게 온보딩 교육 과정을 이수하도록 하여 비용과 조직 요구 사항에 대해 인식하도록 합니다. 이전의 경험과 정책이 다를 것으로 가정하거나, 전혀 생각하고 있지 않을 수 있습니다.
- 워크로드 위치 정의: 국가, 국가 내 지역을 포함하여 워크로드가 운영되는 곳을 정의합니다. 이 정보는 AWS 리전 및 가용 영역에 매핑하는 데 사용됩니다.
- 서비스와 리소스 정의 및 그룹화: 워크로드에 필요한 서비스를 정의합니다. 서비스마다 필요한 리소스 유형, 크기 및 수를 지정합니다. 애플리케이션 서버나 데이터베이스 스토리지와 같은 기능별로 리소스 그룹을 정의합니다. 리소스는 여러 그룹에 속할 수 있습니다.
- 기능별 사용자 정의 및 그룹화: 누구인지 또는 조직에서 어떤 직책을 맡고 있는지가 아니라 무슨 일을 하며 워크로드를 어떻게 사용하는지에 초점을 두고 워크로드와 밀접한 일을 하는 사용자를 정의합니다. 유사한 사용자나 역할을 함께 그룹화합니다. AWS 관리형 정책을 가이드로 사용할 수 있습니다.
- 작업 정의: 이전에 식별된 위치, 리소스 및 사용자를 사용하여 수명 주기(개발, 운영 및 폐기) 동안 각자 워크로드 성과를 거두는 데 필요한 작업을 정의합니다. 각 위치에서 그룹의 개별 요소가 아니라

그룹을 기반으로 작업을 식별합니다. 읽기 또는 쓰기로 광범위하게 시작한 다음 각 서비스에 대한 특정 작업까지 세분화합니다.

- 검토 기간 정의: 워크로드 및 조직 요구 사항은 시간이 지남에 따라 변경될 수 있습니다. 조직의 우선순위와 일치하도록 워크로드 검토 일정을 정의합니다.
- 정책 문서화: 정의된 정책에 조직의 필요에 따라 액세스할 수 있는지 확인합니다. 이러한 정책은 환경의 액세스를 구현, 유지 관리 및 감사하는 데 사용됩니다.

리소스

관련 문서:

- Change Management in the Cloud
- 직무 기능에 대한 AWS 관리형 정책
- AWS 다중 계정 결제 전략
- Actions, Resources, and Condition Keys for AWS Services
- AWS의 관리 및 거버넌스
- Control access to AWS 리전 using IAM policies
- 글로벌 인프라 리전 및 AZ

관련 비디오:

- AWS Management and Governance at Scale

COST02-BP02 목표 및 타겟 이행

워크로드에 대한 비용 및 사용량 목표와 타겟을 모두 이행합니다. 목표는 예상 결과에 대한 조직의 방향성을 제공하고, 타겟은 워크로드에 대해 달성할 수 있는 구체적인 측정 가능한 결과를 제공합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

조직의 비용과 사용량에 대한 목표와 타겟을 설정합니다. AWS에서 성장하는 조직으로서 비용 최적화를 위한 목표를 설정하고 추적하는 것이 중요합니다. 이러한 목표 또는 핵심 성과 지표(KPI)에는 온디맨드 지출 비율이나 AWS Graviton 인스턴스 또는 gp3 EBS 볼륨 유형과 같은 최적화된 특정 서비스의

채택 등이 포함될 수 있습니다. 측정 가능하고 달성 가능한 목표를 설정하여 비즈니스 운영에 중요한 효율성 개선을 측정할 수 있습니다. 목표를 통해 조직은 원하는 성과를 달성하는데 필요한 지침을 얻고 나아갈 방향을 알 수 있습니다.

타겟을 통해 달성해야 할 구체적이고 측정 가능한 결과를 파악할 수 있습니다. 간단히 말해 목표는 나아가고자 하는 방향이고 타겟은 그 방향과의 거리와 목표 달성 시점에 해당합니다(구체성, 측정 가능성, 할당성, 현실성, 적시성(SMART) 지침 활용). 예를 들어, 목표는 비용은 조금(비선형) 늘리면서 플랫폼 사용량은 크게 늘리는 것입니다. 타겟은 플랫폼 사용량을 20% 늘리고 비용 증가를 5% 미만으로 유지하는 것입니다. 워크로드 효율성을 6개월마다 개선하는 것은 일반적인 목표의 또 다른 예입니다. 이에 따른 타겟은 비즈니스당 비용 지표를 6개월마다 5%씩 줄이는 것이 될 수 있습니다. 올바른 지표를 사용하고 조직에 대해 계산된 KPI를 설정하세요. 기본 KPI로 시작하여 나중에 비즈니스 요구 사항에 따라 발전시켜 나갈 수 있습니다.

비용 최적화의 목표는 워크로드 효율성을 높이는 것이며, 이는 즉 시간이 지남에 따라 워크로드의 비즈니스 결과당 비용을 줄이는 것입니다. 이 목표를 모든 워크로드에 적용하고 6개월에서 1년마다 효율성을 5% 높이는 것과 같은 타겟을 설정하세요. 클라우드에서는 비용 최적화 기능과 새로운 서비스 및 기능 릴리스를 통해 이를 달성할 수 있습니다.

타겟은 목표 달성을 위해 도달하고자 하는 정량화할 수 있는 벤치마크이며, 벤치마크는 실제 결과를 타겟과 비교합니다. KPI로 컴퓨팅 서비스(예: 스팟 채택, Graviton 채택, 최신 인스턴스 유형, 온디맨드 적용 범위), 스토리지 서비스(예: EBS GP3 채택, 오래된 EBS 스냅샷, Amazon S3 Standard 스토리지) 또는 데이터베이스 서비스 사용(예: RDS 오픈 소스 엔진, Graviton 채택, 온디맨드 적용 범위)의 단위당 비용에 대한 벤치마크를 설정합니다. 이러한 벤치마크와 KPI는 가장 비용 효과적인 방식으로 AWS 서비스를 사용하고 있는지 확인하는 데 도움이 될 수 있습니다.

다음 표에는 참조용 표준 AWS 지표 목록이 나와 있습니다. 각 조직은 이러한 KPI에 대해 서로 다른 타겟 값을 가질 수 있습니다.

범주	KPI(%)	설명
컴퓨팅	EC2 사용 범위	EC2 인스턴스의 전체 비용(또는 시간)을 기준으로 SP+RI+스팟을 사용하는 EC2 인스턴스 비용(또는 시간) 비교
컴퓨팅	컴퓨팅 SP/RI 사용률	총 가용 SP 또는 RI 시간을 기준으로 SP 또는 RI 사용 시간 비교

범주	KPI(%)	설명
컴퓨팅	EC2/시간당 비용	EC2 비용을 해당 시간에 실행 중인 EC2 인스턴스 수로 나눈 값
컴퓨팅	vCPU 비용	모든 인스턴스의 vCPU당 비용
컴퓨팅	최신 인스턴스 세대	Graviton(또는 기타 최신 세대 인스턴스 유형)의 인스턴스 비율
데이터베이스	RDS 적용 범위	RDS 인스턴스의 총 비용(또는 시간)을 기준으로 RI를 사용한 RDS 인스턴스 비용(또는 시간) 비교
데이터베이스	RDS 사용률	총 가용 RI 시간을 기준으로 RI 사용 시간 비교
데이터베이스	RDS 가동 시간	RDS 비용을 해당 시간에 실행 중인 RDS 인스턴스 수로 나눈 값
데이터베이스	최신 인스턴스 세대	Graviton(또는 기타 최신 인스턴스 유형)의 인스턴스 비율
스토리지	스토리지 사용률	최적화된 스토리지 비용(예: Glacier, Deep Archive 또는 Infrequent Access)을 총 스토리지 비용으로 나눈 값

범주	KPI(%)	설명
태그 지정	태그가 지정되지 않은 리소스	Cost Explorer: 1. 크레딧, 할인, 세금, 환불, 마켓플레이스를 필터링하고 최근 월별 비용을 복사합니다. 2. Cost Explorer에서 태그 없는 리소스만 표시를 선택합니다. 3. 태그 없는 리소스의 양을 월별 비용으로 나눕니다.

이 표를 사용하여 조직의 목표를 기반으로 계산해야 하는 타겟 값 또는 벤치마크 값을 포함합니다. 정확하고 현실적인 KPI를 정의하려면 비즈니스에 대한 특정 지표를 측정하고 해당 워크로드의 비즈니스 성과를 이해해야 합니다. 조직 내에서 성과 지표를 평가할 때는 각기 다른 용도에 맞는 여러 유형의 지표를 구별해야 합니다. 이러한 지표는 전반적인 비즈니스 영향을 직접 측정하기보다는 주로 기술 인프라의 성능과 효율성을 측정합니다. 예를 들어 서버 응답 시간, 네트워크 지연 시간 또는 시스템 가동 시간을 추적할 수 있습니다. 이러한 지표는 인프라가 조직의 기술 운영을 얼마나 잘 지원하는지 평가하는 데 중요합니다. 그러나 고객 만족, 수익 성장 또는 시장 점유율과 같은 광범위한 비즈니스 목표에 대한 직접적인 인사이트를 제공하지는 않습니다. 비즈니스 성과를 포괄적으로 이해하려면 이러한 효율성 지표를 비즈니스 성과와 직접 연관되는 전략적 비즈니스 지표로 보완하세요.

KPI 및 관련 비용 절감 기회를 거의 실시간으로 파악하고 시간 경과에 따른 진행 상황을 추적합니다. KPI 목표 정의 및 추적을 시작하려면 Cloud Intelligence Dashboards(CID)의 KPI 대시보드를 권장합니다. 비용 및 사용량 보고서(CUR)의 데이터를 기반으로 KPI 대시보드는 맞춤형 목표를 설정하고 시간 경과에 따른 진행 상황을 추적할 수 있는 일련의 권장 비용 최적화 KPI를 제공합니다.

KPI 목표를 설정하고 추적할 수 있는 다른 솔루션이 있다면 조직의 모든 클라우드 재무 관리 이해관계자가 이러한 방법을 채택하도록 해야 합니다.

구현 단계

- 예상 사용량 수준 정의: 먼저 사용량 수준에 초점을 맞춥니다. 애플리케이션 소유자, 마케팅 팀, 규모가 큰 비즈니스 팀과 협력하여 워크로드의 예상 사용량 수준을 파악합니다. 시간이 지남에 따라 고객

수요는 어떻게 변화할 수 있으며 시즌성 증가 또는 마케팅 캠페인으로 인해 어떤 변화가 있을 수 있나요?

- 워크로드 리소싱 및 비용 정의: 사용량 수준을 정의한 후 이러한 사용량 수준을 충족하는 데 필요한 워크로드 리소스의 변경 사항을 수량화합니다. 워크로드 구성 요소의 리소스 크기 또는 수를 늘리거나, 데이터 전송을 늘리거나, 워크로드 구성 요소를 특정 수준의 다른 서비스로 변경해야 할 수 있습니다. 각 주요 지점의 비용을 지정하고 사용량이 변경될 때 비용 변동을 예측합니다.
- 비즈니스 목표 정의: 예상 사용량과 비용 변화의 결과를 예상되는 기술 변화나 실행 중인 모든 프로그램과 결합하고 워크로드에 대한 목표를 설정합니다. 목표는 사용량과 비용 그리고 둘 사이의 관계를 다루어야 합니다. 목표는 간단하고 간략하며, 기업에서 어떤 결과를 원하는지 이해하는 데 도움이 되어야 합니다(예: 미사용 리소스는 특정 비용 수준 미만으로 유지). 각각의 사용하지 않은 리소스 유형에 대해 목표를 정의하거나 목표와 타겟에 손실을 초래하는 비용을 정의할 필요는 없습니다. 사용량 변화 없이 비용 변화만 예상되는 경우, 조직 프로그램(예: 훈련 및 교육을 통한 역량 쌓기)이 있는지 확인합니다.
- 타겟 정의: 정의된 각 목표에 대해 측정 가능한 타겟을 지정합니다. 워크로드의 효율성을 높이는 것이 목표라면, 타겟은 개선 수치(일반적으로 소비한 USD당 비즈니스 성과)와 달성 시점을 정량화합니다. 예를 들어, 과다 공급으로 인한 낭비를 최소화하겠다는 목표를 설정할 수 있습니다. 이 목표에서는 프로덕션 워크로드의 첫 번째 계층에서 컴퓨팅 오버프로비저닝으로 인한 낭비가 계층 컴퓨팅 비용의 10%를 초과하지 않도록 하는 것을 타겟으로 삼을 수 있습니다. 또한 두 번째 타겟은 프로덕션 워크로드의 두 번째 계층에서 컴퓨팅 오버프로비저닝으로 인한 낭비가 계층 컴퓨팅 비용의 5%를 초과하지 않아야 한다는 것일 수 있습니다.

리소스

관련 문서:

- 직무에 관한 AWS 관리형 정책
- AWS 다중 계정 결제 전략
- Control access to AWS 리전 using IAM policies
- S.M.A.R.T. Goals
- How to track your cost optimization KPIs with the CID KPI Dashboard

관련 비디오:

- Well-Architected Labs: Goals and Targets (Level 100)

관련 예제:

- What is a unit metric?
- Selecting a unit metric to support your business
- Unit metrics in practice – lessons learned
- How unit metrics help create alignment between business functions

COST02-BP03 계정 구조 구현

조직에 적합한 계정 구조를 구현합니다. 이를 통해 조직 전체에서 비용을 쉽게 할당하고 관리할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

AWS Organizations를 사용하여 다수의 AWS 계정을 생성할 수 있으며, 이를 통해 AWS 기반 워크로드가 확장함에 따라 환경을 중앙에서 통제할 수 있습니다. 조직 단위(OU) 구조로 AWS 계정을 그룹화하고 각 OU에 다수의 AWS 계정을 생성하여 조직 계층 구조를 모델링할 수 있습니다. 계정 구조를 생성하려면 어떤 AWS 계정이 관리 계정인지 먼저 결정해야 합니다. 그런 다음 관리 계정 모범 사례 및 구성원 계정 모범 사례를 따라 설계된 계정 구조를 기반으로 새 AWS 계정 계정을 생성하거나 기존 계정을 구성원 계정으로 선택할 수 있습니다.

이때 조직의 규모나 사용량에 관계없이 하나의 구성원 계정이 연결된 하나 이상의 관리 계정을 항상 보유하는 것이 좋습니다. 모든 워크로드 리소스는 구성원 계정 내에만 상주해야 하며 관리 계정에는 리소스를 생성하면 안 됩니다. 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.

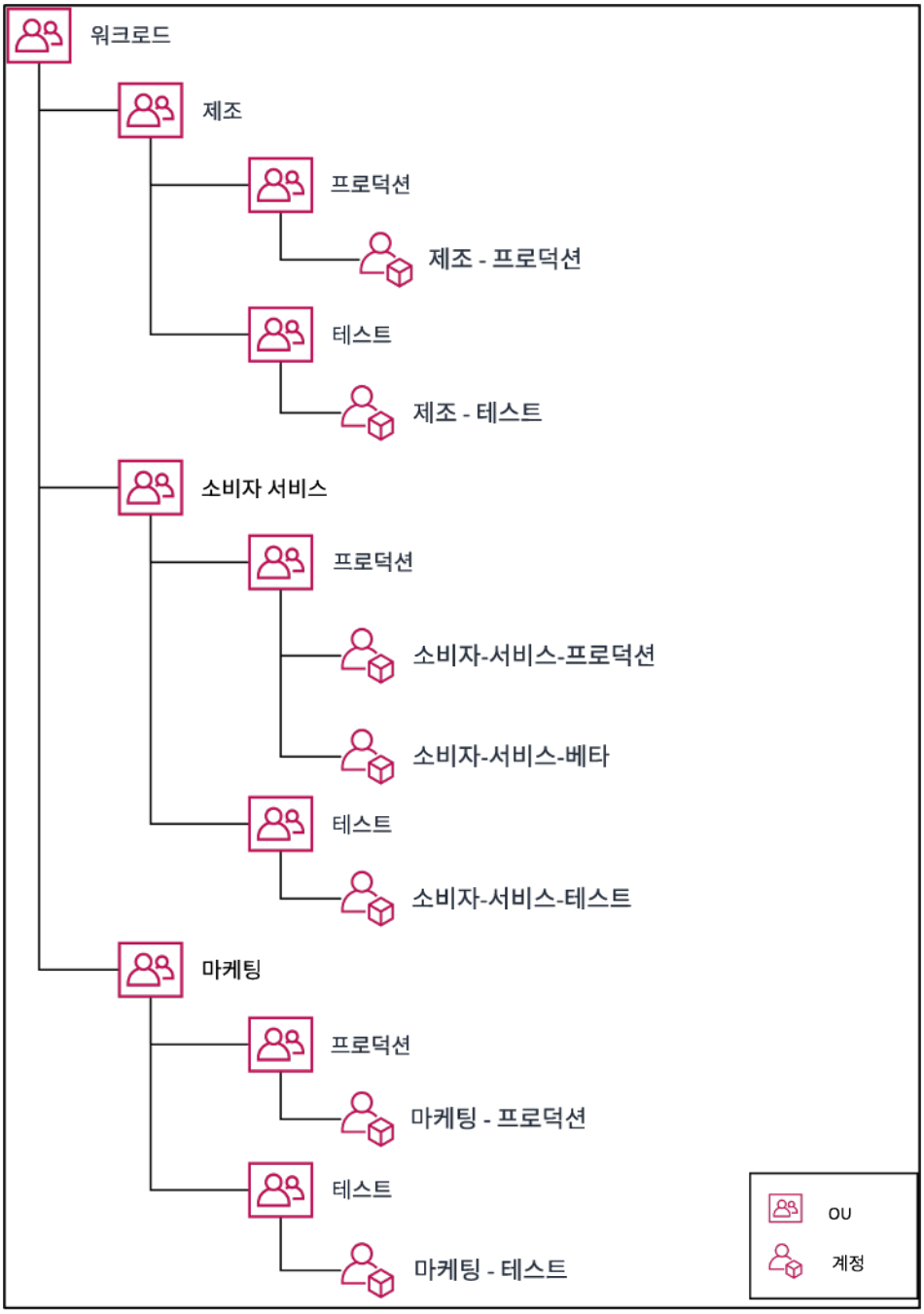
- 조직 단위, 비용 센터 또는 특정 워크로드 간에 관리 또는 재무 및 결제 작업을 분리해야 합니다.
- 특정 워크로드별로 AWS 서비스 한도가 설정됩니다.
- 워크로드와 리소스 간에 격리 및 분리에 대한 요구 사항이 있습니다.

AWS Organizations에서 통합 결제를 사용하는 경우 하나 이상의 구성원 계정과 관리 계정 간의 구조가 생성됩니다. 구성원 계정을 사용하면 비용과 사용량을 그룹으로 분리하고 구별할 수 있습니다. 각 조직 단위(재무, 마케팅, 영업 등), 각 환경 수명 주기(개발, 테스트, 프로덕션 등) 또는 각 워크로드(워크

로드 a, b, c)용으로 별도의 구성원 계정을 생성한 다음 통합 결제를 사용하여 이러한 연결 계정을 집계하는 방식이 흔히 사용됩니다.

통합 결제에서는 여러 구성원 AWS 계정의 결제를 하나의 관리 계정에 통합할 수 있으며, 각 연결 계정의 활동은 계속 확인할 수 있습니다. 관리 계정에 비용 및 사용량이 집계되므로 서비스 대량 구매 할인율을 극대화하고 약정 할인(절감형 플랜 및 예약형 인스턴스)을 최대한 활용하여 가장 높은 할인을 받을 수 있습니다.

다음 다이어그램은 조직 단위(OU)로 AWS Organizations를 사용하여 다양한 계정을 그룹화하고 각 OU에 다양한 AWS 계정을 배치하는 방법을 보여줍니다. 계정 구성을 위한 패턴을 제공하는 다양한 사용 사례 및 워크로드에는 OU를 사용하는 것이 좋습니다.



조직 단위에 여러 AWS 계정을 그룹화하는 예.

AWS Control Tower를 사용하면 여러 AWS 계정을 빠르게 설정하고 구성하여 조직의 요구 사항에 부합하는 거버넌스를 시행할 수 있습니다.

구현 단계

- 분리 요구 사항 정의: 분리 요구 사항은 보안, 신뢰성, 재무 구조와 같은 여러 요인을 결합한 것입니다. 각 요인을 순서대로 살펴보고 워크로드 또는 워크로드 환경이 다른 워크로드와 분리되어야 하는

지 여부를 지정합니다. 보안은 액세스 및 데이터 요구 사항에 대한 준수를 촉진합니다. 신뢰성은 한도를 관리하여 환경과 워크로드가 다른 요인에 영향을 미치지 않도록 합니다. Well-Architected 프레임워크의 보안과 신뢰성 원칙을 정기적으로 검토하고 제공된 모범 사례를 따릅니다. 재무 구조는 재무를 엄격하게 분리합니다(각기 다른 비용 센터, 워크로드 소유권 및 책임). 분리의 일반적인 예로는 별도의 계정에서 실행 중인 프로덕션 및 테스트 워크로드 또는 인보이스 및 결제 데이터를 계정을 소유한 조직의 개별 사업부나 부서 또는 이해관계자에 제공하기 위한 별도의 계정을 사용하는 등이 있습니다.

- 그룹화 요구 사항 정의: 그룹화 요구 사항은 분리 요구 사항에 우선하지 않지만 관리를 지원하는 데 사용됩니다. 분리할 필요가 없는 유사한 환경 또는 워크로드를 함께 그룹화합니다. 예를 들어, 하나 이상의 워크로드에 속하는 여러 테스트 또는 개발 환경을 함께 그룹화합니다.
- 계정 구조 정의: 이러한 분리와 그룹화를 사용하여 각 그룹에 대한 계정을 지정하고 분리 요구 사항을 유지 관리합니다. 이러한 계정은 구성원 또는 연결된 계정입니다. 이러한 구성원 계정을 하나의 관리 또는 지급인 계정으로 그룹화하면 사용량이 결합되어 모든 계정에서 더 큰 대량 구매 할인을 받을 수 있고 모든 계정에 대해 단일 청구서가 제공됩니다. 결제 데이터를 분리하고 각 구성원 계정에 결제 데이터의 개별 보기를 제공할 수 있습니다. 구성원 계정의 사용 내역 또는 결제 데이터가 다른 계정에 표시되지 않아야 하거나 AWS와 별도의 청구서가 필요한 경우 여러 관리 또는 지급인 계정을 정의합니다. 이 경우 구성원 계정마다 고유의 관리 또는 지급인 계정이 있습니다. 리소스는 항상 구성원 또는 연결 계정에 배치되어야 합니다. 관리 또는 지급인 계정은 관리에만 사용해야 합니다.

리소스

관련 문서:

- 비용 할당 태그 사용
- 직무에 관한 AWS 관리형 정책
- AWS 다중 계정 결제 전략
- Control access to AWS 리전 using IAM policies
- AWS Control Tower
- AWS Organizations
- 관리 계정 및 구성원 계정의 모범 사례
- Organizing Your AWS Environment Using Multiple Accounts
- Turning on shared reserved instances and Savings Plans discounts
- Consolidated billing
- Consolidated billing

관련 예제:

- Splitting the CUR and Sharing Access

관련 비디오:

- Introducing AWS Organizations
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations

관련 예제:

- Defining an AWS Multi-Account Strategy for telecommunications companies
- Best Practices for Optimizing AWS 계정
- Best Practices for Organizational Units with AWS Organizations

COST02-BP04 그룹 및 역할 구현

정책에 따라 그룹과 역할을 구현하고 각 그룹에서 인스턴스와 리소스를 생성, 수정 또는 폐기할 수 있는 사용자를 제어합니다. 예를 들어 개발, 테스트 및 프로덕션 그룹을 구현합니다. 이는 AWS 서비스와 서드파티 솔루션에 적용됩니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 지침

사용자 역할 및 그룹은 안전하고 효율적인 시스템을 설계하고 구현하는 데 필요한 기본 구성 요소입니다. 역할 및 그룹은 조직이 통제의 필요성과 유연성 및 생산성에 대한 요구 사항의 균형을 맞추어 궁극적으로 조직의 목표를 달성하고 사용자 요구를 충족하는 데 도움이 됩니다. AWS Well-Architected Framework 보안 원칙의 ID 및 액세스 관리 섹션에서 권장하는 바와 같이, 올바른 조건에서 올바른 사용자에게 올바른 리소스에 대한 액세스를 제공하려면 강력한 ID 관리 및 권한이 필요합니다. 사용자는 작업을 완료하는 데 필요한 액세스 권한만 받습니다. 이렇게 하면 무단 액세스 또는 오용과 관련된 위험을 최소화할 수 있습니다.

정책을 개발한 후에는 조직 내에서 논리적 그룹과 사용자 역할을 만들 수 있습니다. 이를 통해 권한을 할당하고, 사용을 제어하며, 강력한 액세스 제어 메커니즘을 구현하여 민감한 정보에 대한 무단 액세스를 방지할 수 있습니다. 개괄적인 수준에서 사람을 그룹화하는 작업부터 시작합니다. 일반적으로 이렇게 구성되는 그룹은 조직 단위 및 직무 역할(예: IT 부서의 시스템 관리자 또는 재무 관리자, 비즈니스

분석가)과 일치합니다. 유사한 작업을 수행하고 유사한 접근 권한이 필요한 사람이 각각의 그룹으로 분류됩니다. 역할은 그룹이 수행해야 할 작업을 정의합니다. 개별 사용자보다 그룹 및 역할에 대한 권한을 관리하는 것이 더 쉽습니다. 역할과 그룹을 사용하면 모든 사용자에게 일관되고 체계적으로 권한을 할당하여 오류와 불일치를 예방할 수 있습니다.

사용자의 역할이 변경되었을 때, 관리자는 개별 사용자 계정을 재구성하는 대신 역할 또는 그룹 수준에서 액세스를 조정할 수 있습니다. 예를 들어 IT의 시스템 관리자는 모든 리소스를 생성할 수 있는 접근 권한이 필요하지만 분석 팀원은 분석 리소스만 생성하면 됩니다.

구현 단계

- 그룹 구현: 조직 정책에 정의된 사용자 그룹을 사용하여 필요한 경우 해당 그룹을 만듭니다. 사용자, 그룹 및 인증에 대한 모범 사례는 AWS Well-Architected 프레임워크의 보안 원칙을 참조하세요.
- 역할 및 정책 구현: 조직 정책에 정의된 작업을 사용하여 필요한 역할과 액세스 정책을 생성합니다. 역할 및 정책에 대한 모범 사례는 AWS Well-Architected Framework의 보안 원칙을 참조하세요.

리소스

관련 문서:

- 직무에 관한 AWS 관리형 정책
- AWS 다중 계정 결제 전략
- AWS Well-Architected Framework 보안 원칙
- AWS Identity and Access Management (IAM)
- AWS Identity and Access Management 정책

관련 비디오:

- Why use Identity and Access Management

관련 예제:

- Control access to AWS 리전 using IAM policies
- Starting your Cloud Financial Management journey: Cloud cost operations

COST02-BP05 비용 제어 기능 구현

조직 정책 및 정의된 그룹과 역할을 기준으로 제어 기능을 구현합니다. 이렇게 하면 조직 요구 사항에 따라 정의된 비용만 발생합니다. 예를 들어 리전 또는 리소스 유형 액세스를 제어할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 지침

비용 제어를 구현하기 위한 일반적인 첫 번째 단계는 비용 또는 사용량 이벤트가 정책 범위를 벗어날 때 알림을 설정하는 것입니다. 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 Budgets를 사용하면 AWS 비용, 사용량 및 약정 할인(절감형 플랜 및 예약형 인스턴스)에 대한 알림을 설정하고 월간 예산을 정의할 수 있습니다. 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.

AWS Budgets를 통해 예산 한도를 설정한 후 AWS Cost Anomaly Detection을 사용하여 예상치 못한 비용을 줄일 수 있습니다. AWS Cost Anomaly Detection은 기계 학습을 사용하여 비용과 사용량을 지속적으로 모니터링함으로써 비정상적인 지출을 탐지하는 비용 관리 서비스입니다. 이를 통해 비정상적인 지출과 근본 원인을 식별하여 빠르게 조치를 취할 수 있습니다. 먼저 AWS Cost Anomaly Detection에서 비용 모니터링을 생성한 후 달러 임곗값을 설정하여 알림 기본 설정을 선택합니다(예: 1,000 USD 이상의 영향이 있는 이상을 알림). 알림을 수신하고 나면 이상과 비용의 영향에 대한 근본 원인을 분석할 수 있습니다. 또한 AWS Cost Explorer에서 자체적인 이상 분석을 모니터링 및 수행할 수도 있습니다.

AWS Identity and Access Management 및 AWS Organizations 서비스 제어 정책(SCP)을 통해 AWS에서 거버넌스 정책을 시행합니다. IAM은 AWS 서비스 및 리소스에 대한 접근을 안전하게 관리할 수 있는 기능을 제공합니다. IAM을 사용하면 AWS 리소스를 생성 및 관리할 수 있는 사용자, 생성할 수 있는 리소스 유형 및 리소스 생성 위치를 제어할 수 있습니다. 이렇게 하면 정의된 정책 외부에 리소스가 생성될 가능성이 최소화됩니다. 이전에 생성한 역할 및 그룹을 사용하고 IAM 정책을 할당하여 올바른 사용을 시행할 수 있습니다. SCP를 사용하면 조직 내 모든 계정에 대해 사용 가능한 최대 권한을 중앙에서 제어하여 계정이 액세스 제어 지침을 계속 준수할 수 있습니다. SCP는 모든 기능이 활성화된 조직에서만 사용할 수 있으며, 기본적으로 구성원 계정에 대한 작업을 거부하거나 허용하도록 SCP를 구성할 수 있습니다. 액세스 관리 구현에 대한 자세한 내용은 Well-Architected 보안 원칙 백서를 참조하세요.

AWS Service Quotas의 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 Service Quotas를 사용하여 할당량을 늘릴 수 있습니다.

구현 단계

- 지출에 대한 알림 구현: 정의된 조직 정책으로 AWS Budgets를 생성하여 지출이 정책을 벗어날 때 이를 알립니다. 전체 계정 지출에 대해 알리는 비용 예산을 계정당 하나씩 여러 개 구성합니다. 각 계정 내에서 해당 계정 내의 더 작은 단위에 대한 추가 비용 예산을 구성합니다. 이러한 단위는 계정 구조에 따라 다릅니다. 일반적인 예로 AWS 리전, 워크로드(태그 사용) 또는 AWS 서비스가 있습니다. 개인의 이메일 계정이 아닌 이메일 배포 목록을 알림에 대한 수신자로 구성합니다. 금액을 초과할 때에 대한 실제 예산을 구성하거나 예상 예산을 사용하여 예상 사용량에 대해 알릴 수 있습니다. 또한 특정 IAM 또는 SCP 정책을 시행하거나, 대상 Amazon EC2 또는 Amazon RDS 인스턴스를 중지할 수 있는 AWS 예산 작업을 사전 구성할 수 있습니다. 예산 작업은 자동으로 시작하거나 워크플로 승인이 필요할 수 있습니다.
- 비정상적인 지출에 대한 알림 구현: AWS Cost Anomaly Detection을 사용하여 조직 내 뜻밖의 비용을 줄이고 비정상적인 지출 가능성의 근본 원인을 분석할 수 있습니다. 비용 모니터링을 생성하여 지정된 세분화에서 비정상적인 지출을 식별하고 AWS Cost Anomaly Detection에서 알림을 구성하면, 비정상적인 지출이 감지되었을 때 알림이 전송됩니다. 이를 통해 이상의 근본 원인을 분석하고 비용에 대한 영향을 이해할 수 있습니다. AWS Cost Categories를 사용하는 동시에 AWS Cost Anomaly Detection을 구성하여 어떤 프로젝트 팀이나 사업부에서 예상치 못한 비용의 근본 원인을 분석하고, 시기 적절하며 필요한 조치를 취할지 식별할 수 있습니다.
- 사용량에 대한 제어 구현: 정의된 조직 정책으로 IAM 정책 및 역할을 구현하여 사용자가 수행할 수 있는 작업과 수행할 수 없는 작업을 지정합니다. AWS 정책에 여러 조직 정책이 포함될 수 있습니다. 정책을 정의한 것과 동일한 방식으로 광범위하게 시작한 다음 각 단계에서 보다 세분화된 제어를 적용합니다. 서비스 한도도 효과적인 사용량 제어 방식입니다. 모든 계정에 올바른 서비스 한도를 설정합니다.

리소스

관련 문서:

- 직무에 관한 AWS 관리형 정책
- AWS 다중 계정 결제 전략
- Control access to AWS 리전 using IAM policies
- AWS Budgets

- AWS Cost Anomaly Detection
- AWS 비용 관리

관련 비디오:

- How can I use AWS Budgets to track my spending and usage

관련 예제:

- IAM 액세스 관리 정책 예
- Example service control policies
- AWS 예산 작업
- 태그로 EC2 리소스 액세스를 제어하는 IAM 정책 생성
- 특정 Amazon EC2 리소스에 대한 IAM Identity 액세스 제한
- 채팅 애플리케이션 내 Amazon Q Developer를 사용한 비용 이상 탐지를 위한 Slack 통합

COST02-BP06 프로젝트 수명 주기 추적

프로젝트, 팀 및 환경의 수명 주기를 추적, 측정 및 감사하여 불필요한 리소스 사용 및 이에 따른 비용 지출을 막으세요.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 가이드

조직은 프로젝트 수명 주기를 효과적으로 추적함으로써 향상된 계획, 관리 및 리소스 최적화를 통해 더 나은 비용 관리를 실현할 수 있습니다. 추적을 통해 얻은 인사이트는 비용 효과성과 프로젝트의 전반적인 성공에 기여하는 정보 기반 의사 결정에 매우 중요합니다.

워크로드의 전체 수명 주기를 추적하면 워크로드 또는 워크로드 구성 요소가 더 이상 필요하지 않은 시기를 파악할 수 있습니다. 기존 워크로드와 구성 요소가 사용 중인 것처럼 보일 수 있지만 AWS에서 새 서비스나 기능이 출시되면 폐기되거나 채택될 수 있습니다. 워크로드의 이전 단계를 확인합니다. 워크로드가 프로덕션 단계로 전환된 후에는 이전 환경을 폐기하거나 다시 필요할 때까지 환경의 용량을 크게 줄일 수 있습니다.

리소스에 타임프레임 또는 알림을 태그하여 워크로드가 검토된 시간을 지정할 수 있습니다. 예를 들어 개발 환경을 몇 달 전에 마지막으로 검토했다면 다시 검토하여 새로운 서비스를 채택할 수 있는지 또는

환경이 사용 중인지 알아보는 것이 좋습니다. AWS의 myApplications를 사용하여 애플리케이션을 그룹화하고 태그를 지정하여 중요도, 환경, 최근 검토 날짜, 비용 센터와 같은 메타데이터를 관리하고 추적할 수 있습니다. 워크로드의 수명 주기를 추적하고 애플리케이션의 비용, 상태, 보안 태세 및 성능을 모니터링하고 관리할 수 있습니다.

AWS는 엔터티 수명 주기 추적에 사용할 수 있는 다양한 관리 및 거버넌스 서비스를 제공합니다. AWS Config 또는 AWS Systems Manager를 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. AWS에서 제공하는 다양한 이벤트 및 지표와 현재 시스템을 통합하면 중요한 수명 주기 이벤트 보기를 만들고 리소스를 사전에 관리하여 불필요한 비용을 줄일 수 있습니다.

애플리케이션 수명 주기 관리(ALM)와 마찬가지로 프로젝트 수명 주기 추적에는 설계 및 개발, 테스트, 프로덕션, 지원, 워크로드 이중화 등 여러 프로세스, 도구 및 팀이 함께 작업해야 합니다.

프로젝트 수명 주기의 각 단계를 주의 깊게 모니터링함으로써 중요한 인사이트를 얻고 통제를 강화하여 성공적인 프로젝트 계획, 구현 및 완료를 촉진합니다. 이렇게 세심한 감독을 거치면 프로젝트가 품질 표준을 충족할 뿐만 아니라 제시간에 예산 범위 내에서 납품되어 전반적인 비용 효율성을 높인다는 것을 확인할 수 있습니다.

엔터티 수명 주기 추적에 대한 자세한 내용은 AWS Well-Architected 운영 우수성 원칙 백서를 참조하세요.

구현 단계

- 프로젝트 수명 주기 모니터링 프로세스 수립: Cloud Center of Excellence 팀은 프로젝트 수명 주기 모니터링 프로세스를 수립해야 합니다. 프로젝트의 통제, 가시성 및 성과를 개선하기 위해 워크로드 모니터링에 대한 구조적이고 체계적인 접근 방식을 수립합니다. 모니터링 프로세스를 투명하고 협력적이며 지속적인 개선에 집중하도록 만들어 효과와 가치를 극대화합니다.
- 워크로드 검토 수행: 조직 정책에 정의된 대로 일정한 주기를 설정하여 기존 프로젝트를 감사하고 워크로드 검토를 수행합니다. 감사에 드는 노력은 조직의 대략적인 위험, 가치 또는 비용에 비례해야 합니다. 감사에 포함할 주요 영역은 조직의 인시던트 또는 가동 중단 위험, 가치 또는 조직에 대한 기여도(수익 또는 브랜드 평판으로 측정), 워크로드 비용(총 리소스 비용 및 운영 비용으로 측정), 워크로드 사용량(단위 시간당 조직 성과 수로 측정)입니다. 수명 주기 동안 이러한 영역이 변경되면 전체 또는 부분 폐기와 같은 워크로드 조정이 필요합니다.

리소스

관련 문서:

- Guidance for Tagging on AWS
- 애플리케이션 수명 주기 관리(ALM)란 무엇인가요?
- 직무에 관한 AWS 관리형 정책

관련 예제:

- Control access to AWS 리전 using IAM policies

관련 도구

- AWS Config
- AWS Systems Manager
- AWS Budgets
- AWS Organizations
- AWS CloudFormation

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.

모범 사례

- COST03-BP01 세부 정보 소스 구성
- COST03-BP02 비용 및 사용량에 조직 정보 추가
- COST03-BP03 비용 귀속 범주 식별
- COST03-BP04 조직 지표 설정
- COST03-BP05 결제 및 비용 관리 도구 구성
- COST03-BP06 워크로드 지표를 기준으로 비용 할당

COST03-BP01 세부 정보 소스 구성

비용 및 사용량 데이터의 분석과 투명성을 향상하기 위한 비용 관리 및 보고 도구를 설정합니다. 비용과 사용량을 추적하고 분리하는 데 도움이 되는 로그 항목을 생성하도록 워크로드를 구성합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

비용 관리 도구의 시간 단위 세분화와 같은 상세한 청구 정보를 통해 조직은 추가 세부 정보와 함께 소비량을 추적하고 비용 증가 원인을 파악할 수 있습니다. 이러한 데이터 소스는 전체 조직의 비용 및 사용량을 가장 정확하게 보여줍니다.

AWS Data Exports를 사용하여 AWS Cost and Usage Report(CUR) 2.0의 내보내기를 생성할 수 있습니다. 이것은 AWS로부터 상세한 비용 및 사용량 데이터를 받을 수 있는 새로운 방법이자 권장되는 방법입니다. 일부 개선 사항과 함께 모든 유료 AWS 서비스의 일별 또는 시간별 사용 세부 수준, 요금, 비용 및 사용 속성(CUR과 동일한 정보)을 제공합니다. 태그 지정, 위치, 리소스 속성 및 계정 ID를 포함하여 가능한 모든 차원을 CUR에서 사용할 수 있습니다.

생성하려는 내보내기 유형에 따라 표준 데이터 내보내기, QuickSight 통합을 통한 Cost and Usage Dashboard로 내보내기 또는 기존 데이터 내보내기의 세 가지 내보내기 유형이 있습니다.

- 표준 데이터 내보내기: Amazon S3에 반복적으로 전송되는 테이블의 사용자 지정 내보내기입니다.
- Cost and Usage Dashboard: QuickSight로 내보내고 통합하여 사전 구축된 Cost and Usage Dashboard를 배포합니다.
- 레거시 데이터 내보내기: 레거시 AWS Cost and Usage Report(CUR) 내보내기입니다.

다음과 같은 사용자 지정을 사용하여 데이터 내보내기를 생성할 수 있습니다.

- 리소스 ID 포함
- 분할 비용 할당 데이터
- 시간 단위 세분화
- 버전 관리
- 압축 유형 및 파일 형식

Amazon ECS 또는 Amazon EKS에서 컨테이너를 실행하는 워크로드의 경우, 분할 비용 할당 데이터를 활성화하면 컨테이너 워크로드가 공유 컴퓨팅 및 메모리 리소스를 사용하는 방식에 따라 컨테이너

비용을 개별 사업부 및 팀에 할당할 수 있습니다. 분할 비용 할당 데이터를 통해 AWS Cost and Usage Report에서 새로운 컨테이너 수준 리소스에 대한 비용 및 사용 데이터를 확인할 수 있습니다. 분할 비용 할당 데이터는 클러스터에서 실행되는 개별 ECS 서비스 및 작업의 비용을 기준으로 계산됩니다.

Cost and Usage Dashboard는 Cost and Usage Dashboard 테이블을 정기적으로 S3 버킷에 내보내고 사전 구축된 Cost and Usage Dashboard를 QuickSight에 배포합니다. 사용자 지정 기능 없이 비용 및 사용 데이터의 대시보드를 빠르게 배포하려는 경우 이 옵션을 사용합니다.

원하는 경우 레거시 모드에서 CUR을 계속 내보낼 수 있습니다. 이때 AWS Glue와 같은 기타 처리 서비스를 통합하여 분석용으로 데이터를 준비하고 Amazon Athena를 통해 SQL로 데이터를 쿼리하여 데이터 분석을 수행할 수 있습니다.

구현 단계

- 데이터 내보내기 생성: 원하는 데이터로 사용자 지정 내보내기를 생성하고 내보내기 스키마를 제어합니다. 기본 SQL을 사용하여 결제 및 비용 관리 데이터 내보내기를 생성하고 QuickSight와 통합하여 결제 및 비용 관리 데이터를 시각화합니다. 표준 모드에서 데이터를 내보내 Amazon Athena와 같은 기타 처리 도구를 사용하여 데이터를 분석할 수도 있습니다.
- 비용 및 사용 보고서 구성: 결제 콘솔을 사용하여 하나 이상의 비용 및 사용 보고서를 구성합니다. 모든 식별자 및 리소스 ID를 포함하는 시간 단위의 세분화로 보고서를 구성합니다. 다양한 세분화의 다른 보고서를 생성하여 더 개략적인 요약 정보를 제공할 수도 있습니다.
- Cost Explorer에서 시간별 세부 수준 구성: 지난 14일 동안의 시간별 세부 수준으로 비용 및 사용 데이터에 액세스하려면 결제 콘솔에서 시간별 및 리소스 수준 데이터를 활성화하는 방법을 고려합니다.
- 애플리케이션 로깅 구성: 애플리케이션에서 제공하는 각 비즈니스 성과를 추적하고 측정할 수 있도록 이에 대한 로깅을 확인합니다. 비용 및 사용량 데이터와 일치하도록 이 데이터의 세분화가 시간 단위 이상인지 확인합니다. 로깅 및 모니터링에 대한 자세한 내용은 Well-Architected 운영 우수성 원칙을 참조하세요.

리소스

관련 문서:

- AWS Data Exports
- AWS Glue
- QuickSight
- AWS 비용 관리 요금

- AWS 리소스에 태그 지정
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports
- Well-Architected 운영 우수성 요소

관련 예제:

- AWS Account Setup
- Data Exports for AWS Billing and Cost Management
- AWS Cost Explorer Common Use Cases

COST03-BP02 비용 및 사용량에 조직 정보 추가

조직, 워크로드 속성, 비용 할당 범주에 따라 태그 지정 스키마를 정의하여, 리소스를 필터링하고 검색하거나 비용 관리 도구에서 비용과 사용량을 모니터링할 수 있습니다. 가능하면 모든 리소스에서 목적, 팀, 환경, 비즈니스와 관련이 있는 기타 기준에 따라 일관적인 태그 지정을 구현합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

AWS에서 태그 지정을 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: 비용 센터, 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.

AWS 리소스(예: Amazon Elastic Compute Cloud 인스턴스 또는 Amazon Simple Storage Service 버킷)에 태그를 적용하고 활성화하면 AWS에서 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 Organizations에서 태그 정책을 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 태그 편집기를 사용하면 여러 리소스의 태그를 추가, 삭제 및 관리할 수 있습니다. Tag Editor에서는 태그를 관리할 리소스를 검색한 후 검색 결과에 나온 리소스에서 바로 태그를 관리합니다.

AWS Cost Categories를 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 분류하도록 범주 규칙을 정의합니다. 태그 지정에 더해 또 다른 수준의 관리 기능을 제공할 수 있습니다. 특정 계정과 태그를 여러 프로젝트에 매핑할 수도 있습니다.

구현 단계

- 태그 지정 체계 정의: 비즈니스 전반의 모든 이해관계자를 모아 스키마를 정의합니다. 여기에는 대개 기술직, 재무직 및 경영진이 포함됩니다. 모든 리소스가 보유해야 하는 태그 목록과 리소스가 보유해야 하는 태그 목록을 정의합니다. 조직 전체에 걸쳐 태그 이름과 값이 일치하는지 확인합니다.
- 리소스 태그 지정: 정의된 비용 속성 범주를 사용하여 범주에 따라 워크로드의 모든 리소스에 태그를 지정합니다. CLI, 태그 편집기, AWS Systems Manager 등의 도구를 사용하여 효율성을 높입니다.
- AWS Cost Categories 구현: 태그 지정을 구현하지 않고 비용 범주를 생성할 수 있습니다. 비용 범주는 기존의 비용 및 사용량 규모를 사용합니다. 스키마에서 범주 규칙을 생성하고, 비용 범주에 구현합니다.
- 태그 지정 자동화: 모든 리소스에서 많은 태그 지정을 유지하려면 리소스가 생성될 때 자동으로 태그가 지정되도록 태그 지정을 자동화합니다. 리소스가 생성될 때 리소스에 태그가 지정되었는지 확인하기 위해 AWS CloudFormation과 같은 서비스를 사용합니다. 또한 워크로드를 주기적으로 스캔하고 태그가 지정되지 않은 리소스를 제거하는 마이크로서비스를 사용하거나 Lambda 함수를 사용해 자동으로 태그를 지정하는 사용자 지정 솔루션을 생성할 수 있습니다. 이는 테스트 및 개발 환경에서 매우 유용합니다.
- 태그 지정 시 모니터링 및 보고: 조직 전체에서 높은 수준의 태그 지정을 유지하려면 워크로드 전체의 태그를 보고하고 모니터링합니다. AWS Cost Explorer를 사용하여 태그가 지정되거나 태그가 지정되지 않은 리소스의 비용을 보거나 태그 편집기와 같은 서비스를 사용할 수 있습니다. 태그가 지정되지 않은 리소스의 수를 정기적으로 검토하고 원하는 태그 지정 수준에 도달할 때까지 태그를 추가하는 작업을 수행합니다.

리소스

관련 문서:

- Tagging Best Practices
- AWS CloudFormation 리소스 태그
- AWS Cost Categories
- AWS 리소스에 태그 지정
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

관련 비디오:

- How can I tag my AWS resources to divide up my bill by cost center or project
- Tagging AWS Resources

COST03-BP03 비용 귀속 범주 식별

조직 내 비용을 내부 소비 주체에 할당하는 데 활용할 수 있는 조직 범주(예: 사업부, 부서, 프로젝트)를 파악합니다. 이러한 범주를 사용하여 지출에 대한 책임을 강화하고, 비용 인식을 제고하고, 효과적인 소비 행동을 유도하세요.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

비용 범주화 프로세스는 예산 책정, 회계, 재무 보고, 의사 결정, 벤치마킹, 프로젝트 관리에 매우 중요합니다. 지출을 분류하고 범주화하면 팀이 클라우드 여정 전반에서 발생하는 비용 유형을 더 잘 이해할 수 있어 정보에 입각한 결정을 내리고 효과적으로 예산을 관리할 수 있습니다.

클라우드 지출 책임은 엄격한 수요 및 비용 관리를 위한 강력한 인센티브를 제공합니다. 그 결과 클라우드 지출의 대부분을 소비 사업부 또는 팀에 할당하는 조직에서 클라우드 비용을 크게 절감할 수 있습니다. 또한 클라우드 지출을 할당하면 조직이 중앙화된 클라우드 거버넌스의 모범 사례를 더 많이 채택하는 데 도움이 됩니다.

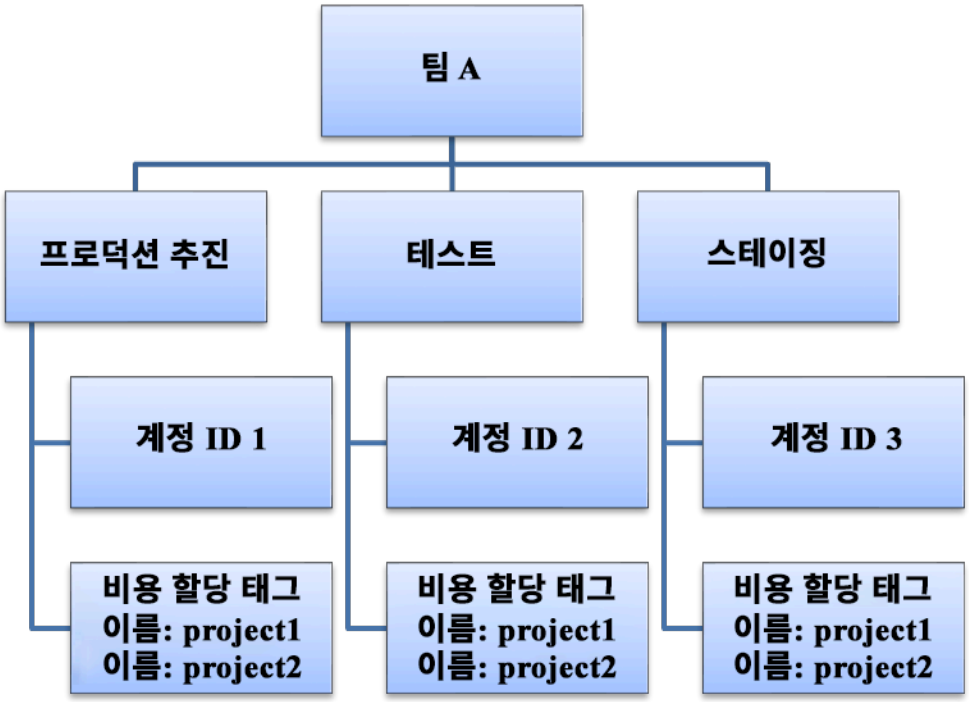
재무 팀 및 기타 이해관계자와 협력하여 정기적인 주기의 회의 중에 조직 내 비용 할당 방식 관련 요구사항을 파악합니다. 워크로드 비용은 개발, 테스트, 프로덕션 및 폐기를 포함한 전체 수명 주기에 걸쳐 할당되어야 합니다. 조직에서 학습, 직원 개발 및 아이디어 창출에 대해 발생한 비용의 귀속 방법을 파

악합니다. 그러면 이 목적으로 사용되는 계정을 일반적인 IT 비용 예산 대신 교육 및 개발 예산에 올바르게 할당할 수 있습니다.

조직 내 이해관계자와 함께 비용 귀속 범주를 정의한 후 AWS Cost Categories를 사용하여 AWS 클라우드에서 비용 및 사용 정보를 특정 프로젝트 비용 또는 부서나 사업부에 대한 AWS 계정과 같은 유의미한 범주로 그룹화합니다. 계정, 태그, 서비스 또는 요금 유형과 같은 다양한 차원을 사용하여 정의한 규칙에 따라 사용자 지정 범주를 생성하고 비용과 사용량 정보를 이러한 범주에 매핑할 수 있습니다. 비용 범주가 설정되면 범주별로 비용과 사용량 정보를 확인할 수 있어 조직이 더 나은 전략 결정과 구매 결정을 내릴 수 있습니다. 이들 범주는 AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report에도 표시됩니다.

예를 들어, 사업부(DevOps 팀)에 대한 비용 범주를 생성하고 각 범주 아래에서 정의된 그룹을 기반으로 여러 차원(AWS 계정, 비용 할당 태그, 서비스 또는 요금 유형)을 가진 여러 규칙(각 하위 범주에 대한 규칙)을 생성합니다. 비용 범주를 사용하면 규칙 기반 엔진을 통해 비용을 정리할 수 있습니다. 사용자가 구성한 규칙이 비용을 범주로 정리합니다. 이들 규칙 내에서 각 범주에 대한 여러 차원(예: 특정 AWS 계정, AWS 서비스, 요금 유형)을 사용하여 필터링할 수 있습니다. 이후 AWS 결제 및 비용 관리 및 비용 관리 콘솔에서 여러 제품에 걸쳐 이러한 범주를 사용할 수 있습니다. 여기에 AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report 및 AWS Cost Anomaly Detection도 추가되었습니다.

예를 들어, 다음 다이어그램은 여러 팀(비용 범주), 여러 환경(규칙), 및 여러 리소스 또는 자산(차원)을 가진 각 환경을 구성하여 조직의 비용과 사용량 정보를 그룹화하는 방법을 보여줍니다.



비용 및 사용 조직도

비용 점주를 사용하여 비용 그룹을 만들 수도 있습니다. 비용 범주를 생성하면(사용 레코드에 대한 비용 범주를 생성한 후 값을 업데이트하기 위해 최대 24시간 허용) AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report, AWS Cost Anomaly Detection에 나타납니다. AWS Cost Explorer 및 AWS Budgets에서 비용 범주가 추가적 청구 차원으로 표시됩니다. 이를 사용하여 특정 비용 범주 값을 필터링하거나 비용 범주를 기준으로 그룹화할 수 있습니다.

구현 단계

- 조직 범주 정의: 내부 이해관계자 및 사업부와 회의하여 조직의 구조 및 요구 사항을 반영하는 범주를 정의합니다. 이러한 범주는 사업부, 예산, 비용 센터 또는 부서와 같은 기존 재무 범주의 구조에 직접 매핑되어야 합니다. 교육과 같이 클라우드가 비즈니스에 제공하는 성과를 살펴보세요. 이들은 조직 범주이기도 합니다.
- 직무 범주 정의: 내부 이해관계자 및 사업부와 회의하여 비즈니스 내에서 맡은 역할을 반영하는 범주를 정의합니다. 워크로드 또는 애플리케이션 이름과 프로덕션, 테스트, 개발 등의 환경 유형이 이에 해당될 수 있습니다.
- AWS 비용 범주 정의: 비용 범주를 생성하고 AWS Cost Categories를 사용하여 비용 및 사용 정보를 체계적으로 정리하고 AWS 비용 및 사용을 유의미한 범주에 매핑합니다. 한 리소스에 여러 범주를 할당할 수 있으며 리소스는 서로 다른 여러 범주에 있을 수 있으므로 범주화된 구조 내에서 비용을 관리할 수 있도록 필요한 만큼 범주를 정의합니다(AWS Cost Categories 사용).

리소스

관련 문서:

- AWS 리소스에 태그 지정
- 비용 할당 태그 사용
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports
- AWS Cost Categories
- Managing your costs with AWS Cost Categories
- Creating cost categories
- Tagging cost categories

- Splitting charges within cost categories
- AWS Cost Categories Features

관련 예제:

- Organize your cost and usage data with AWS Cost Categories
- Managing your costs with AWS Cost Categories

COST03-BP04 조직 지표 설정

이 워크로드에 필요한 조직 지표를 설정합니다. 워크로드 지표의 예로는 생성된 고객 보고서 또는 고객에게 제공된 웹 페이지가 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

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 우선순위를 지정하거나 각 구성 요소에 대한 지표를 정의하고 추적할 수 있습니다. 팀과 협력하여 사용할 지표를 파악합니다. 이 단위는 워크로드의 효율성 또는 각 비즈니스 결과의 비용을 파악하는 데 사용됩니다.

구현 단계

- 워크로드 성과 정의: 비즈니스 이해관계자를 만나 워크로드에 대한 성과를 정의합니다. 이는 고객 사용량의 기본 척도이며 기술 지표가 아니라 비즈니스 지표여야 합니다. 워크로드당 거시 지표 수가 적어야 합니다(5개 미만). 워크로드가 서로 다른 사용 사례에 대해 여러 성과를 생성하는 경우 이들을 단일 지표로 그룹화합니다.
- 워크크로드 구성 요소 성과 정의: 선택적으로 크고 복잡한 워크로드가 있거나 잘 정의된 입력 및 출력을 사용하여 워크로드를 마이크로서비스 등의 구성 요소로 쉽게 나눌 수 있는 경우 각 구성 요소에 대한 지표를 정의합니다. 노력은 구성 요소의 가치와 비용을 반영해야 합니다. 가장 큰 구성 요소로 시작하여 더 작은 구성 요소로 진행합니다.

리소스

관련 문서:

- AWS 리소스에 태그 지정
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

COST03-BP05 결제 및 비용 관리 도구 구성

조직의 정책에 따라 비용 관리 도구를 구성하여 클라우드 지출을 관리하고 최적화합니다. 여기에는 비용 및 사용 데이터를 구성 및 추적하고, 통합 과금 및 액세스 권한을 통해 제어 기능을 강화하며, 예산 책정 및 예측을 통해 계획을 개선하고, 알림 또는 경고를 수신하며, 리소스 및 가격 최적화를 통해 비용을 절감하는 서비스, 툴 및 리소스가 포함됩니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

강한 책임감을 갖게 하려면 비용 배분 전략의 일환으로 계정 전략을 먼저 고려합니다. 이 단계를 제대로 해내면 더 이상의 조치는 필요 없을 수도 있습니다. 그렇지 않다면 인식이 결여되고 문제가 더 발생할 수 있습니다.

클라우드 지출에 대한 책임을 장려하려면 비용 및 사용량을 파악할 수 있는 도구에 대한 액세스 권한을 사용자에게 부여하세요. AWS에서는 다음과 같은 목적으로 모든 워크로드와 팀을 구성하는 것이 좋습니다.

- 구성: 자체 태그 전략 및 분류법을 사용하여 비용 할당 및 거버넌스 기준을 설정합니다. AWS Control Tower 또는 AWS 조직과 같은 도구를 사용하여 여러 AWS 계정을 생성합니다. 지원되는 AWS 리소스에 태그를 지정하고 조직 구조(사업부, 부서 또는 프로젝트)에 따라 의미 있게 분류합니다. 특정 비용 센터의 계정 이름에 태그를 지정하고 AWS Cost Categories와 매핑하여 사업부 계정을 해당 비용 센터로 그룹화하면 사업 단위 책임자가 한 곳에서 여러 계정의 소비 내역을 볼 수 있습니다.
- 액세스: 통합 결제를 통해 조직 전체의 청구 정보를 추적합니다. 올바른 이해관계자 및 비즈니스 소유자가 액세스할 수 있는지 확인합니다.
- 제어: 적절한 가드레일로 효과적인 거버넌스 메커니즘을 구축하여 서비스 제어 정책(SCP), 태그 정책, IAM 정책 및 예산 알림을 사용할 때 예상치 못한 시나리오를 방지합니다. 예를 들어 효과적인 제어 메커니즘을 사용하여 팀이 선호 지역에서만 특정 리소스를 생성하도록 허용하고 특정 태그(예: 비용 센터) 없이 리소스가 생성되지 않도록 할 수 있습니다.
- 현재 상태: 현재 비용 및 사용량 수준을 보여주는 대시보드를 구성합니다. 대시보드는 운영 대시보드와 같이 작업 환경 내의 가시성이 높은 위치에서 사용할 수 있어야 합니다. 데이터를 내보내고 AWS

Cost Optimization Hub 또는 지원되는 모든 제품에서 비용 및 사용 대시보드를 사용하여 이러한 가시성을 만들 수 있습니다. 다양한 페르소나에 대해 서로 다른 대시보드를 만들어야 할 수도 있습니다. 예를 들어 관리자 대시보드는 엔지니어링 대시보드와 다를 수 있습니다.

- 알림: 비용 또는 사용량이 정의된 한도를 초과하고 AWS Budgets 또는 AWS Cost Anomaly Detection에서 이상 징후가 발생할 경우 알림을 제공합니다.
- 보고서: 모든 비용 및 사용량 정보를 요약합니다. 상세하고 귀속 가능한 비용 데이터를 통해 클라우드 지출에 대한 인식과 책임성을 높입니다. 보고서를 사용하는 팀과 관련성이 높고 권장 사항이 포함된 보고서를 만듭니다.
- 추적: 구성된 목표 또는 목표를 기준으로 현재 비용 및 사용량을 보여줍니다.
- 분석: 팀원이 다양한 필터(리소스, 계정, 태그 등)를 사용하여 시간별, 일별 또는 월별 세부 수준까지 사용자 지정 및 심층 분석을 수행할 수 있습니다.
- 검사: 리소스 배포 및 비용 최적화 기회에 대한 최신 정보를 얻습니다. Amazon CloudWatch, Amazon SNS 또는 Amazon SES를 사용하여 조직 수준에서 리소스 배포에 대한 알림을 받을 수 있습니다. AWS Trusted Advisor 또는 AWS Compute Optimizer를 사용하여 비용 최적화 권장 사항을 검토하세요.
- 추세 보고서: 필요한 기간의 비용 및 사용량 변동을 보여주는 기능을 필요한 세부 수준으로 제공합니다.
- 예측: 향후 예상 비용을 표시하고 리소스 사용량을 예측하며 생성하는 예측 대시보드에 대한 지출을 표시합니다.

AWS Cost Optimization Hub를 사용하면 중앙 위치에서 통합된 잠재적 비용 절감 기회를 파악하고 Amazon Athena 통합을 위한 데이터 내보내기를 생성할 수 있습니다. 또한 AWS Cost Optimization Hub를 사용하여 대화형 비용 분석 및 안전한 비용 인사이트 공유에 QuickSight를 활용할 수 있는 Cost and Usage Dashboard를 배포할 수 있습니다.

조직에 필수적인 기술 또는 역량이 없다면 AWS ProServ, AWS Managed Services(AMS) 또는 AWS 파트너와 협력할 수 있습니다. 서드파티 도구를 사용할 수도 있지만 반드시 가치 제안을 확인해야 합니다.

구현 단계

- 팀 기반 도구 액세스 허용: 계정을 구성하고, 도구 사용에 필수적인 비용 및 사용 보고서에 대한 액세스 권한이 있는 그룹을 만들어, AWS Identity and Access Management를 통해 AWS Cost Explorer와 같은 도구에 대한 액세스를 관리하는 데 사용합니다. 이 그룹에는 애플리케이션을 소유하거나 관리하는 모든 팀의 담당자가 포함되어야 합니다. 그래야 모든 팀이 비용 및 사용량 정보에 액세스하여 사용량을 추적할 수 있습니다.

- 비용 태그 및 범주 구성: 팀, 사업부, 애플리케이션, 환경, 프로젝트 전반의 비용을 체계적으로 정리합니다. 리소스 태그를 사용하여 비용 할당 태그별로 비용을 정리할 수 있습니다. 태그, 계정, 서비스 등을 사용하여 크기를 기반으로 비용 범주를 생성하여 비용을 매핑합니다.
- AWS Budgets 구성: 워크로드의 모든 계정에서 AWS Budgets를 구성합니다. 태그 및 비용 범주를 사용하여 전체 계정 지출에 대한 예산과 워크로드에 대한 예산을 설정합니다. AWS Budgets에서 알림을 구성하여 책정된 예산을 초과하거나 예상 비용이 예산을 초과하는 경우 알림을 받습니다.
- AWS Cost Anomaly Detection 구성: 생성한 계정, 핵심 서비스 또는 비용 범주에 대해 AWS Cost Anomaly Detection을 사용하여 비용과 사용량을 모니터링하고 비정상적인 지출을 탐지할 수 있습니다. 집계된 보고서에서 개별적으로 알림을 수신하고, 이메일이나 Amazon SNS 주제로 알림을 수신하면 이상이 발생한 근본 원인을 분석하여 알아내고 비용을 높이는 요소를 찾아낼 수 있습니다.
- 비용 분석 도구 사용: 앞으로의 분석에 대한 비용 데이터를 시각화합기 위해 워크로드 및 계정에 대해 AWS Cost Explorer를 구성합니다. 전체 지출, 워크로드의 주요 사용량 지표 및 과거 비용 데이터를 기반으로 미래의 비용을 예측하는 워크로드 대시보드를 만듭니다.
- 비용 절감 분석 도구 사용: AWS Cost Optimization Hub를 사용하여 미사용 리소스 삭제, 적정 규모 조정, 절감형 플랜, 예약, 컴퓨팅 최적화 도구 권장 사항 등 맞춤형 권장 사항을 통해 비용 절감 기회를 식별할 수 있습니다.
- 고급 도구 구성: 선택적으로 시각적 객체를 생성하여 대화형 분석 및 비용 인사이트 공유를 쉽게 수행할 수 있습니다. AWS Cost Optimization Hub의 데이터 내보내기를 사용하면 QuickSight를 기반으로 조직에 필요한 Cost and Usage Dashboard를 생성하여 추가 세부 정보와 세부 수준을 제공할 수 있습니다. 또한 고급 쿼리에 대해 Amazon Athena의 데이터 내보내기를 사용하여 고급 분석 기능을 구현하고 QuickSight에서 대시보드를 만들 수 있습니다. AWS 파트너와 협력하여 통합 클라우드 청구서 모니터링 및 최적화를 위한 클라우드 관리 솔루션을 채택합니다.

리소스

관련 문서:

- What is AWS 결제 및 비용 관리 and Cost Management?
- 모범 사례 AWS 환경 설정
- Best Practices for Tagging AWS Resources
- AWS 리소스에 태그 지정
- AWS Cost Categories
- Analyzing your costs with AWS Budgets
- Analyzing your costs with AWS Cost Explorer

- What is AWS Data Exports?

관련 비디오:

- Deploying Cloud Intelligence Dashboards
- Get Alerts on any FinOps or Cost Optimization Metric or KPI

관련 예제:

- Cost and Usage Dashboard powered by QuickSight
- AWS Cost and Usage Governance 워크숍

COST03-BP06 워크로드 지표를 기준으로 비용 할당

사용량 지표 또는 비즈니스 성과에 따라 워크로드의 비용을 할당하여 워크로드 비용 효율성을 측정합니다. 분석 서비스를 통해 비용 및 사용량 데이터를 분석하는 프로세스를 구현하면 인사이트와 차지백 기능을 제공할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 가이드

비용 최적화는 최저 가격으로 비즈니스 성과를 거두는 것입니다. 이를 위해서는 워크로드 지표(워크로드 효율성으로 측정됨)를 기준으로 워크로드 비용을 할당해야 합니다. 로그 파일 또는 기타 애플리케이션 모니터링을 통해 정의된 워크로드 지표를 모니터링합니다. 이 데이터를 워크로드 비용과 결합합니다. 워크로드 비용은 특정 태그 값 또는 계정 ID에 연결된 비용을 조회하여 확인할 수 있습니다. 시간별 분석을 수행합니다. 요청 속도가 다양한 정적 비용 구성 요소(예: 백엔드 데이터베이스를 영구적으로 실행)가 있는 경우 일반적으로 효율성이 달라집니다(예: 사용량이 아침 9시에서 저녁 5시에 최고조에 달하고 야간에는 요청이 거의 없음). 정적 비용과 가변 비용 간의 관계를 이해하면 최적화 활동에 집중하는 데 도움이 됩니다.

Amazon Elastic Container Service(Amazon ECS) 및 Amazon API Gateway의 컨테이너식 애플리케이션과 같은 리소스에 비해, 공유 리소스에 대한 워크로드 지표를 생성하는 작업은 어려울 수 있습니다. 하지만 사용량을 분류하고 비용을 추적할 수 있는 몇 가지 방법이 있습니다. Amazon ECS 및 AWS Batch 공유 리소스를 추적해야 하는 경우 AWS Cost Explorer에서 분할 비용 할당 데이터를 활성화할 수 있습니다. 분할 비용 할당 데이터를 사용하면 컨테이너화된 애플리케이션의 비용 및 사용량을 이해하여 최적화하고 공유 컴퓨팅 및 메모리 리소스가 소비되는 방식에 따라 개별 비즈니스 엔터티에 애플리케이션 비용을 다시 할당할 수 있습니다.

구현 단계

- 워크로드 지표에 비용 할당: 정의된 지표와 구성된 태그를 사용하여 워크로드 출력과 워크로드 비용을 결합하는 지표를 생성합니다. Amazon Athena 및 Amazon QuickSight와 같은 분석 서비스를 사용하여 전체 워크로드와 모든 구성 요소에 대한 효율성 대시보드를 생성합니다.

리소스

관련 문서:

- AWS 리소스에 태그 지정
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

관련 예제:

- AWS 분할 비용 할당 데이터를 사용하여 Amazon ECS 및 AWS Batch의 비용 가시성 향상

리소스 폐기

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

모범 사례

- COST04-BP01 수명 주기 동안 리소스 추적
- COST04-BP02 폐기 프로세스 구현
- COST04-BP03 리소스 폐기
- COST04-BP04 리소스 자동 폐기
- COST04-BP05 데이터 보존 정책 적용

COST04-BP01 수명 주기 동안 리소스 추적

수명 주기 동안 리소스 및 리소스와 시스템의 관련성을 추적하는 방법을 정의하고 구현합니다. 태그 지정 기능을 사용하여 리소스의 워크로드나 기능을 파악할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

더 이상 필요하지 않은 워크로드 리소스를 폐기합니다. 일반적으로 테스트에 사용되는 리소스가 그 예입니다. 테스트가 완료된 후에는 리소스를 제거해도 됩니다. 태그를 사용하여 리소스를 추적(및 해당 태그에서 보고서를 실행)하면 사용되지 않거나 해당 라이선스가 만료될 예정이므로 폐기할 자산을 식별할 수 있습니다. 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 `feature-X testing`과 같은 값을 지정할 수 있습니다. 또 다른 예로, 폐기를 위해 특정 시간이나 기간을 정의하기 위해 삭제할 항목의 태그 키 이름 및 값과 같은 리소스에 대한 `LifeSpan` 또는 `TTL`을 사용할 수 있습니다.

구현 단계

- 태그 지정 체계 구현: 리소스가 속한 워크로드를 식별하는 태그 지정 체계를 구현하여 워크로드 내의 모든 리소스에 태그가 적절히 지정되도록 합니다. 태그 지정을 사용하면 목적, 팀, 환경 또는 비즈니스에 맞는 다른 기준에 따라 리소스를 분류할 수 있습니다. 태그 지정 사용 사례, 전략 및 기술에 대한 자세한 내용은 AWS Tagging Best Practices를 참조하세요.
- 워크로드 처리량 또는 출력 모니터링 구현: 워크로드 처리량 모니터링 또는 경보를 구현하여 입력 요청 또는 출력 완료 시 트리거합니다. 워크로드 요청 또는 출력이 0으로 떨어질 때 즉, 워크로드 리소스가 더 이상 사용되지 않을 때 알림을 제공하도록 구성합니다. 워크로드가 정상 조건에서 주기적으로 0으로 떨어질 경우 시간 계수를 통합합니다. 미사용 리소스 또는 활용률이 낮은 리소스에 대한 자세한 내용은 AWS Trusted Advisor Cost Optimization checks를 참조하세요.
- AWS 리소스 그룹화: AWS 리소스에 대한 그룹을 생성합니다. AWS Resource Groups를 사용하여 동일한 AWS 리전에 있는 AWS 리소스를 구성하고 관리할 수 있습니다. 조직 내 리소스를 식별하고 정렬하는 데 도움이 되도록 대부분의 리소스에 태그를 추가할 수 있습니다. 태그 편집기를 사용하여 지원되는 리소스에 태그를 대량으로 지정합니다. AWS Service Catalog를 사용하여 승인된 제품으로 구성된 포트폴리오를 생성 및 관리하고 최종 사용자에게 배포하며 제품 수명 주기를 관리하는 방법을 고려하세요.

리소스

관련 문서:

- AWS Auto Scaling
- AWS Trusted Advisor

- AWS Trusted Advisor Cost Optimization Checks
- AWS 리소스에 태그 지정
- 사용자 지정 지표 게시

관련 비디오:

- How to optimize costs using AWS Trusted Advisor

관련 예제:

- AWS 리소스 구성
- AWS Trusted Advisor를 사용하여 비용 최적화

COST04-BP02 폐기 프로세스 구현

미사용 리소스를 식별하고 폐기하는 프로세스를 구현합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

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.

구현 단계

- 폐기 프로세스 생성 및 구현: 워크로드 개발자 및 소유자와 협력하여 워크로드와 해당 리소스에 대한 폐기 프로세스를 구축합니다. 이 프로세스에서는 워크로드가 사용 중인지 그리고 각 워크로드 리소스가 사용 중인지 확인하는 방법을 다룹니다. 규제 요구 사항을 준수하면서 리소스를 폐기하고 리소스를 서비스에서 제거하는 데 필요한 단계를 자세히 알아봅니다. 라이선스 또는 연결된 스토리지와 같은 관련 리소스도 다룹니다. 워크로드 소유자에게 폐기 프로세스가 시작되었음을 알립니다.

  다음 폐기 단계를 사용하여 프로세스의 일부로 무엇을 확인해야 하는지 알 수 있습니다.

  - 폐기할 리소스 식별: AWS 클라우드에서 폐기 대상인 리소스를 식별합니다. 필요한 모든 정보를 기록하고 폐기를 예약합니다. 타임라인에서 프로세스 중 예상치 못한 문제가 발생했는지 여부와 시기를 반드시 설명합니다.
  - 조정 및 커뮤니케이션: 워크로드 소유자와 협력하여 폐기할 리소스를 확인합니다.

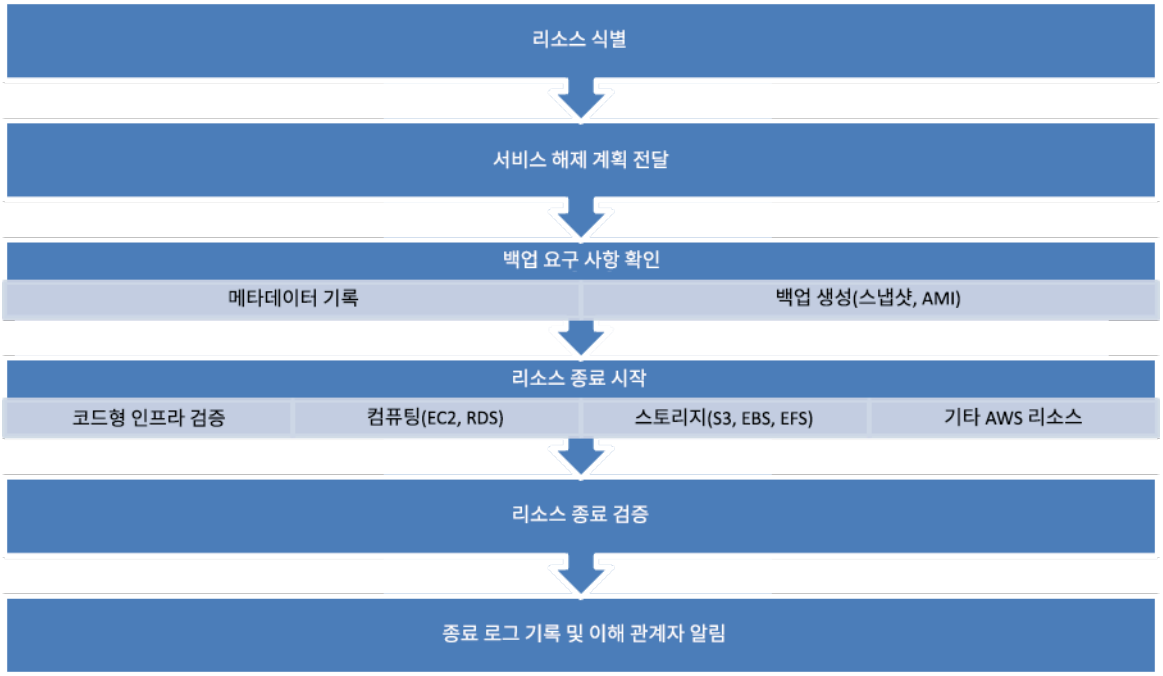
- 메타데이터 기록 및 백업 생성: 프로덕션 환경의 리소스에 필요하거나 중요한 리소스인 경우 메타데이터(예: 퍼블릭 IP, 리전, AZ, VPC, 서브넷 및 보안 그룹)를 기록하고 백업(예: Amazon Elastic Block Store 스냅샷 및 AMI, 키 내보내기 및 인증서 내보내기 실행)을 생성합니다.
- 코드형 인프라 검증: 필요한 경우 리소스를 재배포할 수 있도록 리소스가 AWS CloudFormation, Terraform, AWS 클라우드 개발 키트 (AWS CDK) 또는 기타 코드형 인프라 배포 도구를 통해 배포되었는지를 확인합니다.
- 액세스 방지: 일정 기간 제한적인 제어를 적용하여 리소스가 필요한지 확인하는 동안 리소스 사용을 방지합니다. 필요한 경우 리소스 환경을 원래 상태로 되돌릴 수 있는지 확인합니다.
- 내부 폐기 프로세스 준수: 조직의 관리 작업 및 폐기 프로세스를 따릅니다. 여기에는 조직 도메인에서 리소스 제거, DNS 레코드 제거, 구성 관리 도구, 모니터링 도구, 자동화 도구 및 보안 도구에서 리소스 제거 등이 포함됩니다.

리소스가 Amazon EC2 인스턴스인 경우 다음 목록을 참조합니다. Amazon EC2 리소스를 삭제하거나 종료하려면 어떻게 해야 하나요?

- 모든 Amazon EC2 인스턴스 및 로드 밸런서를 중지하거나 종료합니다. Amazon EC2 인스턴스는 종료된 후에 잠시 콘솔에 표시됩니다. 실행 중 상태가 아닌 인스턴스에 대해서는 요금이 부과되지 않습니다.
- Auto Scaling 인프라를 삭제합니다.
- 모든 전용 호스트를 해제합니다.
- 모든 Amazon EBS 볼륨 및 Amazon EBS 스냅샷을 삭제합니다.
- 모든 탄력적 IP 주소를 해제합니다.
- 모든 Amazon Machine Image(AMI) 등록을 취소합니다.
- 모든 AWS Elastic Beanstalk 환경을 종료합니다.

리소스가 Amazon S3 Glacier 스토리지의 객체이고 최소 스토리지 기간을 충족하기 전에 아카이브를 삭제하면 비례 할당으로 계산된 조기 삭제 요금이 청구됩니다.. Amazon S3 Glacier의 최소 스토리지 지속 시간은 사용된 스토리지 클래스에 따라 다릅니다. 각 스토리지 클래스의 최소 스토리지 기간에 대한 요약은 Amazon S3 스토리지 클래스 전반에 걸친 성능을 참조하세요. 조기 삭제 요금 계산 방법에 대한 자세한 내용은 Amazon S3 요금을 참조하세요.

다음 간단한 폐기 프로세스 순서도는 폐기 단계를 간략히 보여줍니다. 리소스를 폐기하기 전에 폐기 대상으로 식별된 리소스가 조직에서 사용하지 않는 것이 맞는지 확인해야 합니다.



리소스 폐기 흐름.

리소스

관련 문서:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS CloudTrail

관련 비디오:

- Delete CloudFormation stack but retain some resources
- Amazon EC2 인스턴스를 시작한 사용자 식별

관련 예제:

- Amazon EC2 리소스 삭제 또는 종료
- Amazon EC2 인스턴스를 시작한 사용자 식별

COST04-BP03 리소스 폐기

정기적 감사 또는 사용량 변화와 같은 이벤트에 의해 시작된 리소스를 폐기합니다. 폐기는 일반적으로 정기적으로 수행되며 수동 또는 자동으로 수행할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 지침

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

구현 단계

- 리소스 폐기: 더 이상 필요하지 않거나 라이선스 계약이 종료된 AWS 리소스의 사용 중지 단계에 해당합니다. 스냅샷 생성 또는 백업과 같은 원치 않는 중단이 발생하지 않도록, 폐기 단계 및 리소스 폐기로 진행하기 전에 최종 확인을 완료해야 합니다. 폐기 프로세스를 사용하여, 미사용 리소스로 식별된 각 리소스를 폐기합니다.

리소스

관련 문서:

- AWS Auto Scaling
- AWS Trusted Advisor

COST04-BP04 리소스 자동 폐기

중요하지 않은 리소스, 필수가 아닌 리소스 또는 사용률이 낮은 리소스를 파악하고 폐기하는 과정에서 워크로드가 리소스 종료를 정상적으로 처리하도록 설계합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 지침

자동화를 사용하여 폐기 프로세스와 관련된 비용을 줄이거나 제거합니다. 자동 폐기를 수행하도록 워크로드를 설계하면 수명 주기 동안 전체 워크로드 비용을 절감할 수 있습니다. Amazon EC2 Auto

Scaling 또는 Application Auto Scaling을 사용하여 폐기 프로세스를 수행할 수 있습니다. API 또는 SDK를 사용하여 워크로드 리소스를 자동으로 폐기하는 사용자 지정 코드를 구현할 수도 있습니다.

최신 애플리케이션은 서버리스 서비스 채택을 우선시하는 전략인 서버리스 우선으로 구축됩니다. AWS에서는 스택의 세 계층(컴퓨팅, 통합, 데이터 저장소) 모두를 위한 서버리스 서비스를 개발했습니다. 서버리스 아키텍처를 사용하면 자동 스케일 업 및 스케일 다운을 통해 트래픽이 적은 기간에 비용을 절감할 수 있습니다.

구현 단계

- Amazon EC2 Auto Scaling 또는 Application Auto Scaling 구현: 지원되는 리소스의 경우 Amazon EC2 Auto Scaling 또는 Application Auto Scaling으로 구성합니다. 이러한 서비스는 AWS 서비스를 이용할 때 사용률과 비용 효율성을 최적화하는 데 도움이 될 수 있습니다. 수요가 낮아지면 이러한 서비스는 과도한 지출을 방지할 수 있도록 모든 초과 리소스 용량을 자동으로 제거합니다.
- 인스턴스를 종료하도록 CloudWatch 구성: CloudWatch 경보를 사용하여 종료하도록 인스턴스를 구성할 수 있습니다. 폐기 프로세스의 지표를 사용하여 Amazon Elastic Compute Cloud 작업으로 경보를 구현합니다. 롤아웃하기 전에 비프로덕션 환경에서 작업을 확인합니다.
- 워크로드 내에 코드 구현: AWS SDK 또는 AWS CLI를 사용하거나 워크로드 리소스를 폐기할 수 있습니다. 애플리케이션 내에서 AWS와 통합되고, 더 이상 사용되지 않는 리소스를 종료하거나 제거하는 코드를 구현합니다.
- 서버리스 서비스 사용: 애플리케이션을 구축하고 실행하기 위해 AWS에서 서버리스 아키텍처 및 이벤트 기반 아키텍처 구축을 우선시합니다. AWS에서는 자동으로 최적화된 리소스 사용률 및 자동화된 폐기(스케일 인 및 스케일 아웃) 기능을 기본적으로 제공하는 여러 서버리스 기술 서비스를 제공합니다. 서버리스 애플리케이션을 통해 리소스 사용률이 자동으로 최적화되고 과다 프로비저닝에 대한 비용을 지불할 필요가 없습니다.

리소스

관련 문서:

- Amazon EC2 Auto Scaling
- Getting Started with Amazon EC2 Auto Scaling
- Application Auto Scaling
- AWS Trusted Advisor
- AWS의 서버리스
- 인스턴스를 중지, 종료, 재부팅 또는 복구하는 경보 생성

- Amazon CloudWatch 경보에 종료 작업 추가하기

관련 예제:

- Scheduling automatic deletion of AWS CloudFormation stacks

COST04-BP05 데이터 보존 정책 적용

조직의 요구 사항에 따라 객체 삭제를 처리할 수 있도록 지원되는 리소스에 대한 데이터 보존 정책을 정의합니다. 더 이상 필요 없는 불필요하거나 분리된 리소스와 객체를 파악하여 삭제합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

데이터 보존 정책 및 수명 주기 정책을 사용하여 폐기 프로세스 관련 비용과 식별된 리소스의 스토리지 비용을 줄입니다. 자동화된 스토리지 클래스 마이그레이션 및 삭제를 수행하기 위한 데이터 보존 정책 및 수명 주기 정책을 정의하면 수명 주기 중 전반적인 스토리지 비용을 줄일 수 있습니다. Amazon Data Lifecycle Manager를 사용하여 Amazon Elastic Block Store 스냅샷 및 Amazon EBS 지원 Amazon Machine Image(AMI)의 생성 및 삭제를 자동화하고, Amazon S3 Intelligent Tiering 또는 Amazon S3 수명 주기 구성을 사용하여 Amazon S3 객체의 수명 주기를 관리할 수 있습니다. API 또는 SDK를 사용하여 사용자 지정 코드를 구현해 자동으로 삭제되는 객체에 대한 수명 주기 정책 및 정책 규칙을 생성할 수도 있습니다.

구현 단계

- Amazon Data Lifecycle Manager 사용: Amazon Data Lifecycle Manager의 수명 주기 정책을 사용하여 Amazon EBS 스냅샷 및 Amazon EBS 지원 AMI의 삭제를 자동화합니다.
- 버킷에서 수명 주기 구성 설정: 버킷에서 Amazon S3 수명 주기 구성을 사용하여 비즈니스 요구 사항에 따라 객체 수명 주기 동안 Amazon S3에서 수행할 작업과 객체 수명 주기 종료 시 삭제를 정의합니다.

리소스

관련 문서:

- AWS Trusted Advisor
- Amazon Data Lifecycle Manager
- Amazon S3 버킷에서 수명 주기 구성 설정

관련 비디오:

- Automate Amazon EBS Snapshots with Amazon Data Lifecycle Manager
- Empty an Amazon S3 bucket using a lifecycle configuration rule

관련 예제:

- Empty an Amazon S3 bucket using a lifecycle configuration rule

비용 효율적인 리소스

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

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

주제

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

서비스를 선택할 때 비용 평가

모범 사례

- COST05-BP01 조직의 비용 요구 사항 파악
- COST05-BP02 워크로드의 모든 구성 요소 분석
- COST05-BP03 각 구성 요소의 철저한 분석 수행
- COST05-BP04 비용 효율적인 라이선스가 포함된 소프트웨어 선택
- COST05-BP05 조직의 우선순위에 따라 비용을 최적화할 이 워크로드의 구성 요소 선택
- COST05-BP06 시간별로 사용량이 달라지는 경우 비용 분석 수행

COST05-BP01 조직의 비용 요구 사항 파악

팀원과 협의하여 이 워크로드의 비용 최적화와 기타 원칙(예: 성능, 신뢰성) 사이의 균형을 정의합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

대부분의 조직에서 정보 기술(IT) 부서는 여러 소규모 팀으로 구성되어 있으며, 팀마다 팀원의 전문성과 기술을 반영하여 고유한 과제와 중점 영역을 가지고 있습니다. 조직의 전반적인 목표, 우선순위, 세

부 목표 및 각 부서 또는 프로젝트가 이러한 전반적인 목표에 어떻게 기여하는지 이해해야 합니다. 인력, 장비, 기술, 자재 및 외부 서비스를 포함한 모든 필수 리소스를 분류하는 것은 조직 목표를 달성하고 종합적인 예산 계획을 수립하는 데 매우 중요합니다. 비용 식별 및 이해를 위해 이러한 체계적인 접근 방식을 채택하는 것은 조직에서 현실적이고 강력한 비용 계획을 수립하는 데 필수적입니다.

워크로드에 사용할 서비스를 선택할 때는 조직의 우선순위를 이해하는 것이 중요합니다. 비용 최적화와 기타 AWS Well-Architected Framework 원칙(예: 성능 및 신뢰성) 사이에서 균형을 맞춥니다. 이 프로세스는 조직의 목표, 시장 상황 및 운영 역학의 변화를 반영하기 위해 체계적이고 정기적으로 수행되어야 합니다. 완전한 비용 최적화 워크로드는 조직의 요구 사항과 가장 일치하는 솔루션을 의미하며 꼭 비용이 가장 낮은 솔루션이 아닐 수 있습니다. 조직 내 모든 팀(예: 제품, 비즈니스, 기술, 재무)과 회의를 통해 정보를 수집합니다. 주력할 영역을 결정하거나 수행할 조치를 선택할 때 정보를 토대로 결정을 내릴 수 있도록 상충하는 이해관계나 대안 사이에서 장단점의 영향을 평가합니다.

예를 들어, 비용 최적화보다 새로운 기능의 시장 출시를 앞당기는 데 더 역점을 둘 수 있습니다. 아니면 데이터 유형에 최적화된 데이터베이스로 마이그레이션하고 애플리케이션을 업데이트하는 대신, 시스템 마이그레이션 작업을 간소화하기 위해 비관계형 데이터용 솔루션으로 관계형 데이터베이스를 선택할 수도 있습니다.

구현 단계

- 조직의 비용 요구 사항 파악: 제품 관리, 애플리케이션 소유자, 개발 및 운영 팀, 관리 및 재무 역할 등 조직의 다양한 팀원을 만납니다. 이 워크로드와 그 구성 요소에 대해 Well-Architected 원칙의 우선순위를 정합니다. 원칙을 순서대로 나열해야 합니다. 각 원칙에 가중치를 더할 수도 있습니다. 이를 통해 원칙에 얼마나 더 많은 초점이 맞춰져 있는지 또는 두 원칙 사이에 초점이 얼마나 비슷한지 알 수 있습니다.
- 기술적 부채 해결 및 문서화: 워크로드 검토 중에 기술 부채를 해결합니다. 백로그 항목을 문서화하여 향후 워크로드를 재검토합니다. 워크로드를 리팩터링 또는 다시 설계하여 한층 더 최적화하는 것을 목표로 합니다. 장단점을 다른 이해관계자에게 명확하게 전달하는 것이 중요합니다.

리소스

관련 모범 사례:

- REL11-BP07 가용성 목표 및 가동 시간 서비스 수준에 관한 계약(SLA)을 충족하도록 제품 설계
- OPS01-BP06 장단점 평가

관련 문서:

- AWS 총 소유 비용(TCO) 계산기
- Amazon S3 storage classes
- 클라우드 제품

COST05-BP02 워크로드의 모든 구성 요소 분석

현재 크기나 비용을 막론하고 모든 워크로드 구성 요소가 분석되도록 해야 합니다. 검토 작업은 현재 비용과 예상 비용 등의 잠재적인 이점을 반영해야 합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

조직에 비즈니스 가치를 제공하도록 설계된 워크로드 구성 요소에는 다양한 서비스가 포함될 수 있습니다. 각 구성 요소에 대해 비즈니스 요구 사항을 해결하기 위해 특정 AWS 클라우드 서비스를 선택할 수도 있습니다. 이 선택은 이러한 서비스에 대한 친숙도나 사용 경험 등 여러 요인의 영향을 받을 수 있습니다.

COST05-BP01 조직의 비용 요구 사항 파악에서 언급한 대로, 조직의 요구 사항을 파악한 후 워크로드의 모든 구성 요소를 철저히 분석합니다. 현재 및 예상 비용과 크기를 고려하여 각 구성 요소를 분석합니다. 수명 주기 동안 발생할 수 있는 워크로드 절감 효과와 비교하여 분석 비용을 고려합니다. 이 워크로드의 모든 구성 요소를 분석하는 데 드는 노력은 구성 요소를 최적화함으로써 예상되는 잠재적 절감 또는 개선 효과에 상응해야 합니다. 예를 들어 제안된 리소스의 비용이 월 10 USD이고 과소 예측된 로드가 월 15 USD를 초과하지 않는 경우, 비용을 50%(월 5 USD) 절감하기 위해 들여야 하는 하루분의 노력이 시스템 수명 동안 얻을 수 있는 잠재적 이점보다 더 클 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. 예를 들어 처음에는 EMR 서버리스를 사용하는 것이 적절할 수 있습니다. 그러나 이 서비스의 소비가 증가함에 따라 EMR on EC2로 전환하면 워크로드의 해당 구성 요소에 대한 비용을 줄일 수 있습니다.

AWS Cost Explorer 및 AWS Cost and Usage Report(CUR)를 사용하여 개념 증명(PoC) 또는 환경 실행 비용을 분석할 수 있습니다. AWS Pricing Calculator를 사용하여 워크로드 비용을 추정할 수도 있습니다.

기술 팀이 워크로드를 검토하기 위해 따라야 할 워크플로를 작성하세요. 이 워크플로를 단순하게 유지하되, 팀이 워크로드의 각 구성 요소와 가격을 이해할 수 있도록 필요한 모든 단계를 다루어야 합니다. 그러면 조직은 각 팀의 특정 요구 사항에 따라 이 워크플로를 따르고 사용자 지정할 수 있습니다.

1. 워크로드에 사용 중인 각 서비스 나열: 이는 좋은 출발점입니다. 현재 사용 중인 모든 서비스와 비용의 출처를 파악하세요.
2. 해당 서비스의 요금 책정 방식 이해: 각 서비스의 요금 모델을 이해합니다. AWS 서비스마다 사용량, 데이터 전송, 기능별 요금 등의 요소에 따라 가격 책정 모델이 다릅니다.
3. 예상치 못한 워크로드 비용이 발생하고 예상 사용량 및 비즈니스 성과와 일치하지 않는 서비스에 집중: AWS Cost Explorer 또는 AWS Cost and Usage Report를 사용하여 비용이 가치나 사용량에 비례하지 않는 이상치 또는 서비스를 식별합니다. 비용을 비즈니스 성과와 연관시켜 최적화 작업의 우선순위를 정하는 것이 중요합니다.
4. AWS Cost Explorer, CloudWatch Logs, VPC 흐름 로그 및 Amazon S3 Storage Lens를 통해 이러한 고비용의 근본 원인 파악: 이러한 도구는 고비용을 진단하는 데 중요한 역할을 합니다. 각 서비스는 사용량과 비용을 확인하고 분석하기 위한 다양한 렌즈를 제공합니다. 예를 들어 Cost Explorer에서는 전체 비용 추세를 파악할 수 있고, CloudWatch Logs는 운영 인사이트를 제공하며, VPC 흐름 로그는 IP 트래픽을 표시하고, Amazon S3 Storage Lens는 스토리지 분석에 유용합니다.
5. AWS Budgets를 사용하여 서비스 또는 계정의 일정 금액에 대한 예산 설정: 예산을 설정하면 비용을 사전에 관리할 수 있습니다. AWS Budgets를 사용하여 사용자 지정 예산 임곗값을 설정하고 비용이 임곗값을 초과할 경우 알림을 받습니다.
6. Amazon CloudWatch 경보를 구성하여 청구 및 사용 알림 전송: 비용 및 사용량 지표에 대한 모니터링 및 알림을 설정합니다. CloudWatch 경보를 통해 특정 임곗값 위반 시 이를 알릴 수 있으므로 개입 응답 시간이 향상됩니다.

현재 속성에 관계없이 모든 워크로드 구성 요소에 대한 전략적 검토를 통해 시간이 지남에 따라 눈에 띄는 개선과 비용 절감이 이루어지도록 합니다. 이 검토 프로세스에 투자하는 노력은 실현될 수 있는 잠재적 이점을 주의 깊게 고려하여 신중하게 이루어져야 합니다.

구현 단계

- 워크로드 구성 요소 나열: 워크로드 구성 요소 목록을 작성합니다. 이 목록을 사용하여 각 구성요소가 분석되었는지 확인할 수 있습니다. 여기에 드는 노력은 조직의 우선순위에 정의된 워크로드에 대한 중요도를 반영해야 합니다. 리소스를 그룹화하면 프로덕션 데이터베이스 스토리지의 경우처럼 데이터베이스가 여러 개인 경우 효율성이 향상됩니다.

- 구성 요소 목록의 우선순위 지정: 구성 요소 목록을 가져와서 작업량순으로 우선순위를 지정합니다. 일반적으로 구성 요소의 비용(가장 비싼 것부터 가장 싼 것까지) 또는 조직 우선순위에 정의된 중요도의 순서를 따릅니다.
- 분석 수행: 목록의 각 구성 요소에 대해 사용 가능한 옵션과 서비스를 검토하고 조직의 우선순위에 가장 잘 맞는 옵션을 선택합니다.

리소스

관련 문서:

- AWS Pricing Calculator
- AWS Cost Explorer
- Amazon S3 storage classes
- AWS 클라우드 제품

관련 비디오:

- AWS Cost Optimization Series: CloudWatch

COST05-BP03 각 구성 요소의 철저한 분석 수행

조직에서 발생하는 각 구성 요소의 전반적인 비용을 확인합니다. 그런 다음 운영 및 관리 비용을 감안하여 총 소유 비용을 계산합니다(특히, 클라우드 제공업체에서 관리형 서비스를 사용하는 경우). 검토 작업은 잠재적 이점을 반영해야 합니다(예를 들어 분석에 소요된 시간이 구성 요소의 비용에 비례).

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

팀에서는 이렇게 절약된 시간을 기술적 부채 청산, 혁신 및 부가 가치 기능과 비즈니스를 차별화할 수 있는 요인을 만드는 데 집중할 수 있다는 점을 고려하세요. 예를 들어 온프레미스 환경에서 클라우드로 데이터베이스를 최대한 빠르게 리프트 앤 시프트(리호스팅이라고도 함)하고 최적화는 나중에 수행해야 할 수 있습니다. AWS에서 라이선스 비용이 거의 발생하지 않거나, 전혀 발생하지 않을 수 있는 관리형 서비스를 사용하여 실현한 비용 절감 이점을 파악하는 것이 좋습니다. AWS의 관리형 서비스는 서비스 유지 관리를 위한 운영 및 관리 부담(예: OS 패치 적용 또는 업그레이드)을 없애 혁신 및 비즈니스에 집중할 수 있도록 합니다.

관리형 서비스는 클라우드 규모로 운영되므로 거래 또는 서비스당 비용을 절감할 수 있습니다. 애플리케이션의 핵심 아키텍처를 변경하지 않고 실질적인 이점을 달성하기 위해 잠재적인 최적화를 수행할 수 있습니다. 예를 들어, Amazon Relational Database Service(RDS) 등과 같은 서비스형 데이터베이스 플랫폼으로 마이그레이션하거나 AWS Elastic Beanstalk과 같은 완전관리형 플랫폼으로 애플리케이션을 마이그레이션하여 데이터베이스 인스턴스를 관리하는 데 소요되는 시간을 단축할 수 있습니다.

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 SDK를 사용해 AWS Managed Services의 속성을 수정하여 변화하는 수요에 맞게 리소스 요구를 조정할 수 있습니다. 예를 들어, Amazon EMR 클러스터나 Amazon Redshift 클러스터의 노드 수를 늘리거나 줄여서 스케일 아웃 또는 스케일 인할 수 있습니다.

또한 AWS 리소스의 여러 인스턴스를 압축하여 리소스 사용 밀도를 높일 수도 있습니다. 예를 들어 단일 Amazon Relational Database Service(RDS) 데이터베이스 인스턴스에 소형 데이터베이스 여러 개를 프로비저닝할 수 있습니다. 그리고 사용량이 증가하면 스냅샷 및 복원 프로세스를 사용하여 데이터베이스 중 하나를 전용 Amazon RDS 데이터베이스 인스턴스로 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 API와 SDK를 사용하면 서비스를 수정하는 데 필요한 지속적인 작업을 사실상 수행하지 않아도 됩니다.

Amazon RDS, Amazon Redshift, Amazon ElastiCache에서는 관리형 데이터베이스 서비스를 제공합니다. Amazon Athena, Amazon EMR, Amazon OpenSearch Service에서는 관리형 분석 서비스를 제공합니다.

AMS는 엔터프라이즈 고객 및 파트너를 대신하여 AWS 인프라를 운영하는 서비스입니다. 이 서비스를 사용하면 규정을 준수하는 안전한 환경에 워크로드를 배포할 수 있습니다. AMS는 자동화가 포함된 엔터프라이즈 클라우드 운영 모델을 사용하므로 고객은 조직 요구 사항을 충족하면서 클라우드로 더 빠르게 이전하고 지속적인 관리 비용을 절감할 수 있습니다.

구현 단계

- 철저한 분석 수행: 구성 요소 목록을 사용하여 가장 높은 우선순위부터 가장 낮은 우선순위까지 각 구성 요소를 살펴봅니다. 우선순위가 높고 비용이 많이 드는 구성 요소의 경우 추가 분석을 수행하고 사용 가능한 모든 옵션과 장기적인 영향을 평가합니다. 우선순위가 낮은 구성 요소의 경우 사용량 변화로 인해 구성 요소의 우선순위가 변경되는지 평가한 다음 적절한 작업에 대한 분석을 수행합니다.

- 관리형 및 비관리형 리소스 비교: 관리하는 리소스의 운영 비용을 고려하고 AWS 관리형 리소스와 비교합니다. 예를 들어, Amazon EC2 인스턴스에서 실행 중인 데이터베이스를 검토한 다음, Amazon EC2 기반 Apache Spark를 실행하는 것에 비해 Amazon RDS 옵션(AWS 관리형 서비스) 또는 Amazon EMR을 사용하는 경우를 비교합니다. 자체 관리형 워크로드에서 AWS 완전관리형 워크로드로 이전하는 경우 옵션을 주의해서 살펴보세요. 여기서 고려해야 할 가장 중요한 세 가지 요소는 사용하려는 관리형 서비스의 유형, 데이터 마이그레이션에 사용할 프로세스 및 AWS 공동 책임 모델에 대한 이해도입니다.

리소스

관련 문서:

- AWS 총 소유 비용(TCO) 계산기
- Amazon S3 storage classes
- AWS 클라우드 제품
- AWS Shared Responsibility Model

관련 비디오:

- Why move to a managed database?
- What is Amazon EMR and how can I use it for processing data?

관련 예제:

- Why to move to a managed database
- Consolidate data from identical SQL Server databases into a single Amazon RDS for SQL Server database using AWS DMS
- Deliver data at scale to Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Migrate an ASP.NET web application to AWS Elastic Beanstalk

COST05-BP04 비용 효율적인 라이선스가 포함된 소프트웨어 선택

오픈 소스 소프트웨어에는 워크로드 비용에서 상당한 부분을 차지할 수 있는 소프트웨어 라이선스 비용이 없습니다. 라이선스가 부여된 소프트웨어가 필요한 경우 CPU와 같은 임의의 속성에 바인딩된 라

이선스를 피하고 결과 또는 성과에 바인딩된 라이선스를 찾습니다. 이러한 라이선스의 비용은 해당 라이선스가 제공하는 혜택에 더 근접하게 조정됩니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 가이드

오픈 소스는 소프트웨어가 특정 무료 배포 기준을 준수함을 알리는 소프트웨어 개발이라는 맥락에서 시작되었습니다. 오픈 소스 소프트웨어는 누구나 검사, 수정, 개선할 수 있는 소스 코드로 구성됩니다. 라이선스 비용을 최소화하기 위해 비즈니스 요구 사항, 엔지니어의 역량, 예상 사용량 또는 기타 기술 종속성에 따라 AWS에서 오픈 소스 소프트웨어를 사용하는 것을 고려할 수 있습니다. 즉, 오픈 소스 소프트웨어를 사용하면 소프트웨어 라이선스 비용을 줄일 수 있습니다. 라이선스 비용은 워크로드 규모가 확장됨에 따라 워크로드 비용에 상당한 영향을 미칠 수 있습니다.

총 비용 대비 라이선스 소프트웨어의 이점을 측정하여 워크로드를 최적화하세요. 라이선스 변경 사항과 이러한 변경 사항이 워크로드 비용에 미치는 영향을 모델링합니다. 공급업체가 데이터베이스 라이선스 비용을 변경하는 경우 이 비용이 워크로드의 전반적인 효율성에 어떤 영향을 미치는지 조사합니다. 공급업체의 기간별 요금 발표를 고려하여 제품 전반의 라이선스 변경 추세를 파악합니다. 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.

예를 들어 Linux 운영 체제로 us-east-1에서 Amazon EC2 인스턴스를 운영하면 Windows에서 실행되는 다른 Amazon EC2 인스턴스를 실행하는 것에 비해 비용을 약 45% 절감할 수 있습니다.

AWS Pricing Calculator는 Amazon RDS 인스턴스 및 다양한 데이터베이스 엔진과 같은 다양한 라이선스 옵션을 사용하여 다양한 리소스의 비용을 비교할 수 있는 포괄적인 방법을 제공합니다. 또한 AWS Cost Explorer는 기존 워크로드, 특히 다른 라이선스와 함께 제공되는 워크로드의 비용에 대해 매우 유용한 관점을 제공합니다. 라이선스 관리를 위해 AWS License Manager에서는 소프트웨어 라이선스를 감독하고 처리할 수 있는 간소화된 방법을 제공합니다. AWS 클라우드에서 선호하는 오픈 소스 소프트웨어를 배포하고 운영할 수 있습니다.

구현 단계

- 라이선스 옵션 분석: 사용 가능한 소프트웨어의 라이선스 약관을 검토합니다. 필요한 기능을 갖춘 오픈 소스 버전을 찾고 라이선스가 부여된 소프트웨어의 혜택이 비용보다 큰지 확인합니다. 소프트웨어에서 제공하는 이점과 소프트웨어의 비용이 일치하는 것이 유리한 약관입니다.
- 소프트웨어 공급자 분석: 공급자의 과거 요금 또는 라이선스 변경을 검토합니다. 특정 공급자의 하드웨어 또는 플랫폼에서 실행에 대한 징벌적 조건과 같이 성과에 부합하지 않는 변경 사항을 찾습니다. 또한 공급업체가 감사를 수행하는 방법과 부과될 수 있는 페널티를 알아봅니다.

리소스

관련 문서:

- Open Source at AWS
- AWS 총 소유 비용(TCO) 계산기
- Amazon S3 storage classes
- 클라우드 제품

관련 예제:

- Open Source Blogs
- AWS Open Source Blogs
- Optimization and Licensing Assessment

COST05-BP05 조직의 우선순위에 따라 비용을 최적화할 이 워크로드의 구성 요소 선택

워크로드에 대한 모든 구성 요소를 선택할 때는 비용을 고려해야 합니다. 여기에는 애플리케이션 수준 및 관리형 서비스 또는 서버리스, 컨테이너 또는 이벤트 기반 아키텍처를 사용한 전반적인 비용 절감이 포함됩니다. 오픈 소스 소프트웨어, 라이선스 요금이 없는 소프트웨어 또는 비용 절감을 위한 대안을 사용하여 라이선스 비용을 최소화합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

모든 구성 요소를 선택할 때 서비스 및 옵션 비용을 고려합니다. 이 과정에서 Amazon Relational Database Service(RDS), Amazon DynamoDB, Amazon Simple Notification Service(SNS), Amazon Simple Email Service(Amazon SES) 등의 애플리케이션 수준 서비스와 관리형 서비스를 사용하여 전체적인 조직 비용을 절감할 수 있습니다.

컴퓨팅 구성 요소의 경우에는 서버리스 서비스와 컨테이너를 사용합니다(예: 정적 웹 사이트의 경우 AWS Lambda 및 Amazon Simple Storage Service(S3)). 가능하면 애플리케이션을 컨테이너화하고 Amazon Elastic Container Service(Amazon ECS) 또는 Amazon Elastic Kubernetes Service(Amazon EKS)와 같은 AWS 관리형 컨테이너를 사용합니다.

오픈 소스 소프트웨어 또는 라이선스 요금이 없는 소프트웨어를 사용하여 라이선스 비용을 최소화합니다(예: 컴퓨팅 워크로드용 Amazon Linux 또는 Amazon Aurora로 데이터베이스 마이그레이션).

Lambda, Amazon Simple Queue Service(Amazon SQS), Amazon SNS, Amazon SES와 같은 서버리스 또는 애플리케이션 수준 서비스를 사용할 수 있습니다. 이러한 서비스를 사용하면 리소스를 관리할 필요가 없으며 코드 실행, 대기열 서비스 및 메시지 전송 기능이 제공됩니다. 또 다른 이점은 사용량에 따라 성능과 비용이 조정되므로 효율적인 비용 할당 및 귀속이 가능하다는 것입니다.

이벤트 기반 아키텍처 사용은 서버리스 서비스에서도 가능합니다. 이벤트 기반 아키텍처는 푸시 기반이므로 이벤트가 라우터에 나타날 때 모든 것이 온디맨드 방식으로 발생합니다. 이렇게 하면 이벤트가 있는지 확인하기 위한 지속적인 폴링 비용을 지불하지 않습니다. 즉, 네트워크 대역폭 소비, CPU 사용률, 유휴 폴링 용량, SSL/TLS 핸드셰이크가 줄어듭니다.

서버리스에 대한 자세한 내용은 Well-Architected Serverless Application Lens 백서를 참조하세요.

구현 단계

- 각 서비스를 선택하여 비용 최적화: 우선순위가 지정된 목록 및 분석을 사용하여 조직의 우선순위와 가장 잘 일치하는 각 옵션을 선택합니다. 수요를 충족하기 위해 용량을 늘리는 대신 더 저렴한 비용으로 더 뛰어난 성능을 선사할 수 있는 다른 옵션을 고려해 보세요. 예를 들어, AWS에서 데이터베이스에 대해 예상되는 트래픽을 검토해야 하는 경우 인스턴스 크기를 늘리거나 Amazon ElastiCache 서비스(Redis 또는 Memcached)를 사용하여 데이터베이스에 캐시된 메커니즘을 제공하는 것을 고려하세요.
- 이벤트 기반 아키텍처 평가: 서버리스 아키텍처를 사용하면 분산된 마이크로서비스 기반 애플리케이션을 위한 이벤트 기반 아키텍처를 구축할 수도 있습니다. 이러한 아키텍처는 확장 가능하고 복원력이 있으며 민첩하고 비용 효과적인 솔루션을 구축하는 데 도움이 됩니다.

리소스

관련 문서:

- AWS 총 소유 비용(TCO) 계산기
- AWS Serverless
- 이벤트 기반 아키텍처(EDA)란 무엇인가요?
- Amazon S3 storage classes
- 클라우드 제품
- Amazon ElastiCache (Redis OSS)

관련 예제:

- Getting started with event-driven architecture
- 이벤트 중심 아키텍처
- How Statsig runs 100x more cost-effectively using Amazon ElastiCache (Redis OSS)
- AWS Lambda 함수 작업의 모범 사례

COST05-BP06 시간별로 사용량이 달라지는 경우 비용 분석 수행

워크로드는 시간이 지남에 따라 바뀔 수 있습니다. 일부 서비스 또는 기능은 다양한 사용 수준에서 더 비용 효율적입니다. 예상 사용량에 따라 시간별로 각 구성 요소 분석을 수행하면 수명 주기 동안 워크로드를 비용 효과적으로 유지할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

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 다시 시작해야 합니다. 사용량의 큰 변화는 대체 서비스가 더 적합하다는 의미일 수 있습니다.

데이터를 AWS 클라우드로 이전해야 하는 경우 데이터세트가 파일, 데이터베이스, 머신 이미지, 블록 볼륨 또는 테이프 백업이든 상관없이 AWS에서 제공하는 광범위한 서비스 및 파트너 도구를 사용하여 데이터세트를 마이그레이션할 수 있습니다. 예를 들어, 많은 양의 데이터를 AWS 안팎으로 이동하거나 엣지에서 데이터를 처리하기 위해 AWS 목적별 디바이스 중 하나를 사용하여 페타바이트 단위의 데이터를 오프라인에서 비용 효율적으로 이동할 수 있습니다. 또 다른 예로, 더 빠른 데이터 전송 속도의 경우 직접 연결 서비스가 비즈니스에 필요한 일관된 연결을 제공하는 VPN보다 더 저렴할 수 있습니다.

시간의 흐름에 따라 달라지는 사용량에 대한 비용 분석을 기반으로 규모 조정 활동을 검토합니다. 결과를 분석하여 여러 인스턴스 유형 및 구매 옵션으로 인스턴스를 추가하도록 규모 조정 정책을 조정할 수 있는지 알아봅니다. 설정을 검토하여 플릿 크기가 더 작은 사용자 요청을 처리하기 위해 최솟값을 줄일 수 있는지 확인하고 더 많은 리소스를 추가하여 예상되는 높은 수요를 충족합니다.

조직의 이해관계자와 논의하여 시간의 흐름에 따라 달라지는 사용량에 대한 비용 분석을 수행하고 AWS Cost Explorer의 예측 기능을 사용하여 서비스 변경이 미치는 잠재적인 영향을 예측합니다. AWS Budgets, CloudWatch 결제 경보 및 AWS Cost Anomaly Detection을 사용하여 사용 수준 트리거를 모니터링해 가장 비용 효과적인 서비스를 식별하여 더 빠르게 구현합니다.

구현 단계

- 예측된 사용 패턴 정의: 마케팅 및 제품 소유자와 같은 조직과 협력하여 워크로드의 예상 사용 패턴과 예측 사용 패턴을 문서화합니다. 비즈니스 이해관계자와 과거 비용 및 예측 비용과 사용량 증가에 대해 논의하여 비즈니스 요구 사항에 따라 증가되도록 합니다. 더 많은 사용자가 AWS 리소스를 사용할 것으로 예상되는 일, 주 또는 월을 식별합니다. 리소스 사용이 늘어난다는 것은 기존 리소스 용량을 증가하거나 추가 서비스를 채택하여 비용을 줄이고 성능을 개선해야 함을 나타냅니다.
- 예측 사용량에 대한 비용 분석 수행: 정의된 사용 패턴을 사용하여 이러한 각 지점에서 분석을 수행합니다. 분석 작업은 잠재적 성과를 반영해야 합니다. 예를 들어 사용량 변화가 큰 경우 비용과 변경 사항을 확인하기 위해 철저한 분석을 수행해야 합니다. 다시 말해, 비용이 높아지면 비즈니스를 위한 사용량도 따라서 증가해야 합니다.

리소스

관련 문서:

- AWS 총 소유 비용(TCO) 계산기
- Amazon S3 storage classes
- 클라우드 제품
- Amazon EC2 Auto Scaling
- Cloud Data Migration
- AWS Snow Family

관련 비디오:

- AWS OpsHub for Snow Family

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 시작되는 반복적인 프로세스일 수 있습니다. 규모 조정에 들어가는 노력의 비용이 워크로드 수명에 걸친 잠재적 절감액보다 높다면 이 작업은 단 한 번만 수행될 수도 있습니다.

AWS에서는 다음과 같은 다양한 방식이 사용됩니다.

모범 사례

- COST06-BP01 비용 모델링 수행
- COST06-BP02 데이터를 기준으로 리소스 유형, 크기, 개수 선택
- COST06-BP03 지표를 기준으로 리소스 유형, 크기, 개수 자동 선택
- COST06-BP04 공유 리소스 사용 고려

COST06-BP01 비용 모델링 수행

조직 요구 사항(예: 비즈니스 요구 사항 및 기존 약정)을 파악하고 워크로드 및 각 구성 요소의 비용 모델링(전반적인 비용)을 수행합니다. 그리고 예상되는 다양한 부하에서 워크로드의 벤치마크 활동을 수행하여 비용을 비교합니다. 모델링 작업은 잠재적 이점을 반영해야 합니다. 예를 들면 구성 요소의 비용과 비례하여 시간을 소비해야 합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

워크로드와 각 구성 요소에 대한 비용 모델링을 수행하여 리소스 간의 균형을 파악하고 특정 성능 수준을 고려하여 워크로드의 각 리소스에 대해 적합한 크기를 찾습니다. 비용 고려 사항을 이해하면 계획된 워크로드 배포에 대한 가치 실현 성과를 평가할 때 조직의 비즈니스 사례 및 의사 결정 프로세스를 알릴 수 있습니다.

그리고 예상되는 다양한 부하에서 워크로드의 벤치마크 활동을 수행하여 비용을 비교합니다. 모델링 작업은 잠재적 이점을 반영해야 합니다(예: 소요 시간이 구성 요소 비용 또는 예상 절감액에 비례). 모범 사례는 AWS Well-Architected Framework 성능 효율성 원칙의 검토 섹션을 참조하세요.

예를 들어, 컴퓨팅 리소스로 구성된 워크로드에 대한 비용 모델링을 생성하기 위해 AWS Compute Optimizer에서 워크로드 실행을 위한 비용 모델링을 지원할 수 있습니다. 이 서비스는 사용량 기록을 기준으로 컴퓨팅 리소스에 적합한 크기 권장 사항을 제공합니다. AWS Compute Optimizer 내에서 보다 정확한 권장 사항을 제공하는 데 도움이 되는 메모리 지표를 수집하도록 CloudWatch 에이전트가 Amazon EC2 인스턴스에 배포되었는지 확인합니다. 기계 학습을 활용하여 위험 수준에 따라 여러 권장 사항을 제시하는 무료 서비스이므로 컴퓨팅 리소스에 사용하기에 적합한 데이터 소스입니다.

다른 서비스 및 워크로드 구성 요소(예: AWS Trusted Advisor, Amazon CloudWatch 및 Amazon CloudWatch Logs)에 대한 작업 크기를 올바르게 조정하기 위해 사용자 지정 로그와 함께 데이터 소스

로 사용할 수 있는 여러 서비스가 있습니다. AWS Trusted Advisor는 리소스를 확인하여 사용률이 낮은 리소스에 플래그를 지정합니다. 그러면 리소스의 크기를 올바르게 조정하고 비용 모델링을 생성할 수 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 99번째 백분위수를 적절하게 선택합니다.
- 모든 워크로드 주기를 포함하는 데 필요한 분석 기간의 정확한 세부 수준을 선택합니다. 예를 들어 분석을 2주 동안 수행하는 경우 사용률이 높은 월 단위 주기를 분석하지 못하여 리소스가 너무 적게 프로비저닝될 수 있습니다.
- 기존 약정, 다른 워크로드에 대해 선택한 요금 모델, 더 빠르게 혁신하는 기능을 고려하여 계획된 워크로드에 대해 올바른 AWS 서비스를 선택하고 핵심 비즈니스 가치에 집중합니다.

구현 단계

- 리소스에 대한 비용 모델링 수행: 테스트할 특정 리소스 유형 및 크기의 별도 계정에 워크로드 또는 개념 증명을 배포합니다. 테스트 데이터로 워크로드를 실행하고 테스트 실행 시간의 비용 데이터와 함께 출력 결과를 기록합니다. 그런 다음 워크로드를 다시 배포하거나 리소스 유형 및 크기를 변경하고 테스트를 다시 실행합니다. 비용 모델링을 생성하는 동안 이러한 리소스와 함께 사용할 수 있는 제품의 라이선스 비용과 이러한 리소스를 배포하고 관리하기 위한 예상 운영(노동력 또는 엔지니어링) 비용을 포함합니다. 기간(시간별, 일별, 월별, 연간 또는 3년)에 따른 비용 모델링을 고려합니다.

리소스

관련 문서:

- AWS Auto Scaling
- 올바른 크기 조정을 위한 기회 식별
- Amazon CloudWatch의 기능
- Cost Optimization: Amazon EC2 Right Sizing
- AWS Compute Optimizer
- AWS 요금 계산기

관련 예제:

- Perform a Data-Driven Cost Modelling
- 계획된 AWS 리소스 구성 비용 예측
- Choose the right AWS tools

COST06-BP02 데이터를 기준으로 리소스 유형, 크기, 개수 선택

워크로드 및 리소스 특성에 대한 데이터를 기준으로 리소스 크기나 유형을 선택합니다. 예를 들어, 컴퓨팅, 메모리, 처리량, 쓰기 집약형과 같은 기준이 적용됩니다. 일반적으로는 워크로드의 이전(온프레미스) 버전, 설명서 또는 워크로드와 관련된 기타 정보 출처를 사용해 리소스 사용량을 선택합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

Amazon EC2는 다양한 사용 사례에 맞게 선택할 수 있도록 CPU, 메모리, 스토리지 및 네트워킹 용량 수준이 서로 다른 광범위한 인스턴스 유형을 제공합니다. 이러한 인스턴스 유형에는 CPU, 메모리, 스토리지, 네트워킹 기능이 서로 다르게 조합되어 있어 프로젝트에 적합한 리소스 조합을 다양하게 선택할 수 있습니다. 인스턴스 유형마다 크기가 다양하므로 워크로드의 수요에 따라 리소스를 조정할 수 있습니다. 필요한 인스턴스 유형을 결정하려면 인스턴스에서 실행하려는 애플리케이션 또는 소프트웨어의 시스템 요구 사항에 관한 세부 정보를 수집해야 합니다. 이러한 세부 정보에는 다음이 포함되어야 합니다.

- 운영 체제
- CPU 코어 수
- GPU 코어
- 시스템 메모리(RAM) 용량
- 스토리지 유형 및 공간
- 네트워크 대역폭 요구 사항

컴퓨팅 요구 사항의 목적과 필요한 인스턴스를 파악한 다음 다양한 Amazon EC2 인스턴스 패밀리를 살펴봅니다. Amazon은 다음 인스턴스 유형 패밀리를 제공합니다.

- 범용
- 컴퓨팅 최적화
- 메모리 최적화

- 스토리지 최적화
- 가속 컴퓨팅
- HPC 최적화

특정 Amazon EC2 인스턴스 패밀리가 충족할 수 있는 구체적인 목적과 사용 사례를 더 자세히 이해하려면 AWS 인스턴스 유형을 참조하세요.

시스템 요구 사항 수집은 요구 사항에 가장 적합한 구체적인 인스턴스 패밀리와 인스턴스 유형을 선택하는 데 매우 중요합니다. 인스턴스 유형 이름은 패밀리 이름과 인스턴스 크기로 구성됩니다. 예를 들어 t2.micro 인스턴스는 T2 패밀리에 속하며 마이크로 크기입니다.

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 또는 게시된 솔루션)를 사용하여 선택합니다. AWS 요금 계산기 또는 비용 관리 도구를 사용하면 정보를 바탕으로 인스턴스 유형, 크기 및 구성에 대해 결정을 내리는 데 도움이 될 수 있습니다.

구현 단계

- 데이터를 기반으로 리소스 선택: 비용 모델링 데이터를 사용하여 예상 워크로드 사용 수준을 선택하고 지정된 리소스 유형과 크기를 선택합니다. 비용 모델링 데이터를 기반으로 인스턴스에 필요한 데이터 전송 속도를 고려하여 가상 CPU 수, 총 메모리(GiB), 로컬 인스턴스 스토어 볼륨(GB), Amazon EBS 볼륨, 네트워크 성능 수준을 결정합니다. 항상 상세한 분석과 정확한 데이터를 기반으로 선택하여 비용을 효과적으로 관리하는 동시에 성능을 최적화합니다.

리소스

관련 문서:

- AWS 인스턴스 유형
- AWS Auto Scaling
- Amazon CloudWatch의 기능
- Cost Optimization: EC2 Right Sizing

관련 비디오:

- Selecting the right Amazon EC2 instance for your workloads

- Right size your service

관련 예제:

- It just got easier to discover and compare Amazon EC2 instance types

COST06-BP03 지표를 기준으로 리소스 유형, 크기, 개수 자동 선택

현재 실행 중인 워크로드의 지표를 사용하여 비용을 최적화하기에 적합한 크기와 유형을 선택합니다. 컴퓨팅, 스토리지, 데이터 및 네트워킹 서비스의 처리량, 크기 및 스토리지를 적절하게 프로비저닝합니다. 자동 조정 등의 피드백 루프나 워크로드의 사용자 지정 코드로 이 프로비저닝을 수행할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 지침

실행 중인 워크로드에서 나오는 활성 지표를 사용하여 해당 워크로드를 변경하는 피드백 루프를 워크로드 내에 생성합니다. AWS Auto Scaling과 같은 관리형 서비스를 사용하여 규모 조정 작업을 대신 수행하도록 해당 서비스를 구성할 수 있습니다. 또한 AWS에서는 최소한의 노력으로 리소스를 수정할 수 있는 API, SDK 및 기능을 제공합니다. Amazon 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 Amazon Simple Storage Service Intelligent-Tiering과 같은 자동 유형 또는 크기 선택 기능이 기본적으로 포함되어 있습니다. Amazon S3 Intelligent Tiering은 사용 패턴에 따라 자주 접근하고 자주 접근하지 않는 두 개의 접근 계층 간에 자동으로 데이터를 이동합니다.

구현 단계

- 워크로드 지표를 구성하여 관찰성 개선: 워크로드의 핵심 지표를 캡처합니다. 이러한 지표는 워크로드 출력과 같은 고객 경험을 나타내며 CPU 및 메모리 사용량과 같은 리소스 유형 및 크기 간의 차이에 부합합니다. 컴퓨팅 리소스의 경우 성능 데이터를 분석하여 Amazon EC2 인스턴스 크기를 적절하게 조정합니다. 유휴 인스턴스와 사용률이 낮은 인스턴스를 식별합니다. 확인해야 할 핵심 지표는 CPU 사용량 및 메모리 사용률(예를 들어, AWS Compute Optimizer 및 메모리 사용률을 활성화하여 적절한 크기 지정에서 설명한 대로 90%의 시간에 40%의 CPU 사용률)입니다. 4주 동안 최대 CPU 사용량과 메모리 사용률이 40% 미만인 인스턴스를 식별합니다. 이것이 비용을 절감하기 위해 적정하게 크기를 조정하는 인스턴스입니다. Amazon S3와 같은 스토리지 리소스의 경우 Amazon

S3 Storage Lens를 사용할 수 있으며, 이를 통해 기본적으로 버킷 수준에서 다양한 범주의 28개 지표와, 대시보드에서 14일의 기록 데이터를 확인할 수 있습니다. 요약 및 비용 최적화 또는 이벤트를 기준으로 Amazon S3 Storage Lens 대시보드를 필터링하여 특정 지표를 분석할 수 있습니다.

- 적정 크기 조정 권장 사항 보기: 비용 관리 콘솔에서 AWS Compute Optimizer 및 Amazon EC2 적정 크기 조정 도구의 적정 크기 조정 권장 사항을 사용하거나, 리소스의 AWS Trusted Advisor 적정 크기 조정을 검토하여 워크로드를 조정합니다. 서로 다른 리소스의 크기를 적절하게 조정할 때는 올바른 도구를 사용하고 Amazon EC2 인스턴스, AWS 스토리지 클래스 또는 Amazon RDS 인스턴스 유형에 관계없이 적정 크기 조정 지침을 따르는 것이 중요합니다. 스토리지 리소스의 경우 객체 스토리지 사용량, 활동 추세에 대한 가시성을 제공하고 비용 최적화를 위한 실행 가능한 권장 사항을 제시하며 데이터 보호 모범 사례를 적용할 수 있는 Amazon S3 Storage Lens를 사용할 수 있습니다. Amazon S3 Storage Lens가 조직 전체의 지표 분석에서 도출한 상황별 권장 사항을 사용하여 스토리지를 최적화하는 즉각적인 단계를 수행할 수 있습니다.
- 지표를 기준으로 리소스 유형 및 크기 자동 선택: 워크로드 지표를 사용하여 워크로드 리소스를 수동 또는 자동으로 선택합니다. 컴퓨팅 리소스의 경우, AWS Auto Scaling을 구성하거나 애플리케이션 내에서 코드를 구현하면 자주 변경해야 하는 경우 필요한 작업량을 줄이고 수동 프로세스보다 빨리 변경 사항을 구현할 수 있습니다. 단일 Auto Scaling 그룹 내에서 온디맨드 인스턴스 및 스팟 인스턴스 플릿를 자동으로 확장할 수 있습니다. 스팟 인스턴스 사용에 대한 할인을 받을 수 있을 뿐만 아니라 예약 인스턴스 또는 Savings Plans를 사용하여 일반 온디맨드 인스턴스 요금의 할인된 요금을 받을 수 있습니다. 이 모든 요소를 결합하여 Amazon EC2 인스턴스의 비용 절감을 최적화하고 애플리케이션에 대해 원하는 규모와 성능을 결정할 수 있습니다. 또한 Auto Scaling 그룹(ASG)의 속성 기반 인스턴스 유형 선택(ABS) 전략을 사용하여 vCPU, 메모리 및 스토리지와 같은 일련의 속성으로서의 인스턴스 요구 사항을 표현할 수 있습니다. 신세대 인스턴스 유형이 릴리스되면 이를 자동으로 사용하고 Amazon EC2 스팟 인스턴스를 통해 더 광범위한 용량에 액세스할 수 있습니다. Amazon EC2 플릿과 Amazon EC2 Auto Scaling은 지정된 속성에 맞는 인스턴스를 선택하고 시작하여 인스턴스 유형을 수동으로 선택할 필요성이 사라집니다. 스토리지 리소스의 경우, 데이터 액세스 패턴이 변경되면 성능 영향이나 운영 오버헤드 없이 자동으로 스토리지 비용이 절감되는 스토리지 클래스를 자동으로 선택할 수 있는 Amazon S3 Intelligent Tiering 및 Amazon EFS Infrequent Access 기능을 사용할 수 있습니다.

리소스

관련 문서:

- AWS Auto Scaling
- AWS Right-Sizing
- AWS Compute Optimizer

- Amazon CloudWatch의 기능
- CloudWatch Getting Set Up
- CloudWatch Publishing Custom Metrics
- Getting Started with Amazon EC2 Auto Scaling
- Amazon S3 Storage Lens
- Amazon S3 Intelligent-Tiering
- Amazon EFS Infrequent Access
- Launch an Amazon EC2 Instance Using the SDK

관련 비디오:

- Right Size Your Services

관련 예제:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive scaling with Amazon EC2 Auto Scaling
- Amazon S3 Storage Lens로 비용 최적화 및 사용량에 대한 가시성 확보

COST06-BP04 공유 리소스 사용 고려

여러 사업부를 위해 조직 수준에서 이미 배포된 서비스의 경우 공유 리소스를 사용하여 활용도를 높이고 총 소유 비용(TCO)을 줄이는 것을 고려합니다. 공유 리소스는 기존 솔루션을 사용하거나 구성 요소를 공유하거나 두 가지 방법을 모두 사용하여 관리 및 비용을 중앙 집중화하는 비용 효과적인 옵션이 될 수 있습니다. 모니터링, 백업 및 연결과 같은 일반적인 기능을 계정 경계 내에서 또는 전용 계정에서 관리합니다. 또한 표준화를 구현하고, 중복을 줄이고, 복잡성을 줄임으로써 비용을 절감할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 지침

여러 워크로드로 인해 동일한 기능이 발생하는 경우 기존 솔루션과 공유 구성 요소를 사용하여 관리를 개선하고 비용을 최적화합니다. 비프로덕션 데이터베이스 서버 또는 디렉터리 서비스와 같은 기존 리

소스(특히 공유 리소스)를 사용하여 보안 모범 사례 및 조직 규정에 따라 클라우드 비용을 절감하는 것을 고려합니다. 최적의 가치 실현과 효율성을 위해서는 쇼백 및 차지백을 사용하여 소비를 발생시키는 비즈니스의 관련 영역에 비용을 다시 할당하는 것이 중요합니다.

쇼백은 클라우드 비용을 소비자, 사업부, 총계정원장 계정 또는 기타 책임 주체와 같이 비용의 원인이 될 수 있는 범주로 분류한 보고서입니다. 쇼백의 목표는 팀, 사업부 또는 개인에게 사용된 클라우드 리소스의 비용을 보여 주는 것입니다.

차지백은 특정 재무 관리 프로세스에 적합한 전략을 기반으로 중앙 서비스 지출을 비용 단위에 할당하는 것을 의미합니다. 고객의 경우 차지백은 하나의 공유 서비스 계정에서 발생한 비용을 고객 보고 프로세스에 적합한 여러 금융 비용 범주에 청구합니다. 차지백 메커니즘을 설정하여 여러 사업부, 제품 및 팀에서 발생한 비용을 보고할 수 있습니다.

워크로드는 중요한 워크로드와 중요하지 않은 워크로드로 분류할 수 있습니다. 이 분류에 따라 덜 중요한 워크로드에 대해서는 일반 구성이 적용된 공유 리소스를 사용합니다. 비용을 더욱 최적화하려면 중요한 워크로드에 사용할 전용 서버를 예약합니다. 리소스를 공유하거나 여러 계정에 프로비저닝하여 효율적으로 관리합니다. 개발, 테스트 및 프로덕션 환경이 서로 다르더라도 안전한 공유가 가능하며 조직 구조를 변경하지 않아도 됩니다.

컨테이너식 애플리케이션에 대한 이해도를 높이고 비용 및 사용을 최적화하려면 애플리케이션이 공유 컴퓨팅 및 메모리 리소스를 사용하는 방식을 기반으로 개별 비즈니스 엔터티에 비용을 할당하는 데 도움이 되는 분할 비용 할당 데이터를 사용합니다. 분할 비용 할당 데이터를 사용하면 Amazon Elastic Container Service(Amazon ECS) 또는 Amazon Elastic Kubernetes Service(Amazon EKS)에서 실행되는 컨테이너 워크로드에서 작업 수준의 쇼백 및 차지백을 달성할 수 있습니다.

분산 아키텍처의 경우 각 VPC의 워크로드에 필요한 공유 서비스에 대한 중앙 집중식 액세스를 제공하는 공유 서비스 VPC를 구축합니다. 이러한 공유 서비스에는 디렉터리 서비스 또는 VPC 엔드포인트와 같은 리소스가 포함될 수 있습니다. 관리 오버헤드와 비용을 줄이려면 각 VPC에 리소스를 구축하는 대신 중앙 위치에서 리소스를 공유합니다.

공유 리소스를 사용하면 운영 비용을 절감하고 리소스 활용도를 극대화하며 일관성을 개선할 수 있습니다. 다중 계정 설계에서는 일부 AWS 서비스를 중앙에서 호스팅하고 허브에 있는 여러 애플리케이션 및 계정을 사용하여 이러한 서비스에 액세스하면 비용을 절감할 수 있습니다. AWS Resource Access Manager(AWS RAM)를 사용하여 VPC 서브넷 및 AWS Transit Gateway Attachment, AWS Network Firewall 또는 Amazon SageMaker AI 파이프라인과 같은 기타 공통 리소스를 공유할 수 있습니다. 다중 계정 환경에서 AWS RAM을 사용하면 리소스를 한 번 생성하여 다른 계정과 공유할 수 있습니다.

조직은 공유 비용에 효과적으로 태그를 지정하고 비용 중 상당 부분에 태그가 지정되지 않았거나 할당되지 않은 상태는 아닌지 확인해야 합니다. 공유 비용을 효과적으로 할당하지 않고 공유 비용 관리를

책임지는 사람이 없는 경우 공유 클라우드 비용이 급증할 수 있습니다. 리소스, 워크로드, 팀 또는 조직 수준에서 비용이 발생한 부분을 알아야 합니다. 이러한 지식이 있다면 달성한 비즈니스 성과와 비교할 때 해당 수준에서 얼마나 큰 가치가 제공되었는지 더욱 정확하게 이해할 수 있기 때문입니다. 궁극적으로 조직은 클라우드 인프라 공유를 통해 비용을 절감할 수 있습니다. 공유 클라우드 리소스에 대한 비용 할당을 장려하여 클라우드 지출을 최적화합니다.

구현 단계

- 기존 리소스 평가: 워크로드에 유사한 서비스를 사용하는 기존 워크로드를 검토합니다. 워크로드의 구성 요소에 따라 비즈니스 논리 또는 기술 요구 사항이 허용하는 경우 기존 플랫폼을 고려합니다.
- AWS RAM에서 리소스 공유 사용 및 그에 따라 제한: 조직 내 다른 AWS 계정과 리소스를 공유하는 데 AWS RAM를 사용합니다. 리소스를 공유할 때 여러 계정에 리소스를 복제할 필요가 없으므로 리소스 유지 관리에 따른 운영 부담이 최소화됩니다. 이 프로세스는 또한 생성한 리소스를 계정의 역할 및 사용자는 물론 다른 AWS 계정과 안전하게 공유하는 데 도움이 됩니다.
- 리소스에 태그 지정: 비용 보고 후보인 리소스에 태그를 지정하고 비용 범주 내에서 분류합니다. 비용 할당을 위해 이러한 비용 관련 리소스 태그를 활성화하여 AWS 리소스 사용을 명확하게 확인할 수 있도록 합니다. 비용 및 사용 가시성을 적절한 수준으로 세분화하는 데 집중하고, 비용 할당 보고 및 KPI 추적을 통해 클라우드 소비 행동에 영향을 미칩니다.

리소스

관련 모범 사례:

- SEC03-BP08 안전하게 조직과 리소스 공유

관련 문서:

- What is AWS Resource Access Manager?
- AWS services that you can use with AWS Organizations
- Shareable AWS resources
- AWS Cost and Usage (CUR) Queries

관련 비디오:

- AWS Resource Access Manager - granular access control with managed permissions
- How to design your AWS cost allocation strategy

- AWS Cost Categories

관련 예제:

- How-to chargeback shared services: An AWS Transit Gateway example
- How to build a chargeback/showback model for Savings Plans using the CUR
- Using VPC Sharing for a Cost-Effective Multi-Account Microservice Architecture
- Improve cost visibility of Amazon EKS with AWS Split Cost Allocation Data
- AWS 분할 비용 할당 데이터를 사용하여 Amazon ECS 및 AWS Batch의 비용 가시성 향상

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. 2주에서 1개월의 정기적인 주기로 분석을 수행하는 것이 좋습니다. 이 분석을 통해 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 찾아보세요.

요금 모델: AWS는 조직의 요구 사항에 적합한 가장 비용 효율적인 방법으로 리소스 요금을 지불할 수 있는 다수의 요금 모델을 제공합니다. 다음 섹션에서 아래의 각 구매 모델에 대해 설명합니다.

- 온디맨드 인스턴스
- 스팟 인스턴스
- 약정 할인 - 절감형 플랜
- 약정 할인 - 예약형 인스턴스/용량

- 지리적 선택
- 서드파티 계약 및 요금

온디맨드 인스턴스: 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 1초 단위로 청구될 수 있습니다(예: Amazon RDS 또는 Linux EC2 인스턴스). 정기적으로 사용량이 급증하는 단기 워크로드(예: 4개월 동안 진행되는 프로젝트)나 중단할 수 없는 예측이 불가능한 워크로드에는 온디맨드 모델을 사용하는 것이 좋습니다. 온디맨드는 무중단 런타임이 요구되지만 약정 할인(절감형 플랜 또는 예약형 인스턴스)을 받을 수 있을 만큼 오래 실행되지 않는 사전 프로덕션 환경과 같은 워크로드에도 적합합니다.

스팟 인스턴스: 스팟 인스턴스는 온디맨드 요금에 비해 최대 90% 할인된 요금으로 장기 약정 없이 제공되는 예비 Amazon EC2 컴퓨팅 용량입니다. 스팟 인스턴스를 사용하면 애플리케이션 실행 비용을 대폭 절감하거나 동일한 예산으로 애플리케이션의 컴퓨팅 용량을 확장할 수 있습니다. 온디맨드와 달리, 스팟 인스턴스는 Amazon EC2에서 용량을 다시 확보해야 하거나 스팟 인스턴스 가격이 구성된 가격을 초과하는 경우 2분 전 경고와 함께 인스턴스가 중단될 수 있습니다. 평균적으로 스팟 인스턴스가 중단되는 비율은 해당 시간의 5% 미만입니다.

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

스팟 인스턴스는 여러 AWS 서비스(예: Amazon EC2 Auto Scaling 그룹, Amazon EMR, Amazon Elastic Container Service(Amazon ECS), AWS Batch)에도 통합됩니다.

스팟 인스턴스를 회수해야 하는 경우 Amazon EC2는 CloudWatch Events와 인스턴스 메타데이터를 통해 전달되는 스팟 인스턴스 중단 공지를 통해 2분 전에 경고를 보냅니다. 애플리케이션은 이 2분의 시간을 사용하여 상태를 저장하거나, 실행 중인 컨테이너를 비우거나, 최종 로그 파일을 업로드하거나, 로드 밸런서에서 애플리케이션 자체를 제거할 수 있습니다. 2분 후에는 스팟 인스턴스를 최대 절전 모드로 전환하거나, 중지하거나, 종료할 수 있습니다.

워크로드에 스팟 인스턴스를 채택할 때는 다음 모범 사례를 고려합니다.

- 가능한 한 많은 인스턴스 유형을 유연하게 사용: 인스턴스 유형의 패밀리와 크기를 유연하게 사용하여 타겟 용량 요구 사항을 충족할 가능성을 높이고, 가능한 최저 비용을 확보하며, 중단의 영향을 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 절감형 플랜/예약형 인스턴스와 함께 사용하는 것이 좋습니다.

약정 할인 – 절감형 플랜: AWS는 특정 양의 리소스 사용을 예약하거나 약정하고 리소스에 대한 할인된 요금을 받아 비용을 절감할 수 있는 다양한 방법을 제공합니다. 절감형 플랜을 사용하면 1년 또는 3년에 대한 시간당 지출 약정을 체결하고 리소스 전체에 할인된 요금을 적용할 수 있습니다. 절감형 플랜은 Amazon EC2, AWS Fargate, AWS Lambda와 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 USD를 약정하고 온디맨드 사용량이 시간당 150 USD인 경우를 고려합니다. 절감형 플랜 요금을 고려하면 특정 사용량에 대해 50% 할인율이 적용됩니다. 따라서 50 USD의 약정에 100 USD의 온디맨드 사용량이 포함됩니다. 결과적으로는 50 USD(약정)와 나머지 온디맨드 사용량에 대한 50 USD를 지불하게 됩니다.

컴퓨팅 절감형 플랜은 가장 유연한 요금제로, 최대 66%의 할인을 제공합니다. 가용 영역, 인스턴스 크기, 인스턴스 패밀리, 운영 체제, 테넌시, 리전 및 컴퓨팅 서비스에 자동으로 적용됩니다.

인스턴스 절감형 플랜은 덜 유연하지만 할인율은 더 높습니다(최대 72%). 가용 영역, 인스턴스 크기, 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 기간에 약정에 포함되는 사용량에 대한 추가 요금이 발생하지 않습니다.

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

절감형 플랜은 할인율이 높은 순서에서 낮은 순서로 구매 계정의 사용량에 먼저 적용된 후 다른 모든 계정의 통합 사용량에 할인율이 높은 순서에서 낮은 순서로 적용됩니다.

모든 절감형 플랜은 관리 계정과 같은 사용량이나 리소스가 없는 계정에서 구입하는 것이 좋습니다. 이렇게 하면 절감형 플랜은 모든 사용량에 걸쳐 가장 높은 할인율이 적용되므로 할인 금액을 최대화할 수 있습니다.

워크로드와 사용량은 시간 경과에 따라 변경되는 것이 일반적입니다. 소액의 절감형 플랜을 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 않습니다. 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 절감형 플랜 권장 사항 도구에서 가능한 잠재적 절감액을 추적하고 모니터링하는 것이 좋습니다. Cost Explorer에서 절감형 플랜 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 않습니다. 이 비율이 늘어난다고 해서 절감액도 커지는 것은 아닙니다. 절감형 플랜을 구매할 경우 적용 범위가 증가하는지 확인합니다. 적용 범위 또는 사용률이 감소하면 이를 정량화하고 숙지합니다. 예를 들어 워크로드 리소스를 새로운 인스턴스 유형으로 마이그레이션할 경우 기존 요금제의 사용률은 감소하지만 성능상의 이점이 절감액을 능가합니다.

약정 할인 – 예약형 인스턴스/약정: 절감형 플랜과 마찬가지로, 예약형 인스턴스(RI)는 최소량의 리소스 실행을 약정할 경우 최대 72%의 할인을 제공합니다. 예약형 인스턴스는 Amazon RDS, Amazon OpenSearch Service, Amazon ElastiCache, Amazon Redshift, DynamoDB에서 사용 가능합니다. Amazon CloudFront와 AWS Elemental MediaConvert는 최소 사용량 약정을 체결할 경우 할인을 제공합니다. 예약 인스턴스는 현재 Amazon EC2에서 사용할 수 있지만, 절감형 플랜은 향상된 유연성과 관리 오버헤드 없이 동일한 수준의 할인을 제공합니다.

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

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

Amazon EC2에서는 컨버터블 RI를 사용할 수 있지만 모든 EC2 인스턴스에는 향상된 유연성과 운영 비용 절감을 제공하는 절감형 플랜을 사용해야 합니다.

예약형 인스턴스를 추적하고 구매할 때도 동일한 프로세스와 지표를 사용해야 합니다. 계정 전체에서 RI 적용 범위를 추적하는 것은 권장되지 않습니다. 또한 사용률을 모니터링하거나 추적하지 않는 것이 좋습니다. 대신 Cost Explorer에서 사용률 보고서를 확인하고 테이블의 순 절감액 열을 사용합니다. 순 절감액이 아주 큰 음수인 경우 미사용 RI를 수정하기 위한 조치를 취해야 합니다.

EC2 플릿: EC2 플릿을 사용하면 타겟 컴퓨팅 용량을 정의한 다음, 인스턴스 유형 및 해당 플릿에 맞게 조정된 온디맨드와 스팟 인스턴스를 지정합니다. 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 글로벌 인프라는 리전 및 가용 영역을 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.

서드파티 계약 및 요금: 클라우드에서 서드파티 솔루션 또는 서비스를 사용하는 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 Amazon 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.

모범 사례

- COST07-BP01 요금 모델 분석 수행
- COST07-BP02 비용을 기준으로 리전 선택
- COST07-BP03 비용 효율적인 조건을 갖춘 서드파티 계약 선택

- COST07-BP04 워크로드의 모든 구성 요소에 대한 요금 모델 구현
- COST07-BP05 관리 계정 수준에서 요금 모델 분석 수행

COST07-BP01 요금 모델 분석 수행

워크로드의 각 구성 요소를 분석합니다. 구성 요소와 리소스가 장기간 실행되는지(약정 할인의 경우) 아니면 동적으로 단기간 실행되는지(스팟 또는 온디맨드의 경우) 결정합니다. 비용 관리 도구의 권장 사항을 바탕으로 워크로드를 분석하고 높은 수익을 달성할 수 있도록 해당 권장 사항에 비즈니스 규칙을 적용합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

AWS는 조직의 요구 사항에 적합하며 제품에 맞게 가장 비용 효율적인 방법으로 리소스 요금을 지불할 수 있는 다수의 요금 모델을 제공합니다. 팀과 협력하여 가장 적합한 요금 모델을 결정하세요. 요금 모델은 가용성에 따라 결정되는 여러 옵션의 조합으로 구성되는 경우가 많습니다.

온디맨드 인스턴스를 사용하면 실행 중인 인스턴스에 따라 시간별 또는 초 단위(최소 60초)로 컴퓨팅 또는 데이터베이스 용량에 대해 사용한 비용을 지불할 수 있습니다. 장기 약정이나 선불 비용 결제는 필요하지 않습니다.

절감형 플랜은 1년 또는 3년 동안 일정한 사용량(시간당 USD로 측정)을 약정하는 대가로, Amazon EC2, Lambda 및 AWS Fargate에서의 사용량을 낮은 가격으로 제공하는 유연한 요금 모델입니다.

스팟 인스턴스는 사전 약정 없이 시간당 온디맨드 가격의 최대 90% 할인된 가격으로 예비 컴퓨팅 용량을 요청할 수 있는 Amazon EC2 요금 메커니즘입니다.

예약형 인스턴스는 최대 75%까지 할인된 가격으로 용량을 준비할 수 있도록 지원합니다. 자세한 내용은 예약을 통한 비용 최적화를 참조하세요.

프로덕션, 품질 및 개발 환경과 관련된 리소스에 대해 절감형 플랜을 포함하도록 선택할 수 있습니다. 또는 샌드박스 리소스는 필요할 때만 켜지므로 해당 환경의 리소스에 대한 온디맨드 모델을 선택할 수도 있습니다. Amazon 스팟 인스턴스를 사용하여 Amazon EC2 비용을 절감하거나 컴퓨팅 절감형 플랜을 사용하여 Amazon EC2, Fargate 및 Lambda 비용을 절감할 수 있습니다. AWS Cost Explorer 권장 사항 도구는 절감형 플랜을 통해 약정 할인을 받을 수 있는 기회를 제공합니다.

이전에 Amazon EC2용 예약형 인스턴스를 구매했거나 조직 내에서 비용 할당 관행을 마련한 경우에는 한동안 Amazon EC2 예약형 인스턴스를 계속 사용할 수 있습니다. 그러나 향후 보다 유연한 비용 절감 메커니즘으로 절감형 플랜을 사용할 전략을 수립할 것을 권장합니다. AWS Cost Management에

서 절감형 플랜(SP) 권장 사항을 새로 고쳐 언제든지 최신 절감형 플랜 권장 사항을 생성할 수 있습니다. 예약형 인스턴스(RI)를 사용하여 Amazon RDS, Amazon Redshift, Amazon ElastiCache, Amazon OpenSearch Service 비용을 절감합니다. 절감형 플랜 및 예약형 인스턴스는 전액 선결제, 부분 선결제, 선결제 없음이라는 3가지 옵션으로 제공됩니다. AWS Cost Explorer RI 및 SP 구매 권장 사항에 나와 있는 권장 사항을 참조하세요.

스팟 워크로드 기회를 찾으려면 전체 사용량을 시간대별로 확인하여 사용량 또는 탄력성이 주기적으로 변경되는 기간을 찾아보세요. 다양한 내결함성 및 유연한 애플리케이션에 스팟 인스턴스를 사용할 수 있습니다. 예를 들어, 상태 비저장 웹 서버, API 엔드포인트, 빅 데이터 및 분석 애플리케이션, 컨테이너형 워크로드, CI/CD 및 기타 유연한 워크로드 등이 있습니다.

사용하지 않을 때(업무 시간 이후 및 주말) Amazon EC2 및 Amazon RDS 인스턴스를 해제할 수 있는지를 분석하세요. 이 전략을 활용하면 연중무휴로 사용할 때보다 비용을 70% 이상 절감할 수 있습니다. 특정 시간에만 사용해야 하는 Amazon Redshift 클러스터가 있는 경우 클러스터를 일시 중지했다가 나중에 다시 시작하면 됩니다. Amazon Redshift 클러스터 또는 Amazon EC2 및 Amazon RDS 인스턴스가 중지되면 컴퓨팅 과금 청구가 정지되고 스토리지 요금만 부과됩니다.

단, 온디맨드 용량 예약(ODCR)은 요금 할인이 아닙니다. 예약 용량에서 인스턴스를 실행하는지 여부와 무관하게 동등한 온디맨드 요금이 용량 예약에 청구됩니다. 실행하려는 리소스에 용량을 충분히 제공해야 하는 경우 이를 고려해야 합니다. ODCR은 더 이상 필요하지 않을 때 취소할 수 있기 때문에 장기 약정에 얽매이지 않으면서 절감형 플랜 또는 예약형 인스턴스가 제공하는 할인 혜택도 누릴 수 있습니다.

구현 단계

- 워크로드 탄력성 분석: Cost Explorer 또는 사용자 지정 대시보드에서 시간 단위의 세부 수준을 사용하여 워크로드 탄력성을 분석합니다. 실행 중인 인스턴스 개수의 정기적인 변경 사항을 확인합니다. 기간이 짧은 인스턴스가 스팟 인스턴스 또는 스팟 플릿에 적합합니다.
  - Well-Architected Lab: Cost Explorer
  - Well-Architected Lab: Cost Visualization
- 기존 요금 계약 검토: 장기적인 요구 사항에 대해 현재 계약 또는 약정을 검토합니다. 현재 사용 중인 약정과 이러한 약정을 얼마나 사용하고 있는지 분석합니다. 기존의 계약상 할인 또는 엔터프라이즈 계약을 활용합니다. 엔터프라이즈 계약은 고객의 요구에 가장 적합한 계약을 맞춤 설정할 수 있는 옵션을 제공합니다. 장기 약정의 경우 특정 인스턴스 유형, 인스턴스 패밀리, AWS 리전, 가용 영역에 대해 특정 인스턴스에 대한 절감형 플램 또는 예약형 인스턴스, 예약 요금 할인을 고려하세요.
- 약정 할인 분석 수행: 계정에서 Cost Explorer를 사용하여 절감형 플랜 및 예약 인스턴스 권장 사항을 검토합니다. 필요한 할인 및 위험과 함께 올바른 권장 사항을 구현하려면 Well-Architected Labs을 따르세요.

리소스

관련 문서:

- Accessing Reserved Instance recommendations
- 인스턴스 구입 옵션
- AWS Enterprise

관련 비디오:

- Save up to 90% and run production workloads on Spot

관련 예제:

- Well-Architected Lab: Cost Explorer
- Well-Architected Lab: Cost Visualization
- Well-Architected Lab: Pricing Models

COST07-BP02 비용을 기준으로 리전 선택

리소스 요금은 리전별로 다를 수 있습니다. 리전별 비용 차이를 식별하고 지연 시간, 데이터 상주 및 데이터 주권 요구 사항을 충족하기 위해 필요한 경우에만 비용이 더 높은 지역에서 배포합니다. 따라서 리전 비용을 고려하면 이 워크로드의 전체 가격을 최저 수준으로 낮출 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 지침

글로벌한 AWS 클라우드 인프라는 전 세계 여러 위치에서 호스트되며, AWS 리전, 가용 영역, 로컬 영역, AWS Outposts, Wavelength 영역을 중심으로 구축되었습니다. 리전은 전 세계에 있는 실제 위치이고 각 리전은 AWS에 가용 영역이 여러 개 있는 개별 지리적 영역입니다. 각 리전 내의 여러 분리된 위치인 가용 영역은 이중화된 전력, 네트워킹 및 연결 기능을 갖추고 있는 하나 이상의 개별 데이터 센터로 구성됩니다.

각 AWS 리전은 현지 시장 조건 내에서 운영되며 예를 들어, 리소스 요금은 토지, 광섬유, 전기 및 세금 비용의 차이 때문에 리전마다 다릅니다. 전 세계에서 사용 가능한 최저 가격으로 리소스를 실행할 수 있도록 솔루션의 한 구성 요소나 전체 솔루션을 운영할 특정 리전을 선택합니다. AWS 계산기를 사용

하여 위치 유형(리전, Wavelength 영역 및 로컬 영역) 및 리전별로 서비스를 검색해 다양한 리전에서 워크로드 비용을 예측합니다.

솔루션 설계 시의 모범 사례는 사용자와 더 가까운 위치에 컴퓨팅 리소스를 배치하여 지연 시간을 줄이고 데이터 주권을 강화하는 것입니다. 비즈니스, 개인정보 처리방침, 성능, 보안 요구 사항에 따라 지리적 위치를 선택합니다. 전 세계에 최종 사용자가 분포하는 애플리케이션의 경우 여러 위치를 사용합니다.

개인정보 처리방침, 보안 및 비즈니스 요구 사항에 대한 의무가 없는 경우 AWS 서비스에 대해 더 저렴한 요금을 제공하는 리전을 사용하여 워크로드를 배포합니다. 예를 들어, 기본 리전이 아시아 태평양(시드니)(`ap-southwest-2`)이고 다른 리전 사용에 대한 제한 사항(예: 개인정보 처리방침, 보안)이 없는 경우, 개발이나 테스트처럼 중요하지 않은 Amazon EC2 인스턴스는 미국 동부(버지니아 북부)(`us-east-1`) 리전에 배포하면 비용을 줄일 수 있습니다.

	규정 준수	지연 시간	비용	서비스/기능
리전 1	✓	15ms	$$	✓
리전 2	✓	20ms	$$$	X
리전 3	✓	80ms	$	✓
리전 4	✓	15ms	$$	✓
리전 5	✓	20ms	$$$	X
리전 6	✓	15ms	$	✓
리전 7	✓	80ms	$	✓
리전 8	✓	15ms	$	X

리전별 기능 매트릭스 표

위의 표는 리전 6이 다른 리전에 비해 지연 시간이 짧고 서비스를 이용할 수 있으며 비용이 가장 저렴한 리전이기 때문에 이 시나리오에 가장 적합한 옵션임을 보여줍니다.

구현 단계

- AWS 리전 요금 검토: 현재 리전의 워크로드 비용을 분석합니다. 서비스 및 사용 유형별로 가장 높은 비용부터 시작하여 사용 가능한 다른 리전의 비용을 계산합니다. 예상 절감액이 구성 요소 또는 워크로드 이동 비용보다 큰 경우 새 리전으로 마이그레이션합니다.
- 다중 리전 배포에 대한 요구 사항 검토: 비즈니스 요구 사항 및 의무(개인정보 처리방침, 보안 또는 성능)를 분석하여 여러 리전을 사용하면 안 되는 제한 사항이 있는지 확인합니다. 단일 리전을 사용하도록 제한하는 의무가 없는 경우 다중 리전을 사용합니다.
- 필요한 데이터 전송 분석: 리전을 선택할 때 데이터 전송 비용을 고려합니다. 데이터는 고객 및 리소스 가까운 곳에 둡니다. 데이터가 흐르고 데이터 전송이 최소화된 더 저렴한 AWS 리전을 선택합니다. 데이터 전송에 대한 비즈니스 요구 사항에 따라 Amazon CloudFront, AWS PrivateLink, AWS Direct Connect, AWS Virtual Private Network를 사용하여 네트워킹 비용을 줄이고 성능을 개선하며 보안을 강화할 수 있습니다.

리소스

관련 문서:

- Accessing Reserved Instance recommendations
- Amazon EC2 요금 정책
- 인스턴스 구입 옵션
- 리전 테이블

관련 비디오:

- Save up to 90% and run production workloads on Spot

관련 예제:

- Overview of Data Transfer Costs for Common Architectures
- Cost Considerations for Global Deployments
- What to Consider when Selecting a Region for your Workloads

COST07-BP03 비용 효율적인 조건을 갖춘 서드파티 계약 선택

비용 효율적인 계약과 조건은 이러한 서비스의 비용이 제공하는 혜택에 따라 규모가 조정됨을 보장합니다. 조직에 추가적인 혜택을 제공할 때 조정되는 계약 및 요금을 선택합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

시장에는 클라우드 환경에서 비용을 관리하는 데 도움이 되는 여러 제품이 출시되어 있습니다. 제품마다 고객 요구 사항에 따라 기능이 약간씩 다를 수 있습니다. 예를 들어 비용 거버넌스 또는 비용 가시성에 초점을 맞춘 제품도 있고 비용 최적화에 초점을 맞춘 제품도 있습니다. 효과적인 비용 최적화 및 거버넌스를 위해서는 필요한 기능과 적절한 요금 모델을 갖춘 올바른 도구를 사용하는 것이 매우 중요합니다. 이러한 제품의 요금 모델은 다양합니다. 월 청구액의 일정 비율을 청구하기도 하고 실현된 절감액의 일정 비율을 청구하기도 합니다. 사실 필요한 기능에 대해서만 비용을 지불하는 것이 가장 좋습니다.

클라우드에서 서드파티 솔루션 또는 서비스를 사용할 때는 원하는 성과에 맞는 요금 구조가 중요합니다. 요금은 제공하는 결과와 가치에 따라 조정되어야 합니다. 예를 들어 제공하는 절감액의 일정 비율에 비례하여 절감액(결과)이 커질수록 더 많은 요금을 부과하는 소프트웨어가 있습니다. 지출 증가에 따라 더 많은 비용을 지불하는 라이선스 계약이 비용 최적화에 항상 가장 적합한 것은 아닙니다. 그러나 공급업체가 청구서의 모든 부분에 대해 명확한 혜택을 제공한다면 조정되는 방식의 이러한 수수료가 정당화될 수 있습니다.

예를 들어 Amazon EC2에 대한 권장 사항을 제공하고 전체 청구 금액의 일정 비율을 부과하는 솔루션의 경우 혜택을 제공하지 않는 다른 서비스를 사용하면 솔루션 요금이 더 비싸집니다. 또 다른 예로는 관리형 리소스 비용의 일정 비율로 요금을 부과하는 관리형 서비스가 있습니다. 인스턴스 크기가 크다고 해서 반드시 관리에 더 많은 노력이 필요한 것은 아니지만 부과되는 요금은 증가할 수 있습니다. 따라서 이러한 서비스 요금 방식에 서비스의 효율성을 증진하는 비용 최적화 프로그램 또는 기능이 포함되어 있는지 확인해야 합니다.

고객은 시장에서 이러한 제품이 더 고급이거나 사용하기 더 쉽다고 생각할 수 있습니다. 이러한 제품의 비용을 고려하고 장기적인 관점에서 잠재적 비용 최적화 결과에 대해 생각해야 합니다.

구현 단계

- 서드파티 계약 및 조건 분석: 서드파티 계약의 요금을 검토합니다. 다양한 사용량에 대해 모델링을 수행하고, 새로운 서비스 사용량과 같은 새로운 비용 또는 워크로드 증가로 인한 현재 서비스의 증가를 고려합니다. 추가 비용이 비즈니스에 필요한 이점을 제공하는지 여부를 파악합니다.

리소스

관련 문서:

- Accessing Reserved Instance recommendations
- 인스턴스 구입 옵션

관련 비디오:

- Save up to 90% and run production workloads on Spot

COST07-BP04 워크로드의 모든 구성 요소에 대한 요금 모델 구현

영구 실행되는 리소스는 절감형 플랜 또는 예약형 인스턴스와 같은 예약 용량을 활용해야 합니다. 단기간 용량은 스팟 인스턴스나 스팟 플릿을 사용하도록 구성됩니다. 온디맨드 인스턴스는 예약 용량을 사용할 만큼 충분히 오래 실행되지 않으며(리소스 유형에 따라 25~75%에 해당하는 기간에 실행됨) 중단할 수 없는 단기 워크로드에만 사용됩니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

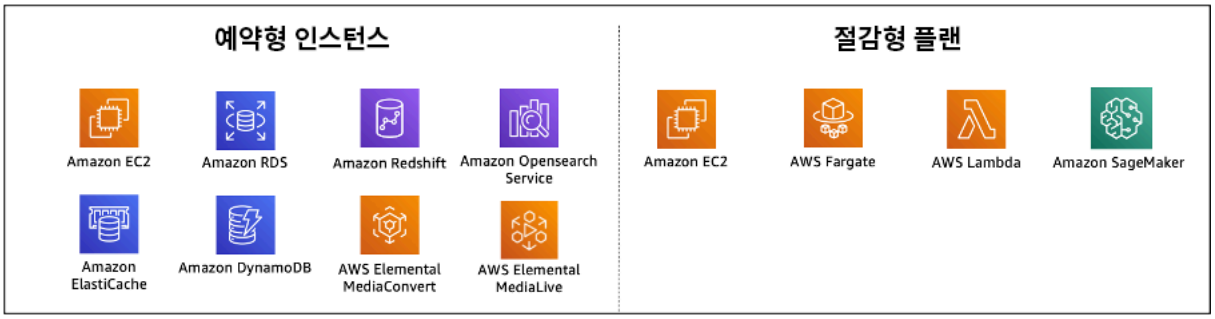
구현 가이드

AWS는 비용 효율성을 개선하기 위해 과거 사용량을 기반으로 여러 약정 권장 사항을 제공합니다. 이러한 권장 사항을 통해 절약할 수 있는 항목과 약정 사용 방법을 파악할 수 있습니다. 이러한 서비스를 온디맨드 또는 스팟으로 사용하거나 예약형 인스턴스(RI) 및 절감형 플랜(SP)을 통해 일정 기간에 대한 약정을 체결하고 온디맨드 비용을 줄일 수 있습니다. 워크로드를 최적화하려면 각 워크로드 구성 요소와 여러 AWS 서비스뿐만 아니라 이러한 서비스에 대한 약정 할인, 구매 옵션 및 스팟 인스턴스도 이해해야 합니다.

워크로드 구성 요소의 요구 사항을 고려하고 이러한 서비스에 대한 각기 다른 요금 모델을 이해합니다. 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.

예를 들어, AWS에서 이 웹 애플리케이션 아키텍처를 살펴보겠습니다. 이 샘플 워크로드는 Amazon Route 53, AWS WAF, Amazon CloudFront, Amazon EC2 인스턴스, Amazon RDS 인스턴스, 로드 밸런서, Amazon S3 스토리지, Amazon Elastic File System(Amazon EFS) 등 여러 개의 AWS 서비스로

구성됩니다. 이러한 각 서비스를 검토하고 다양한 요금 모델을 사용하여 잠재적인 비용 절감의 기회를 파악해야 합니다. 어떤 서비스는 RI 또는 SP에 적합할 수 있지만 어떤 서비스는 온디맨드로만 사용할 수 있습니다. 다음 이미지에서 볼 수 있듯이 일부 AWS 서비스는 RI 또는 SP를 사용하여 약정할 수 있습니다.



예약형 인스턴스 및 절감형 플랜을 사용하여 약정된 AWS 서비스

구현 단계

- 요금 모델 구현: 분석 결과를 사용하여 절감형 플랜, 예약형 인스턴스를 구매하거나 스팟 인스턴스를 구현합니다. 첫 약정 구매인 경우 목록에서 상위 5개 또는 10개의 권장 사항을 선택한 다음 한두 달 동안의 결과를 모니터링하고 분석하세요. AWS Cost Management Console에서 해당 프로세스를 ㅇ나내합니다. 콘솔에서 RI 또는 SP 권장 사항을 검토하고 권장 사항을 사용자 지정(유형, 결제 및 기간)하고 시간별 약정(예: 시간당 20 USD)을 검토한 다음, 장바구니에 추가합니다. 할인은 적격 사용량에 자동으로 적용됩니다. 예를 들어 정기적으로(2주마다 또는 매월) 약정 할인을 소량 구매합니다. 중단될 수 있거나 상태 정보를 저장하지 않는 워크로드에 대해 스팟 인스턴스를 구현합니다. 마지막으로 온디맨드 Amazon EC2 인스턴스를 선택하고 나머지 요구 사항에 맞게 리소스를 할당합니다.
- 워크로드 검토 주기: 요금 모델 적용 범위를 구체적으로 분석하는 워크로드에 대한 검토 주기를 구현합니다. 워크로드가 필요한 적용 범위에 도달하면 몇 달마다 또는 조직 사용량이 바뀔 때 부분적으로 추가 약정 할인을 구매합니다.

리소스

관련 문서:

- Understanding your Savings Plans recommendations
- Accessing Reserved Instance recommendations
- 예약형 인스턴스 구매 방법
- 인스턴스 구입 옵션
- 스팟 인스턴스

- 다른 AWS 서비스의 예약 모델
- Savings Plans Supported Services

관련 비디오:

- Save up to 90% and run production workloads on Spot

관련 예제:

- 절감형 플랜을 구매하기 전에 고려해야 할 사항은 무엇인가요?
- How can I use Cost Explorer to analyze my spending and usage?

COST07-BP05 관리 계정 수준에서 요금 모델 분석 수행

과금 정보 및 비용 관리 도구를 확인하고 관리 계정 수준에서 정기적인 분석을 수행하기 위해 권장되는 할인 혜택과 약정, 예약을 확인하세요.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 지침

정기적인 비용 모델링을 수행하면 여러 워크로드에 걸친 최적화 기회를 구현할 수 있습니다. 예를 들어, 전반적으로 여러 워크로드에 온디맨드 인스턴스를 사용하는 경우 변경 위험이 낮아지고 약정 기반 할인을 구현하여 전반적인 비용을 절감할 수 있습니다. 2주에서 1개월의 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 관리 계정에서 약정 할인의 기회를 찾을 수 있습니다. 관리 계정 수준의 권장 사항은 예약형 인스턴스(RI) 또는 절감형 플랜(SP)이 있는 AWS 조직 내 모든 계정의 사용량을 고려하여 계산됩니다. 또한 할인 공유가 활성화될 때 계산되어 계정 전체에서 비용 절감을 극대화하는 약정을 추천할 수 있습니다.

관리 계정 수준에서 구매하면 대부분의 경우 최대한 비용을 절감할 수 있지만 특정 연결 계정의 사용량에 할인을 먼저 적용하려는 경우 등 연결 계정 수준에서 SP를 구매하는 것을 고려해야 하는 상황도 있을 수 있습니다. 멤버 계정 권장 사항은 개별 계정 수준에서 계산되어 각 계정의 절감 효과를 극대화합니다. 계정에 RI 약정과 SP 약정이 둘 다 있는 경우 다음 순서로 적용됩니다.

1. 영역 RI

2. 스탠다드 RI
3. 컨버터블 RI
4. 인스턴스 절감형 플랜
5. 컴퓨팅 절감형 플랜

관리 계정 수준에서 SP를 구매하면 절감액이 가장 높은 할인율에서 가장 낮은 할인율 순으로 적용됩니다. 관리 계정 수준의 SP는 연결된 모든 계정을 살펴보고 할인율이 가장 높은 곳에 절감액을 적용합니다. 할인 적용 대상을 제한하려는 경우 연결 계정 수준에서 절감형 플랜을 구매하면 해당 계정에서 적격 컴퓨팅 서비스를 실행할 때마다 할인이 먼저 적용됩니다. 해당 계정에서 적합한 컴퓨팅 서비스를 실행하지 않는 경우 할인은 동일한 관리 계정 내의 다른 연결된 계정과 공유됩니다. 할인 공유는 기본적으로 활성화되어 있지만 필요한 경우 비활성화할 수 있습니다.

통합 결제 패밀리에서 절감형 플랜은 먼저 소유자 계정의 사용량에 적용된 다음, 다른 계정의 사용량에 적용됩니다. 이는 공유를 활성화한 경우에만 발생합니다. 절감형 플랜은 가장 높은 절감률에 먼저 적용됩니다. 절감률이 같은 사용량이 여러 개 있는 경우 절감형 플랜 요금이 가장 낮은 첫 번째 사용량에 절감형 플랜이 적용됩니다. 절감형 플랜은 남은 사용량이 더 이상 없거나 약정이 소진될 때까지 계속 적용됩니다. 나머지 사용량은 온디맨드 요금에서 부과됩니다. AWS 비용 관리에서 절감형 플랜 권장 사항을 새로 고쳐 언제든지 최신 절감형 플랜 권장 사항을 생성할 수 있습니다.

인스턴스의 유연성을 분석한 후 권장 사항에 따라 약정할 수 있습니다. 다양한 잠재 리소스 옵션으로 워크로드의 단기 비용을 분석하고, AWS 요금 모델을 분석하며, 이를 비즈니스 요구 사항에 맞춰 조정하여 총 소유 비용을 확인하고 비용 최적화 기회를 모색합니다.

구현 단계

약정 할인 분석 수행: 계정에서 Cost Explorer를 사용하여 절감형 플랜 및 예약형 인스턴스 권장 사항을 검토합니다. 절감형 플랜 권장 사항을 이해하고 월간 지출액과 월간 절감액을 추정해야 합니다. 계정 간 절감액을 극대화하기 위해 RI 또는 절감형 플랜 할인 공유를 활성화한 AWS 조직 내 모든 구성원 계정의 사용량을 고려하여 측정되는 관리 계정 수준의 권장 사항을 검토합니다. 필요한 할인 및 위험과 함께 올바른 권장 사항을 구현하려면 Well-Architected 실습을 따르면 됩니다.

리소스

관련 문서:

- AWS 요금은 어떻게 적용되나요
- 인스턴스 구입 옵션
- Saving Plan Overview

- Saving Plan recommendations
- Accessing Reserved Instance recommendations
- Understanding your Saving Plans recommendation
- How Savings Plans apply to your AWS usage
- 통합 결제에서 절감형 플랜
- Turning on shared reserved instances and Savings Plans discounts

관련 비디오:

- Save up to 90% and run production workloads on Spot

관련 예제:

- 절감형 플랜을 구매하기 전에 고려해야 할 사항은 무엇인가요?
- How can I use rolling Savings Plans to reduce commitment risk?
- 스팟 인스턴스를 사용하는 시기

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.

모범 사례

- COST08-BP01 데이터 전송 모델링 수행
- COST08-BP02 데이터 전송 비용을 최적화할 구성 요소 선택
- COST08-BP03 데이터 전송 비용을 줄이기 위한 서비스 구현

COST08-BP01 데이터 전송 모델링 수행

조직 요구 사항을 수집하고 워크로드 및 각 워크로드 구성 요소의 데이터 전송 모델링을 수행합니다. 그러면 현재 데이터 전송 요구 사항을 충족할 수 있는 최저 비용을 파악할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

클라우드에서 솔루션을 설계할 때 온프레미스 데이터 센터를 사용하여 아키텍처를 설계하는 습관이나 지식 부족으로 인해 데이터 전송 비용을 무시하는 경우가 많습니다. AWS의 데이터 전송 요금은 소스, 대상, 트래픽 볼륨에 따라 결정됩니다. 설계 단계에서 이러한 수수료를 고려하면 비용을 절감할 수 있습니다. 총 소유 비용(TCO)을 정확하게 추정하려면 워크로드에서 데이터 전송이 발생하는 위치, 전송 비용, 관련 이점을 이해하는 것이 매우 중요합니다. 그러면 정보를 바탕으로 결정을 내리고 아키텍처 의사 결정을 수정하거나 수락할 수 있습니다. 예를 들어 가용 영역 간에 데이터를 복제하는 다중 가용 영역 구성이 있는 경우가 있습니다.

워크로드에서 데이터를 전송하는 서비스 구성 요소를 모델링하고, 필요한 신뢰성과 복원력을 달성하기 위해 수용 가능한 비용인지(두 가용 영역의 컴퓨팅 및 스토리지 비용을 지불하는 것과 유사함) 판단합니다. 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.

데이터 전송을 모델링하는 동안 수집되는 데이터의 양과 데이터의 출처를 생각해 보세요. 또한 처리되는 데이터의 양과 필요한 스토리지 또는 컴퓨팅 용량을 고려하세요. 모델링하는 동안 워크로드 아키텍처에 대한 네트워킹 모범 사례를 따라 잠재적 데이터 전송 비용을 최적화합니다.

AWS Pricing Calculator를 통해 특정 AWS 서비스의 예상 비용과 예상 데이터 전송을 확인할 수 있습니다. 테스트 목적 또는 사전 프로덕션 환경에서 워크로드가 이미 실행 중인 경우 AWS Cost Explorer 또는 AWS Cost and Usage Report(CUR)를 사용하여 데이터 전송 비용을 파악하고 모델링합니다. 개념 증명(PoC)을 구성하거나 워크로드를 테스트하고 사실적으로 시뮬레이션된 로드로 테스트를 실행합니다. 다양한 워크로드 수요에서 비용을 모델링할 수 있습니다.

구현 단계

- 요구 사항 식별: 소스와 대상 간에 계획된 데이터 전송의 주요 목표와 비즈니스 요구 사항이 무엇인가요? 최종적으로 기대되는 비즈니스 결과는 무엇인가요? 비즈니스 요구 사항을 수집하고 예상 결과를 정의합니다.
- 소스 및 대상 식별: 데이터 전송의 데이터 소스와 대상이 무엇인가요? 예를 들면 AWS 리전 내부 전송, AWS 서비스로 전송, 인터넷으로 전송 등이 이에 해당될 수 있습니다.
  - AWS 리전 내부 데이터 전송
  - AWS 리전 간 데이터 전송
  - 인터넷으로 데이터 전송
- 데이터 분류 식별: 이 데이터 전송에 대한 데이터 분류가 어떻게 되나요? 어떤 종류의 데이터인가요? 데이터의 크기는 얼마나 되나요? 데이터를 얼마나 자주 전송해야 하나요? 데이터가 민감한가요?

- 사용할 AWS 서비스 또는 도구 식별: 이 데이터 전송에는 어떤 AWS 서비스가 사용되나요? 이미 프로비저닝된 서비스를 다른 워크로드에 사용할 수 있나요?
- 데이터 전송 비용 계산: 이전에 생성한 데이터 전송 모델링의 AWS 요금을 사용하여 워크로드에 대한 데이터 전송 비용을 계산합니다. 워크로드 사용량의 증가와 감소 모두에 대해 여러 사용 수준의 데이터 전송 비용을 계산합니다. 워크로드 아키텍처에 대한 여러 옵션이 있는 경우 비교를 위해 각 옵션의 비용을 계산합니다.
- 성과에 비용 연결: 발생한 각 데이터 전송 비용에 대해 워크로드에 가져다주는 성과를 지정합니다. 구성 요소 간 전송인 경우 분리를 위한 것일 수 있으며, 가용 영역 간 전송인 경우 이중화를 위한 것일 수 있습니다.
- 데이터 전송 모델링 생성: 모든 정보를 수집한 후 여러 사용 사례와 다양한 워크로드에 대한 개념적 기본 데이터 전송 모델링을 생성합니다.

리소스

관련 문서:

- AWS caching solutions
- AWS 요금
- Amazon EC2 요금
- Amazon VPC 요금
- Understanding data transfer charges

관련 비디오:

- Monitoring and Optimizing Your Data Transfer Costs
- S3 Transfer Acceleration

관련 예제:

- Overview of Data Transfer Costs for Common Architectures
- AWS Prescriptive Guidance for Networking

COST08-BP02 데이터 전송 비용을 최적화할 구성 요소 선택

모든 구성 요소를 선택해야 하며, 데이터 전송 비용을 줄이도록 아키텍처를 설계해야 합니다. 이때 광역 네트워크(WAN) 최적화, 다중 가용 영역 구성 등의 구성 요소를 사용할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

데이터 전송을 위한 아키텍팅은 데이터 전송 비용을 최소화합니다. 이 과정에서는 콘텐츠 전송 네트워크를 사용해 사용자와 더 가까운 위치에 데이터를 배치하거나, 온프레미스에서 AWS로의 전용 네트워크 링크를 사용합니다. 또한 WAN 최적화 및 애플리케이션 최적화를 사용하여 구성 요소 간에 전송되는 데이터의 양을 줄일 수도 있습니다.

데이터를 AWS 클라우드에 전송하거나 해당 클라우드 내부에서 전송할 때 데이터 전송을 최적화하는 데 적합한 AWS 서비스를 선택하려면 다양한 사용 사례, 데이터의 특성 및 사용 가능한 네트워크 리소스를 바탕으로 대상을 파악하는 것이 중요합니다. AWS는 여러 데이터 마이그레이션 요구 사항에 맞춤화된 다양한 데이터 전송 서비스를 제공합니다. 조직 내 비즈니스 요구 사항에 따라 적절한 데이터 스토리지 및 데이터 전송 옵션을 선택합니다.

워크로드 아키텍처를 계획하거나 검토할 때는 다음 사항을 고려하세요.

- AWS 내에서 VPC 엔드포인트 사용: VPC 엔드포인트는 VPC와 지원되는 AWS 서비스 간의 프라이빗 연결을 허용합니다. 이렇게 하면 데이터 전송 비용이 발생할 수 있는 퍼블릭 인터넷을 사용하지 않아도 됩니다.
- NAT 게이트웨이 사용: 프라이빗 서브넷의 인스턴스가 인터넷이나 VPC 외부의 서비스에 연결할 수 있도록 NAT 게이트웨이를 사용합니다. 가장 많은 트래픽을 전송하는 NAT 게이트웨이 뒤의 리소스가 NAT 게이트웨이와 동일한 가용 영역에 있는지 확인합니다. 그렇지 않은 경우 리소스와 동일한 가용 영역에 새 NAT 게이트웨이를 생성하여 교차 AZ 데이터 전송 요금을 줄이세요.
- AWS Direct Connect 사용: AWS Direct Connect에서는 퍼블릭 인터넷을 우회하고 온프레미스 네트워크와 AWS 간에 직접적인 프라이빗 연결을 설정합니다. 인터넷을 통해 대량의 데이터를 전송하는 것보다 더 비용 효율적이고 일관적인 방법일 수 있습니다.
- 리전 경계를 넘어선 데이터 전송 방지: AWS 리전 리전 간에(한 리전에서 다른 리전으로) 데이터를 전송하면 일반적으로 요금이 부과됩니다. 다중 리전 경로를 이용하려면 매우 신중하게 결정해야 합니다. 자세한 내용은 다중 리전 시나리오를 참조하세요.
- 데이터 전송 모니터링: Amazon CloudWatch 및 VPC 흐름 로그를 사용하여 데이터 전송 및 네트워크 사용에 대한 세부 정보를 캡처합니다. VPC에서 캡처한 네트워크 트래픽 정보(예: 네트워크 인터페이스에서 주고받는 IP 주소 또는 범위)를 분석합니다.

- 네트워크 사용량 분석: AWS Cost Explorer, CUDOS 대시보드, CloudWatch와 같은 측정 및 보고 도구를 사용하여 워크로드의 데이터 전송 비용을 파악합니다.

구현 단계

- 데이터 전송을 위한 구성 요소 선택: COST08-BP01 데이터 전송 모델링 수행에서 설명한 데이터 전송 모델링을 사용하여 데이터 전송 비용이 가장 큰 영역이나 워크로드 사용량이 변경되는 경우 데이터 전송 비용이 발생하는 영역을 집중적으로 살펴봅니다. 데이터 전송에 대한 필요를 줄이거나, 없애거나, 비용을 낮추는 대체 아키텍처나 추가 구성 요소를 찾습니다.

리소스

관련 모범 사례:

- COST08-BP01 데이터 전송 모델링 수행
- COST08-BP03 데이터 전송 비용을 줄이기 위한 서비스 구현

관련 문서:

- Cloud Data Migration
- AWS caching solutions
- Deliver content faster with Amazon CloudFront

관련 예제:

- Overview of Data Transfer Costs for Common Architectures
- AWS Network Optimization Tips
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format

COST08-BP03 데이터 전송 비용을 줄이기 위한 서비스 구현

데이터 전송을 줄이기 위한 서비스를 구현합니다. 예를 들어 엣지 로케이션 또는 콘텐츠 전송 네트워크(CDN)를 사용하여 최종 사용자에게 콘텐츠를 전송하고, 애플리케이션 서버 또는 데이터베이스 앞에 캐싱 계층을 구축하며, 클라우드 연결에 VPN 대신 전용 네트워크 연결을 사용할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 지침

네트워크 데이터 전송 사용을 최적화하는 데 도움이 되는 다양한 AWS 서비스가 있습니다. 이러한 서비스는 워크로드 구성 요소, 유형 및 클라우드 아키텍처에 따라 클라우드에서 트래픽을 압축, 캐싱, 공유 및 분배하는 데 도움이 될 수 있습니다.

- Amazon CloudFront는 지연 시간이 짧고 전송 속도가 빠른 데이터 전송용 글로벌 콘텐츠 전송 네트워크입니다. 전 세계의 엣지 로케이션에서 데이터를 캐시하여 리소스 부하를 줄여줍니다. CloudFront를 사용하면 전 세계의 많은 사용자에게 콘텐츠를 전송하는 과정의 관리 작업을 줄이고 지연 시간을 최소화할 수 있습니다. Security Savings Bundle을 사용하면 시간이 지남에 따라 사용량을 늘리려는 경우 CloudFront 사용량을 최대 30%까지 절약할 수 있습니다.
- AWS Direct Connect를 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 VPN을 사용하면 프라이빗 네트워크와 AWS 글로벌 네트워크 간에 안전한 프라이빗 연결을 설정할 수 있습니다. 탄력적인 완전관리형 서비스로서 간소화된 연결을 제공하므로 소규모 사무실 또는 비즈니스 파트너에 적합합니다.
- VPC 엔드포인트를 사용하면 프라이빗 네트워킹을 통해 여러 AWS 서비스를 연결할 수 있으며, 퍼블릭 데이터 전송 및 NAT 게이트웨이 비용을 줄일 수 있습니다. 게이트웨이 VPC 엔드포인트는 시간당 비용이 없으며 Amazon S3 및 Amazon DynamoDB를 지원합니다. 인터페이스 VPC 엔드포인트는 AWS PrivateLink를 통해 제공되며 시간당 요금과 GB당 사용 비용이 있습니다.
- NAT 게이트웨이는 독립형 NAT 인스턴스와 달리 규모 조정 및 관리 기능을 기본 제공하여 비용을 절감합니다. 트래픽이 많은 인스턴스와 동일한 가용 영역에 NAT 게이트웨이를 배치하고, Amazon DynamoDB 또는 Amazon S3 액세스가 필요한 인스턴스에는 VPC 엔드포인트를 사용하여 데이터 전송 및 처리 비용을 줄이는 것을 고려하세요.
- 엣지에서 데이터를 수집하고 처리하기 위한 컴퓨팅 리소스가 있는 AWS Snow Family 디바이스를 사용합니다. AWS Snow Family 디바이스(Snowball Edge, Snowball Edge 및 Snowmobile)를 사용하면 페타바이트 규모의 데이터를 AWS 클라우드로 비용 효과적으로, 오프라인으로 이동할 수 있습니다.

구현 단계

- 서비스 구현: 데이터 전송 모델링을 사용하고 VPC 흐름 로그를 검토하여 서비스, 워크로드 유형에 따라 적절한 AWS 네트워크 서비스를 선택합니다. 비용이 가장 크고 볼륨 흐름이 가장 높은 영역을 확인합니다. AWS 서비스를 검토하고 전송을 줄이거나 제거하는 서비스(특히 네트워킹 및 콘텐츠 전

송)가 있는지 평가합니다. 또한 데이터 또는 대량의 데이터에 대한 액세스가 반복되는 캐싱 서비스를 찾습니다.

리소스

관련 문서:

- AWS Direct Connect
- AWS 제품 둘러보기
- AWS caching solutions
- Amazon CloudFront
- AWS Snow Family
- Amazon CloudFront Security Savings Bundle

관련 비디오:

- Monitoring and Optimizing Your Data Transfer Costs
- AWS Cost Optimization Series: CloudFront
- VPC의 NAT 게이트웨이에 대한 데이터 전송 요금을 줄이려면 어떻게 해야 하나요?

관련 예제:

- How-to chargeback shared services: An AWS Transit Gateway example
- Understand AWS data transfer details in depth from cost and usage report using Athena query and QuickSight
- Overview of Data Transfer Costs for Common Architectures
- Using AWS Cost Explorer to analyze data transfer costs
- Cost-Optimizing your AWS architectures by utilizing Amazon CloudFront features
- VPC의 NAT 게이트웨이에 대한 데이터 전송 요금을 줄이려면 어떻게 해야 하나요?

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 제한, 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 수요의 고정 또는 가변 여부에 따라 환경의 규모를 조정하더라도 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 모범 사례에서는 아래와 같은 방식을 사용하는 방법을 설명합니다.

모범 사례

- COST09-BP01 워크로드 수요 분석 수행
- COST09-BP02 수요 관리를 위한 버퍼 또는 제한 구현
- COST09-BP03 동적으로 리소스 공급

COST09-BP01 워크로드 수요 분석 수행

시간별 워크로드 수요를 분석합니다. 분석에서 시기별 추세를 파악하고 전체 워크로드 수명 주기 동안의 작동 상태를 정확하게 반영하는지 확인합니다. 분석 작업은 소요되는 시간 대비 워크로드 비용 등의 제공될 수 있는 이점을 반영해야 합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 지침

클라우드 컴퓨팅에 대한 워크로드 수요를 분석하려면 클라우드 환경에서 시작되는 컴퓨팅 작업의 패턴과 특성을 이해해야 합니다. 이 분석을 통해 사용자는 리소스 할당을 최적화하고 비용을 관리하며 성능이 요구 수준을 충족하는지 확인할 수 있습니다.

워크로드의 요구 사항을 파악합니다. 조직의 요구 사항에는 요청에 대한 워크로드 응답 시간이 표시되어야 합니다. 응답 시간을 사용하면 수요가 관리되는지 여부 또는 리소스 공급이 수요에 따라 변경되어야 하는지 여부를 확인할 수 있습니다.

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

분석 작업에는 규모 조정 구현의 잠재적 이점이 반영되어야 합니다. 구성 요소의 예상 총 비용을 찾아보고 워크로드 수명에 걸쳐 사용량 및 비용이 증가하거나 감소하는지 살펴봅니다.

다음은 클라우드 컴퓨팅에 대한 워크로드 수요 분석을 수행할 때 고려해야 할 몇 가지 주요 측면입니다.

1. 리소스 사용률 및 성과 지표: 시간이 지남에 따라 AWS 리소스가 어떻게 사용되고 있는지 분석합니다. 사용량이 최대일 때와 사용량이 적을 때의 패턴을 확인하여 리소스 할당 및 규모 조정 전략을 최적화합니다. 응답 시간, 지연 시간, 처리량, 오류율과 같은 성과 지표를 모니터링합니다. 이러한 지표는 클라우드 인프라의 전반적인 상태와 효율성을 평가하는 데 도움이 됩니다.
2. 사용자 및 애플리케이션 규모 조정 동작: 사용자 행동과 이것이 워크로드 수요에 미치는 영향을 파악합니다. 사용자 트래픽 패턴을 조사하면 콘텐츠 전송 및 애플리케이션 응답성을 향상시키는 데 도움이 됩니다. 증가하는 수요에 맞추어 워크로드 규모가 어떻게 조정되는지 분석합니다. 로드 변동을 처리할 수 있도록 Auto Scaling 파라미터가 정확하고 효과적으로 구성되었는지 확인합니다.
3. 워크로드 유형: 일괄 처리, 실시간 데이터 처리, 웹 애플리케이션, 데이터베이스 또는 기계 학습과 같이 클라우드에서 실행되는 다양한 유형의 워크로드를 파악합니다. 워크로드 유형마다 리소스 요구 사항 및 성능 프로필이 다를 수 있습니다.
4. 서비스 수준에 관한 계약(SLA): 실제 성능을 SLA와 비교하여 규정 준수를 보장하고 개선이 필요한 영역을 파악합니다.

Amazon CloudWatch를 사용하여 지표를 수집 및 추적하고, 로그 파일을 수집 및 모니터링하며, 경보를 설정하고, AWS 리소스 변경에 자동으로 대응할 수 있습니다. 또한 Amazon CloudWatch를 사용하여 시스템 전반의 리소스 사용률, 애플리케이션 성능, 운영 상태를 파악할 수 있습니다.

AWS Trusted Advisor를 사용하면 모범 사례에 따라 리소스를 프로비저닝하여 시스템 성능 및 신뢰성을 개선하고, 보안을 강화하며, 비용을 절감할 기회를 모색할 수 있습니다. 비프로덕션 인스턴스를 끄고 수요 증가 또는 감소에 맞춰 Amazon CloudWatch 및 Auto Scaling을 사용할 수도 있습니다.

마지막으로, AWS Cost Explorer 또는 QuickSight를 AWS Cost and Usage Report(CUR) 파일 또는 애플리케이션 로그와 함께 사용하여 워크로드 수요에 대한 고급 분석을 수행할 수 있습니다.

전반적으로, 포괄적인 워크로드 수요 분석을 통해 조직은 리소스 프로비저닝, 규모 조정 및 최적화에 대해 합리적인 결정을 내려서 성능, 비용 효율성 및 사용자 만족도를 높일 수 있습니다.

구현 단계

- 기존 워크로드 데이터 분석: 기존 워크로드, 이전 버전의 워크로드 또는 예측된 사용 패턴의 데이터를 분석합니다. Amazon CloudWatch, 로그 파일 및 모니터링 데이터를 사용하여 워크로드가 어떻게 사용되었는지에 대한 인사이트를 얻을 수 있습니다. 워크로드의 전체 주기를 분석하여 월말 또는 연말 이벤트와 같은 주기적 변경 사항에 대한 데이터를 수집합니다. 분석에 반영되는 작업량은 워크로드 특성을 반영해야 합니다. 수요 변화가 가장 많은 고가치 워크로드에 가장 많은 작업량을 배치해야 합니다. 수요 변화가 가장 적은 저가치 워크로드에 가장 적은 작업량을 배치해야 합니다.
- 외부 영향 예측: 워크로드의 수요에 영향을 주거나 변화를 줄 수 있는 조직 전체의 팀원과 만납니다. 일반적인 팀은 영업, 마케팅 또는 비즈니스 개발입니다. 이들 팀과 협력하여 작업 주기를 확인하고 워크로드 수요를 바꾸는 이벤트가 있는지 확인합니다. 이 데이터로 워크로드 수요를 예측합니다.

리소스

관련 문서:

- Amazon CloudWatch
- AWS Trusted Advisor
- AWS X-Ray
- AWS Auto Scaling
- AWS Instance Scheduler
- Getting started with Amazon SQS
- AWS Cost Explorer
- QuickSight

관련 예제:

- Monitor, Track and Analyze for cost optimization
- Searching and analyzing logs in CloudWatch

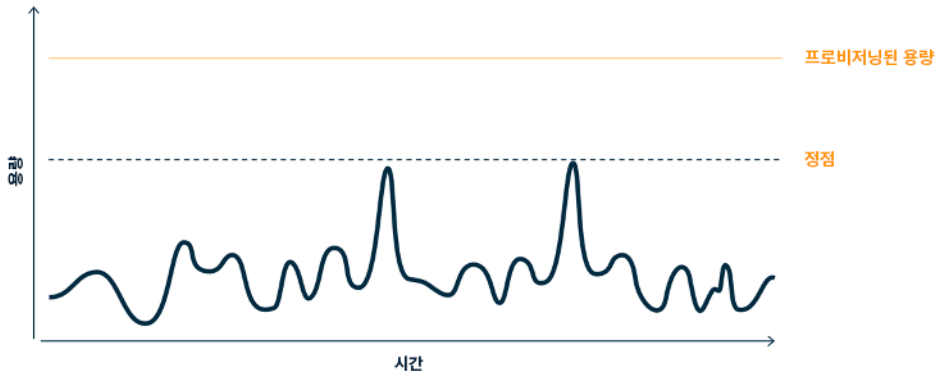
COST09-BP02 수요 관리를 위한 버퍼 또는 제한 구현

버퍼링 및 제한은 워크로드의 수요를 수정하여 평준화합니다. 클라이언트가 재시도를 수행할 때 제한을 구현합니다. 요청을 저장하고 나중에 처리하도록 버퍼링을 구현합니다. 클라이언트가 필요한 시간에 응답을 수신하도록 제한 및 버퍼가 설계되었는지 확인합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

클라우드 컴퓨팅에서 수요를 관리하고 워크로드에 필요한 프로비저닝 용량을 줄이려면 버퍼 또는 제한을 구현하는 것이 중요합니다. 최적의 성능을 위해서는 피크, 요청 변경 속도, 필요한 응답 시간을 포함한 총 수요를 측정하는 것이 핵심입니다. 클라이언트가 요청을 재전송할 수 있게 되면 제한을 적용하는 것이 실용적입니다. 반대로 재시도 기능이 없는 클라이언트의 경우 이상적인 접근 방식은 버퍼 솔루션을 구현하는 것입니다. 이러한 버퍼는 요청 유입을 간소화하고 다양한 작업 속도로 애플리케이션 간의 상호 작용을 최적화합니다.



높은 프로비저닝 용량을 필요로 하는 두 개의 피크가 있는 수요 곡선

위 그림에 표시된 수요 곡선을 바탕으로 워크로드를 가정합니다. 이 워크로드에는 2개의 피크가 있으며, 이러한 피크를 처리하기 위해 주황색 선으로 표시된 리소스 용량이 프로비저닝됩니다. 이 워크로드에 사용되는 리소스와 에너지는 수요 곡선 아래의 영역이 아니라 프로비저닝된 용량 선 아래의 영역으로 표시됩니다. 이 2개의 피크를 처리하려면 프로비저닝된 용량이 필요하기 때문입니다. 워크로드 수요 곡선을 완화하면 워크로드에 프로비저닝된 용량을 줄이고 환경에 미치는 영향도 줄일 수 있습니다. 피크 속도를 낮추려면 제한 또는 버퍼링 솔루션을 구현하는 것이 좋습니다.

이해를 돕기 위해 제한과 버퍼링에 대해 살펴보겠습니다.

제한: 수요의 소스에 재시도 기능이 있는 경우 제한을 구현할 수 있습니다. 제한은 현재 요청을 처리할 수 없는 경우 나중에 다시 시도해야 함을 소스에 알려줍니다. 소스는 일정 기간 기다린 다음 요청을 재시도합니다. 제한을 구현하면 워크로드의 최대 리소스 양과 비용을 제한할 수 있는 장점이 있습니다. AWS에서는 Amazon API Gateway를 사용하여 제한을 구현할 수 있습니다.

버퍼 기반: 버퍼 기반 접근 방식은 생산자(대기열에 메시지를 보내는 구성 요소), 소비자(대기열에서 메시지를 받는 구성 요소) 및 대기열(메시지를 보관함)을 사용하여 메시지를 저장합니다. 메시지는 소비

자가 읽은 후 처리되므로 소비자의 비즈니스 요구 사항을 충족하는 속도로 메시지를 실행할 수 있습니다. 버퍼 중심 방법을 사용하면 생산자의 메시지가 대기열이나 스트림에 보관되어 소비자가 작업 요구 사항에 맞는 속도로 액세스할 수 있습니다.

AWS에서는 여러 서비스 중에서 버퍼링 방식을 구현하는 데 적합한 서비스를 선택할 수 있습니다. Amazon Simple Queue Service(Amazon SQS)는 단일 소비자가 개별 메시지를 읽을 수 있는 대기열을 제공하는 관리형 서비스입니다. Amazon Kinesis에서는 여러 소비자가 같은 메시지를 읽을 수 있는 스트림을 제공합니다.

버퍼링 및 제한을 통해 워크로드에 대한 수요를 수정하여 피크를 원활하게 처리할 수 있습니다. 클라이언트가 작업을 재시도할 때는 제한을 사용하고 요청을 보류했다가 나중에 처리하려면 버퍼링을 사용합니다. 버퍼 기반 접근 방식을 사용할 때는 필요한 시간에 요청을 처리하도록 워크로드를 설계해야 하며 작업에 대한 중복 요청을 처리할 수 있는지 확인해야 합니다. 전체 수요, 변경률 및 필수 응답 시간을 분석하여 필요한 제한 또는 버퍼의 크기를 적절하게 조정합니다.

구현 단계

- 클라이언트 요구 사항 분석: 클라이언트 요청을 분석하여 재시도를 수행할 수 있는지 확인합니다. 재시도를 수행할 수 없는 클라이언트의 경우 버퍼를 구현해야 합니다. 전체 수요, 변경률 및 필요한 응답 시간을 분석하여 필요한 제한 또는 버퍼의 크기를 결정합니다.
- 버퍼 또는 제한 구현: 워크로드에 버퍼 또는 제한을 구현합니다. Amazon Simple Queue Service(Amazon SQS)와 같은 대기열은 워크로드 구성 요소에 버퍼를 제공할 수 있습니다. Amazon API Gateway는 워크로드 구성 요소에 대한 제한을 제공할 수 있습니다.

리소스

관련 모범 사례:

- SUS02-BP06 버퍼링 또는 제한 개선으로 수요 곡선 완화
- REL05-BP02 요청 제한

관련 문서:

- AWS Auto Scaling
- AWS Instance Scheduler
- Amazon API Gateway
- Amazon Simple Queue Service

- Getting started with Amazon SQS
- Amazon Kinesis

관련 비디오:

- Choosing the Right Messaging Service for Your Distributed App

관련 예제:

- Managing and monitoring API throttling in your workloads
- Throttling a tiered, multi-tenant REST API at scale using API Gateway
- Enabling Tiering and Throttling in a Multi-Tenant Amazon EKS SaaS Solution Using Amazon API Gateway
- Application integration Using Queues and Messages

COST09-BP03 동적으로 리소스 공급

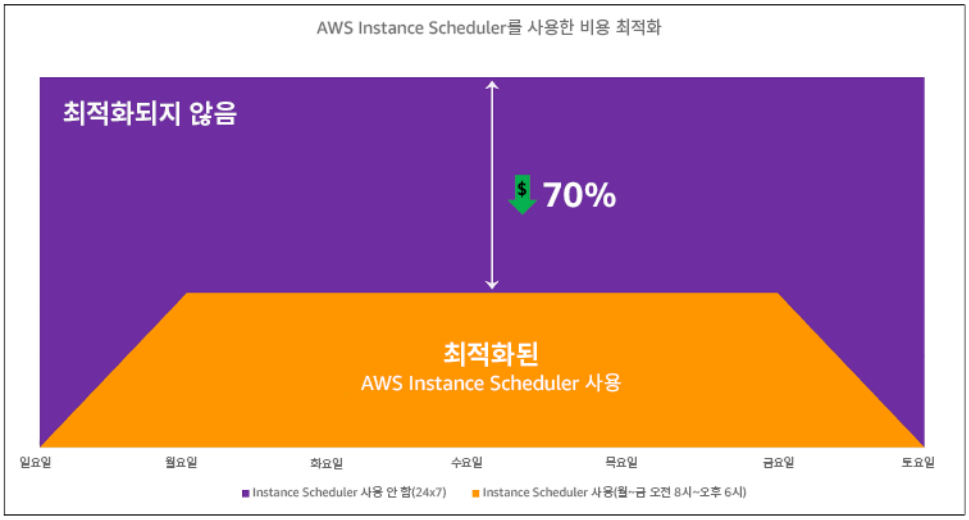
리소스가 계획된 방식으로 프로비저닝됩니다. 자동 크기 조정과 같은 수요 기반이거나, 수요를 예측할 수 있고 리소스가 시간을 기준으로 제공되는 시간 기반일 수 있습니다. 이러한 방법을 사용하면 과다 프로비저닝 또는 과소 프로비저닝을 최소화할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 가이드

AWS 고객이 애플리케이션에 사용할 수 있는 리소스를 늘리고 수요를 충족하는 리소스를 공급할 수 있는 몇 가지 방법이 있습니다. 이러한 옵션 중 하나는 Amazon Elastic Compute Cloud(Amazon EC2) 및 Amazon Relational Database Service(RDS) 인스턴스의 시작 및 중지를 자동화하는 AWS Instance Scheduler를 사용하는 것입니다. 다른 옵션은 AWS Auto Scaling을 사용하는 것입니다. 이 옵션을 사용하면 애플리케이션 또는 서비스의 요구에 따라 컴퓨팅 리소스를 자동으로 조정할 수 있습니다. 수요에 따라 리소스를 공급하면 사용한 리소스에 대해서만 비용을 지불하고 필요할 때 리소스를 해제하여 비용을 절감하며 필요하지 않을 때는 리소스를 종료할 수 있습니다.

AWS Instance Scheduler를 사용하면 정의된 시간에 Amazon EC2 및 Amazon RDS 인스턴스의 중지와 시작을 구성할 수 있습니다. 그러면 예를 들어 저녁 6시 이후에는 필요 없는 Amazon EC2 인스턴스에 매일 오전 8시에 액세스하는 등의 일정한 시간 패턴을 통해 동일한 리소스의 수요를 충족할 수 있습니다. 이 솔루션은 사용하지 않는 리소스는 중지하고 필요할 때 시작하여 운영 비용을 줄여줍니다.



AWS Instance Scheduler를 통한 비용 최적화.

또한 AWS Systems Manager 빠른 설정을 사용하여 간단한 사용자 인터페이스(UI)로 계정 및 리전 전체에서 Amazon EC2 인스턴스의 일정을 쉽게 구성할 수 있습니다. AWS Instance Scheduler로 Amazon EC2 또는 Amazon RDS 인스턴스를 예약하고 기존 인스턴스를 중지 및 시작할 수 있습니다. 그러나 Auto Scaling 그룹(ASG)의 일부이거나 Amazon Redshift 또는 Amazon OpenSearch Service 등과 같은 서비스를 관리하는 인스턴스는 중지 및 시작할 수 없습니다. Auto Scaling 그룹에는 그룹 내 인스턴스에 대한 고유한 일정이 있으며 이러한 인스턴스가 생성됩니다.

AWS Auto Scaling은 용량을 조정하여 최대한 저렴한 비용으로 변화하는 수요를 충족하기 위한 안정적이고 예측 가능한 성능을 유지하는 데 도움이 됩니다. Amazon EC2 인스턴스 및 스팟 플릿, Amazon ECS, Amazon DynamoDB 및 Amazon Aurora와 통합되어 애플리케이션 용량을 조정하는 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 그룹을 조정하는 데 사용할 수 있는 다양한 규모 조정 옵션이 있습니다.

- 항상 현재 인스턴스 수준 유지 관리
- 수동 조정
- 일정에 근거하여 조정
- 온디맨드 기반 조정
- 예측 조정 사용

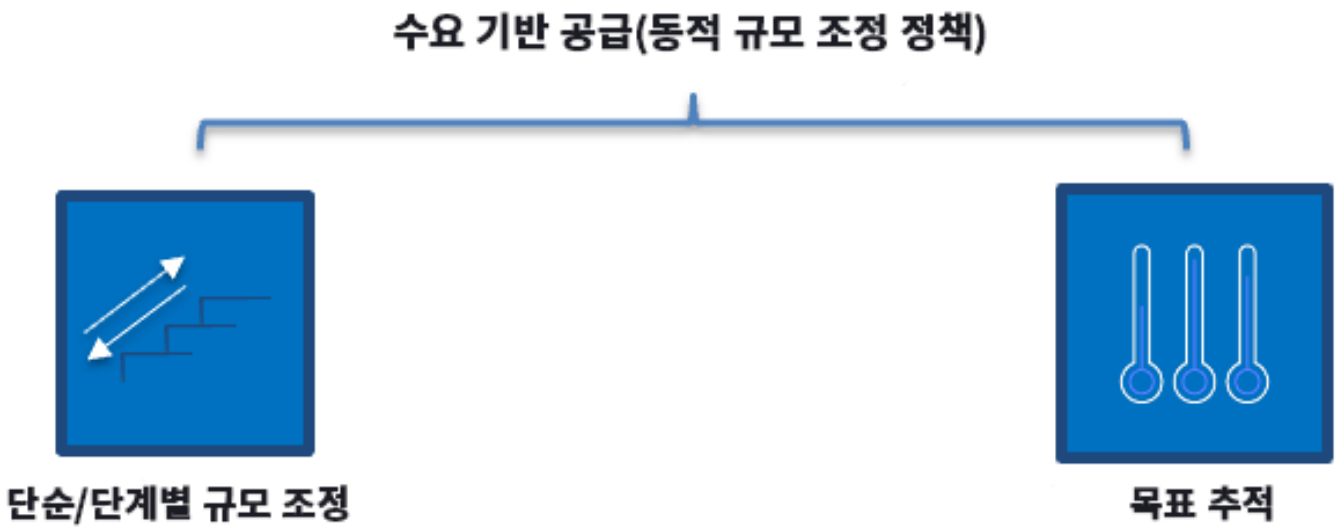
Auto Scaling 정책은 서로 다르며 동적 및 예약 규모 조정 정책으로 분류될 수 있습니다. 동적 정책은 수동 또는 동적 규모 조정으로, 예약 또는 예측 규모 조정입니다. 동적, 예약 및 예측 규모 조정에 규모 조정 정책을 사용할 수 있습니다. 또한 Amazon CloudWatch의 지표와 경보를 사용하여 워크로드에 대한 규모 조정 이벤트를 트리거할 수 있습니다. 최신 기능과 개선 사항에 액세스하려면 시작 템플릿을 사용하는 것이 좋습니다. 시작 구성을 사용할 때 일부 Auto Scaling 기능을 사용할 수 없습니다. 예를 들어, 스팟 및 온디맨드 인스턴스를 모두 시작하거나 여러 인스턴스 유형을 지정하는 Auto Scaling 그룹은 생성할 수 없습니다. 이러한 기능을 구성하려면 시작 템플릿을 사용해야 합니다. 시작 템플릿을 사용할 때는 각 템플릿의 버전을 지정하는 것이 좋습니다. 시작 템플릿 버전 관리를 사용하면 전체 파라미터 세트의 하위 세트를 생성할 수 있습니다. 그런 다음, 해당 세트를 재사용하여 동일한 시작 템플릿의 다른 버전을 만들 수 있습니다.

AWS Auto Scaling을 사용하거나 AWS API 또는 SDK를 사용하여 코드에서 규모 조정을 통합할 수 있습니다. 이렇게 하면 환경을 수동으로 변경하는 데 따른 운영 비용이 제거되므로 전반적인 워크로드 비용이 절감되며 변경을 훨씬 더 빠르게 수행할 수 있습니다. 이는 또한 언제든지 수요에 맞춰 워크로드 리소스를 조정합니다. 이 모범 사례를 따르고 조직에 리소스를 동적으로 공급하려면 AWS 클라우드의 수평적 스케일링 및 수직적 스케일링과 Amazon EC2 인스턴스에서 실행 중인 애플리케이션의 특성을 이해해야 합니다. 이 모범 사례를 따르기 위해 클라우드 재무 관리 팀이 기술 팀과 협력하는 것이 좋습니다.

Elastic Load Balancing은 여러 리소스에 걸쳐 수요를 분산하여 규모를 조정하는 데 도움이 됩니다. ASG 및 Elastic Load Balancing을 사용하면 Auto Scaling 그룹 내에서 어떤 인스턴스에도 부하가 걸리지 않도록 트래픽을 최적으로 라우팅하여 들어오는 요청을 관리할 수 있습니다. 요청은 용량 또는 사용률을 고려하지 않고 대상 그룹의 모든 대상에 라운드 로빈 방식으로 분산됩니다.

일반적인 지표로는 CPU 사용률, 네트워크 처리량 및 Elastic Load Balancing에서 관찰된 요청 및 응답 지연 시간과 같은 표준 Amazon EC2 지표가 있습니다. 가능한 경우 고객 경험을 나타내는 지표를 사용해야 합니다. 일반적으로는 워크로드 내 애플리케이션 코드에서 생성될 수 있는 사용자 지정 지표입니다. 이 문서에서는 수요를 동적으로 충족하는 방법을 자세히 설명하기 위해 Auto Scaling을 수요 기반 공급 모델과 시간 기반 공급 모델의 두 범주로 분류하고 각 모델을 심층적으로 살펴보겠습니다.

수요 기반 공급: 클라우드의 탄력성을 활용하여 거의 실시간에 가까운 수요 상태를 기반으로 변화하는 수요를 충족할 리소스를 공급합니다. 수요 기반 공급에서는 API 또는 서비스 기능을 사용하여 프로그래밍 방식을 통해 아키텍처의 클라우드 리소스 양을 변경합니다. 이렇게 하면 아키텍처 구성 요소의 규모를 조정할 수 있으며, 수요 급증 기간에는 리소스 수를 늘려 성능을 유지하고 수요 감소 기간에는 용량을 줄여 비용을 절감할 수 있습니다.

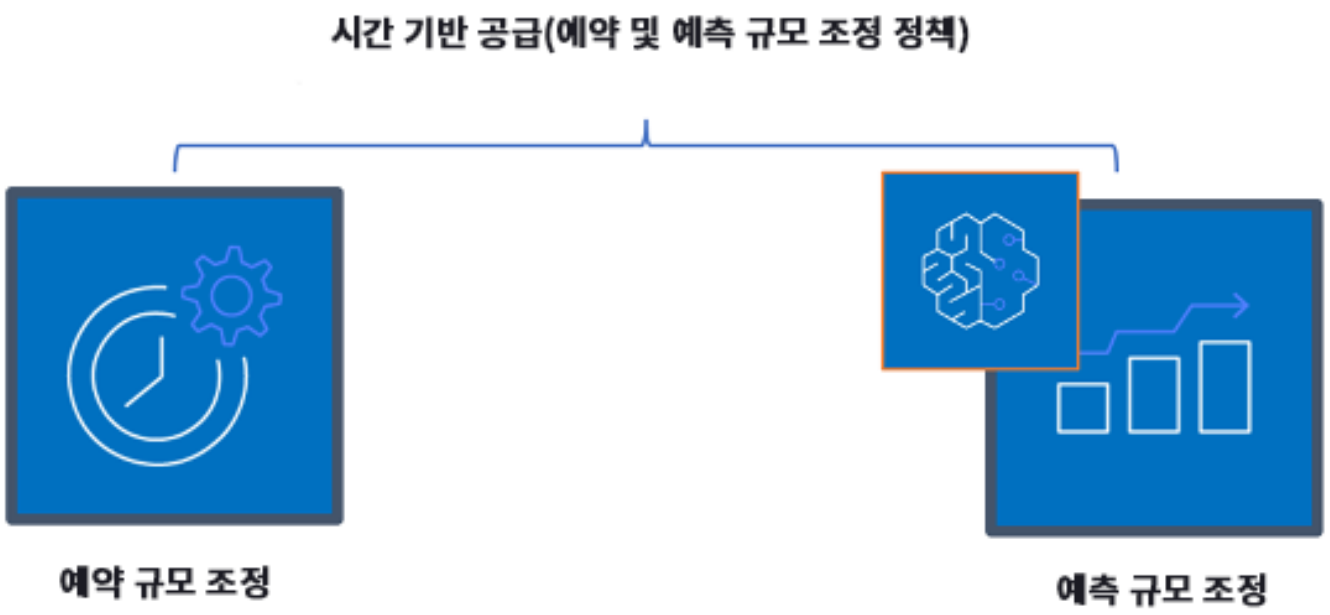


수요 기반 동적 규모 조정 정책

- 단순/단계별 규모 조정: 고객이 수동으로 정의한 단계에 따라 지표를 모니터링하고 인스턴스를 추가 또는 제거합니다.
- 목표 추적: 지표를 사용자가 정의한 목표로 유지하기 위해 인스턴스를 자동으로 추가 또는 제거하는 온도 조절기와 유사한 제어 메커니즘입니다.

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.

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 기간에 추가 리소스를 제공하거나 용량을 늘릴 수 있습니다.



시간 기반 규모 조정 정책

예약 또는 예측 Auto Scaling을 사용하여 시간 기반 접근 방식을 구현할 수 있습니다. 사용자 도달 또는 수요 증가 시 리소스를 사용할 수 있도록 정의된 시간(예: 업무 시간 시작 시)에 워크로드 스케일 아웃 또는 스케일 인을 예약할 수 있습니다. 예측 규모 조정은 패턴을 사용하여 스케일 아웃하는 반면에 예약된 규모 조정은 미리 정의된 시간을 사용하여 스케일 아웃합니다. 또한 Auto Scaling 그룹의 속성 기반 인스턴스 유형 선택(ABS) 전략을 사용하여 vCPU, 메모리 및 스토리지와 같은 일련의 속성으로서 인스턴스 요구 사항을 표현할 수 있습니다. 또한 신세대 인스턴스 유형이 릴리스되면 이를 자동으로 사용하고 Amazon EC2 스팟 인스턴스를 통해 더 광범위한 용량에 액세스할 수 있습니다. Amazon EC2 플릿과 Amazon EC2 Auto Scaling은 지정된 속성에 맞는 인스턴스를 선택하고 시작하여 인스턴스 유형을 수동으로 선택할 필요성이 사라집니다.

또한 AWS API 및 SDK와 AWS CloudFormation을 활용하여 필요한 경우 전체 환경을 자동으로 프로비저닝하고 폐기할 수 있습니다. 이 방식은 정의된 업무 시간이나 일정 기간에만 실행되는 개발 또는 테스트 환경에 적합합니다. API를 사용해 환경 내에서 리소스 규모를 조정할 수 있습니다(수직적 스케일링). 예를 들어 인스턴스 크기나 클래스를 변경하여 프로덕션 워크로드의 규모를 스케일 업할 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 Amazon 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.

구현 단계

- 예약 규모 조정 구성: 예측 가능한 수요 변화를 위해 시간 기반 조정은 적시에 올바른 개수의 리소스를 제공할 수 있습니다. 리소스 생성 및 구성이 수요 변화에 대응할 만큼 충분히 빠르지 않은 경우에도 유용합니다. 워크로드 분석으로 AWS Auto Scaling을 사용하여 예약된 규모 조정을 구성합니다. 시간 기반 일정을 구성하기 위해 예약된 규모 조정의 예측 규모 조정을 사용하여 예상되는 또는 예측 가능한 로드 변화에 따라 Auto Scaling 그룹 내 Amazon EC2 인스턴스 수를 미리 늘릴 수 있습니다.
- 예측 규모 조정 구성: 예측 규모 조정을 사용하여 트래픽 흐름의 일일 및 주간 패턴에 앞서 Auto Scaling 그룹의 Amazon EC2 인스턴스 수를 늘릴 수 있습니다. 시작하는 데 오래 걸리는 애플리케이션이 있고 정기적으로 트래픽이 급증하는 경우 예측 규모 조정 사용을 고려해야 합니다. 예측 규모 조정은 예측한 로드가 발생 전에 용량을 초기화함으로써 반응적인 속성의 동적 규모 조정을 단독으로 사용하는 것과 비교하여 더 빠르게 규모를 조정할 수 있도록 합니다. 예를 들어, 사용자가 업무 시간 시작과 함께 워크로드를 사용하기 시작하고 업무 시간이 지나면 사용하지 않는 경우, 예측 규모 조정은 업무 시간 전에 용량을 추가할 수 있습니다. 그러면 변화하는 트래픽에 대응하기 위한 동적 규모 조정의 지연이 사라집니다.
- 동적 자동 규모 조정 구성: 활성 워크로드 지표를 기반으로 조정을 구성하려면 Auto Scaling을 사용합니다. 분석을 사용하여 올바른 리소스 수준에서 시작하도록 Auto Scaling을 구성하고 워크로드가 필요한 시간 내에 조정되도록 합니다. 단일 Auto Scaling 그룹 내에서 온디맨드 인스턴스 및 스팟 인스턴스 플릿를 자동으로 확장할 수 있습니다. 스팟 인스턴스 사용에 대한 할인을 받을 수 있을 뿐만 아니라 예약 인스턴스 또는 Savings Plans를 사용하여 일반 온디맨드 인스턴스 요금의 할인된 요금을 받을 수 있습니다. 이러한 모든 요소를 결합하면 Amazon EC2 인스턴스의 비용 절감을 최적화할 수 있으며 애플리케이션에서 원하는 규모 및 성능을 얻을 수 있습니다.

리소스

관련 문서:

- AWS Auto Scaling
- AWS Instance Scheduler
- Auto Scaling 그룹의 크기 조정
- Getting Started with Amazon EC2 Auto Scaling

- Getting started with Amazon SQS
- Scheduled Scaling for Amazon EC2 Auto Scaling
- Predictive scaling for Amazon EC2 Auto Scaling

관련 비디오:

- Target Tracking Scaling Policies for Auto Scaling
- AWS Instance Scheduler

관련 예제:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive Scaling with Amazon EC2 Auto Scaling
- AWS CloudFormation과 함께 Instance Scheduler를 사용하여 EC2 인스턴스를 예약하려면 어떻게 해야 하나요?

시간 경과에 따른 최적화

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

AWS에서 신규 서비스와 기능이 릴리스되면 기존에 결정한 아키텍처 관련 사항을 검토하여 해당 사항이 비용 효율적으로 유지되는지 확인하는 것이 좋습니다. 요구 사항이 변경되면 더 이상 필요하지 않은 리소스, 구성 요소 및 워크로드는 과감하게 폐기하세요. 다음 모범 사례를 고려하여 시간 경과에 따른 최적화를 지원하세요.

시간이 지남에 따라 워크로드를 최적화하고 조직의 CFM 문화를 개선하는 동시에, 클라우드에서의 운영 비용을 평가하고, 시간이 많이 걸리는 클라우드 운영을 검토하며, 관련 AWS 서비스, 서드파티 제품 또는 사용자 지정 도구(예: AWS CLI 또는 AWS SDK)를 채택하여 인적 노력과 비용을 줄이도록 이를 자동화하합니다.

주제

- 검토 프로세스 정의 및 정기적인 워크로드 분석
- 운영 자동화

검토 프로세스 정의 및 정기적인 워크로드 분석

모범 사례

- COST10-BP01 워크로드 검토 프로세스 개발
- COST10-BP02 정기적으로 워크로드 검토 및 분석

COST10-BP01 워크로드 검토 프로세스 개발

워크로드 검토 기준과 프로세스를 정의하는 프로세스를 개발합니다. 검토 작업에는 잠재적 이점이 반영되어야 합니다. 예를 들어 핵심 워크로드 또는 비용이 청구 금액의 10%보다 많은 워크로드는 분기별로 또는 연 2회 검토하고, 비용이 청구 금액의 10%보다 적은 워크로드는 연 1회 검토할 수 있습니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 높음

구현 가이드

가장 비용 효율적인 워크로드를 유지하려면 워크로드를 정기적으로 검토하여 새로운 서비스, 기능 및 구성 요소를 구현할 기회가 있는지 확인해야 합니다. 전반적인 비용 절감을 실현하려면 검토 프로세스

가 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 변동성을 고려합니다. 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 고려하세요. 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 나열하세요. 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.

새 서비스, 설계 패턴, 리소스 유형 및 구성을 사용할 수 있게 되면 워크로드 비용을 최적화하기 위해 평가를 위한 프로세스를 정의합니다. 성능 원칙 검토 및 신뢰성 원칙 검토 프로세스와 마찬가지로, 최적화 및 개선 활동과 문제 해결 방법을 식별 및 검증하고 우선순위를 지정한 후 백로그에 통합합니다.

구현 단계

- 검토 빈도 정의: 워크로드 및 해당 구성 요소를 검토해야 하는 빈도를 정의합니다. 지속적인 개선 및 검토 빈도를 위해 시간과 리소스를 할당하여 워크로드의 효율성 및 최적화를 개선합니다. 이는 여러 가지 요소를 조합한 것으로 조직 내 워크로드마다 다를 수 있으며 워크로드의 구성 요소 간에 다를 수 있습니다. 일반적인 요인으로는 수익 또는 브랜드 측면에서 측정한 조직에 대한 중요성, 워크로드의 총 실행 비용(운영 및 리소스 비용 포함), 워크로드의 복잡성, 변경 실행 용이성, 소프트웨어 라이선스 계약, 변경으로 인해 징벌적 라이선스에 의한 라이선스 비용이 크게 증가하는지 여부 등이 있습니다. 기능적 또는 기술적으로 구성 요소를 정의할 수 있습니다(예: 웹 서버 및 데이터베이스, 컴퓨팅 및 스토리지 리소스). 요인을 적절히 절충하고 워크로드와 해당 구성 요소에 대한 기간을 정하세요. 18개월마다 전체 워크로드를 검토하고, 6개월마다 웹 서버를 검토하며, 12개월마다 데이터베이스를 검토하고, 6개월마다 컴퓨팅 및 단기 스토리지를 검토하며, 12개월마다 장기 스토리지를 검토하기로 결정할 수 있습니다.

- 검토 완전성 정의: 워크로드 또는 워크로드 구성 요소를 검토하는 데 얼마나 많은 노력이 드는지 정의합니다. 검토 빈도와 마찬가지로 여러 가지 요소를 비교 검토하여 절충해야 합니다. 개선 기회를 평가하고 우선순위를 지정해 가장 큰 이점이 제공되는 영역에서 작업을 중점적으로 수행하고 동시에 이러한 작업에 어느 정도의 노력이 필요한지 예측합니다. 예상한 성과가 목표에 미치지 못하고 더 많은 노력이 필요하다면 다른 대안을 찾아서 해당 과정을 반복합니다. 개선 가능한 운영 프로세스를 지속적으로 개선하기 위해서는 검토 프로세스에 전담 리소스와 시간을 포함해야 합니다. 예를 들어, 데이터베이스 구성 요소 분석에 1주일, 컴퓨팅 리소스 분석에 1주일, 스토리지 검토에 4시간을 할애하기로 결정할 수 있습니다.

리소스

관련 문서:

- AWS 뉴스 블로그
- 클라우드 컴퓨팅 유형
- AWS의 새로운 소식

관련 예제:

- AWS Support Proactive Services
- SAP 워크로드에 대한 정기 검토를 계획

COST10-BP02 정기적으로 워크로드 검토 및 분석

지정한 각 프로세스를 기준으로 기존 워크로드를 정기적으로 검토하여 새로운 서비스를 채택할 수 있는지, 기존 서비스를 교체할 수 있는지, 워크로드를 재설계할 수 있는지를 확인하세요.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 중간

구현 가이드

AWS는 최신 기술로 보다 빠르게 실험하고 혁신할 수 있도록 새로운 기능을 계속해서 추가하고 있습니다. AWS 새로운 소식에서는 AWS가 이를 수행하는 방법을 자세히 설명하고 AWS 서비스, 기능 및 리전 확장이 출시될 때 간략히 소식을 발표합니다. 공개된 출시 서비스 및 기능을 자세히 살펴보고, 기존 워크로드를 검토하고 분석하는 데 사용할 수 있습니다. 새로운 AWS 서비스 및 기능의 이점을 실현하려면 워크로드에 대한 검토를 실행하고 필요에 따라 새로운 서비스와 기능을 구현해야 합니다. 즉, 워

크로드에 사용하는 기존 서비스를 교체하거나 워크로드를 현대화하여 새로운 AWS 서비스를 채택해야 할 수 있습니다. 예를 들어 워크로드를 검토한 후 메시징 구성 요소를 Amazon Simple Email Service로 대체할 수 있습니다. 이렇게 하면 여러 인스턴스의 운영 및 유지 관리 비용을 없애면서 모든 기능을 더 저렴한 비용으로 제공할 수 있습니다.

워크로드를 분석하고 잠재적인 기회를 강조하려면 최신 서비스뿐만 아니라 새로운 솔루션 구축 방법도 고려해야 합니다. 다른 여러 고객의 아키텍처 설계, 당면 과제 및 솔루션에 대해 알아보려면 AWS에서 This is My Architecture 동영상을 검토하세요. All-In 시리즈에서는 고객 사례와 AWS 서비스의 적용 실례를 살펴볼 수 있습니다. 기본 클라우드 아키텍처 패턴 모범 사례를 설명, 검토 및 분석하는 Back to Basics 동영상 시리즈도 시청할 수 있습니다. 또 다른 소스는 How to Build This 동영상입니다. 이 동영상은 AWS 서비스를 사용하여 최소 기능 제품(MVP)을 실현하는 방법에 대한 아이디어가 있는 사람을 돕고자 마련되었습니다. 이를 통해 아이디어로 똘똘 뭉친 전 세계 빌더가 경험이 풍부한 AWS 솔루션스 아키텍트의 아키텍트 지침을 받을 수 있습니다. 마지막으로, 단계별 자습서로 이루어진 시작하기 리소스 자료를 검토할 수 있습니다.

검토 프로세스를 시작하기 전에 동의한 검토 프로세스를 따르면서 특정 서비스 또는 리전과 성능 요구 사항을 참조하려면 워크로드, 보안 및 데이터 개인 정보 보호 요건에 대한 비즈니스 요구 사항을 준수하세요.

구현 단계

- 정기적 워크로드 검토: 정의된 프로세스를 사용하여 지정된 빈도로 검토를 수행합니다. 각 구성 요소에 대해 적절한 노력을 기울였는지 확인합니다. 이 프로세스는 비용 최적화를 위해 서비스를 선택한 초기 설계 프로세스와 유사합니다. 장기적 이점뿐만 아니라 서비스와 서비스가 제공하는 이점인 변경 비용의 이 시간 요소를 분석합니다.
- 새 서비스 구현: 분석 결과가 변경 실행인 경우 먼저 워크로드의 기준을 수행하여 현재 성과당 비용을 파악합니다. 변경을 실행한 다음 분석을 수행하여 결과별로 새로운 비용을 확인합니다.

리소스

관련 문서:

- AWS 뉴스 블로그
- AWS의 새로운 소식
- AWS 설명서
- AWS Getting Started
- AWS 일반 리소스

관련 비디오:

- AWS - This is My Architecture
- AWS - Back to Basics
- AWS - All-In series
- How to Build This

운영 자동화

모범 사례

- COST11-BP01 운영 자동화 실행

COST11-BP01 운영 자동화 실행

자동화를 통해 관리 작업, 배포, 인적 오류 위험 완화, 규정 준수 및 기타 운영에 드는 시간과 노력을 정량화하는 데 중점을 두고 클라우드에서의 운영 비용을 평가합니다. 운영 노력에 필요한 시간과 관련 비용을 평가하고 가능한 경우 수동 작업을 최소화하기 위해 관리 작업을 자동화합니다.

이 모범 사례가 확립되지 않을 경우 노출되는 위험 수준: 낮음

구현 가이드

운영 자동화를 통해 워크로드를 배포, 관리 또는 운영할 때 일관되고 안정적인 환경을 제공할 수 있어 수동 작업의 빈도를 줄이고, 효율성을 개선하며, 결과적으로 고객에게 이익을 안겨줄 수 있습니다. 인프라 리소스를 수동 운영 작업에 투입할 필요가 없어 대신 가치가 높은 태스크와 혁신에 투자할 수 있으므로 비즈니스 가치가 향상됩니다. 기업은 클라우드에서 워크로드를 관리할 수 있는 검증된 방법을 필요로 합니다. 이 솔루션은 위험과 신뢰성을 극대화하면서 안전하고 빠르며 비용 효율적이어야 합니다.

먼저 전반적인 운영 비용을 살펴보며 필요한 작업을 기준으로 운영 활동의 우선순위를 정하는 것부터 시작하세요. 가령 클라우드에 새 리소스를 배포하거나, 기존 리소스를 최적화하여 변경하거나, 필요한 구성을 구현하는 데 시간이 얼마나 걸리나요? 운영 및 관리 비용을 고려하여 수동 작업으로 인한 총 비용을 살펴봅니다. 관리 작업에 대한 자동화 우선순위를 지정하여 수동 작업의 비율을 줄이세요.

검토 작업에는 잠재적 이점이 반영되어야 합니다. 예를 들어 자동이 아닌 수동으로 작업을 수행하는 데 소요된 시간을 검토할 수 있습니다. 반복적이고 가치가 높으며 시간이 많이 걸리고 복잡한 활동을 자동

화하는 데 우선순위를 둡니다. 작업자가 실수할 위험이 높거나 그 영향력이 큰 활동은 보통 자동화하면 좋습니다. 이러한 위험으로 인해 원치 않는 추가 운영 비용이 발생하는 경우가 많기 때문입니다(예: 운영 팀의 초과 근무).

AWS Systems Manager 또는 AWS Config과 같은 자동화 도구를 사용하여 운영, 규정 준수, 모니터링, 수명 주기 및 종료 프로세스를 간소화합니다. AWS 서비스, 도구 및 서드파티 제품을 사용하여 특정 요구 사항에 맞게 구현하는 자동화를 사용자 지정할 수 있습니다. 다음 표에는 관리 및 운영을 자동화하는 AWS 서비스를 통해 달성할 수 있는 몇 가지 핵심 운영 기능과 특징이 나와 있습니다.

- AWS Audit Manager: AWS 사용량을 지속적으로 감사하여 위험 및 규정 준수 평가를 간소화합니다.
- AWS Backup: 데이터 보호를 중앙에서 관리하고 자동화합니다.
- AWS Config: 컴퓨팅 리소스를 구성하고, 구성 및 리소스 인벤토리를 평가, 감사, 검증합니다.
- AWS CloudFormation: 코드형 인프라를 사용하여 가용성이 높은 리소스를 실행합니다.
- AWS CloudTrail: IT 변경 관리, 규정 준수 및 제어.
- Amazon EventBridge는 이벤트를 예약하고 AWS Lambda를 트리거하여 조치를 취합니다.
- AWS Lambda: 반복 프로세스를 이벤트로 트리거하거나 AWS EventBridge로 확정 일정에 따라 실행하여 자동화합니다.
- AWS Systems Manager: 워크로드 시작 및 중지, 운영 체제 패치, 자동 구성 및 지속적인 관리를 수행합니다.
- AWS Step Functions: 작업을 예약하고 워크플로를 자동화합니다.
- AWS Service Catalog: 템플릿 사용, 규정 준수 및 제어 기능을 갖춘 코드형 인프라.

AWS 제품 및 서비스를 사용하여 즉시 자동화를 채택하고 싶지만, 조직에 기술이 부족한 경우 AWS Managed Services(AMS), AWS Professional Services 또는 AWS 파트너에 문의하여 자동화 채택률을 높이고 클라우드의 운영 효율성을 개선하세요.

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. AMS는 자동화가 포함된 엔터프라이즈 클라우드 운영 모델을 사용하므로 고객은 조직 요구 사항을 충족하면서 클라우드로 더 빠르게 이전하고 지속적인 관리 비용을 절감할 수 있습니다.

AWS Professional Services는 AWS를 통해 원하는 비즈니스 성과를 달성하고 운영을 자동화하는 데 도움이 됩니다. 이를 통해 고객은 클라우드에 최적화된 자동화되고 강력하며 민첩한 IT 운영 및 거버넌스 기능을 배포할 수 있습니다. 자세한 모니터링 예제와 권장 모범 사례는 운영 우수성 원칙 백서를 참조하세요.

구현 단계

- 한 번 구축으로 여러 번 배포: CloudFormation, AWS SDK 또는 AWS CLI와 같은 코드형 인프라를 사용하여 한 번 배포하고 동일한 환경이나 재해 복구 시나리오에 여러 번 사용할 수 있습니다. 배포 중 태그를 지정하여 다른 모범 사례에 나와 있는 대로 소비량을 추적합니다. AWS Launch Wizard를 사용하여 널리 사용되는 엔터프라이즈 워크로드를 배포하는 시간을 줄일 수 있습니다. AWS Launch Wizard에서는 AWS 모범 사례에 따라 엔터프라이즈 워크로드의 크기 조정, 구성 및 배포를 안내합니다. 또한 Service Catalog를 사용하면 누구나 승인된 셀프 서비스 클라우드 리소스를 검색할 수 있도록 AWS에서 사용 가능한 코드형 인프라 승인 템플릿을 생성하고 관리할 수 있습니다.
- 지속적인 규정 준수의 자동화: 사전 정의된 표준에 따라 기록된 구성을 자동으로 평가 및 수정하는 것을 고려해 보세요. AWS Organizations를 AWS Config 및 AWS CloudFormation의 기능과 함께 사용하면 수백 개의 구성원 계정에 대해 대규모 구성 준수를 효율적으로 관리하고 자동화할 수 있습니다. 구성 변경 사항 및 AWS 리소스 간 관계를 검토하고 리소스 구성 기록을 자세히 살펴볼 수 있습니다.
- 모니터링 작업 자동화: AWS에서는 서비스를 모니터링하는 데 사용할 수 있는 다양한 도구를 제공합니다. 이러한 도구를 구성하여 모니터링 작업을 자동화할 수 있습니다. 다중 지점 장애가 발생할 경우 보다 쉽게 디버깅할 수 있도록 워크로드의 모든 부분에서 모니터링 데이터를 수집하는 모니터링 계획을 만들고 구현하세요. 예를 들어, 자동화된 모니터링 도구를 사용하여 시스템 상태 검사, 인스턴스 상태 검사 및 Amazon CloudWatch 경보에 문제가 있는 경우 Amazon EC2를 관찰하여 보고하도록 할 수 있습니다.
- 유지 관리 및 운영 자동화: 사람의 개입 없이 일상적인 작업을 자동으로 실행합니다. AWS 서비스 및 도구를 사용하면 특정 요구 사항에 맞게 구현하고 사용자 지정할 AWS 자동화를 선택할 수 있습니다. 예를 들어 EC2 Image Builder를 사용하여 AWS 또는 온프레미스에서 사용할 가상 머신 및 컨테이너 이미지를 구축, 테스트, 배포하거나 AWS SSM으로 EC2 인스턴스를 패치할 수 있습니다. AWS 서비스로 원하는 작업을 수행할 수 없거나 리소스 필터링을 통해 더 복잡한 작업이 필요한 경우, AWS Command Line Interface(AWS CLI) 또는 AWS SDK 도구를 사용하여 운영을 자동화하세요. AWS CLI는 AWS Management Console을 사용하지 않고 스크립트를 통해 AWS 서비스를 제어하고 관리하는 전체 프로세스를 자동화하는 기능을 제공합니다. AWS 서비스와 상호 작용할 기본 AWS SDK를 선택합니다. 다른 코드 예제는 AWS SDK 코드 예제 리포지토리를 참조하세요.
- 자동화를 통해 지속적인 수명 주기 생성: 규정이나 중복성뿐만 아니라 비용 최적화를 위한 성숙한 수명 주기 정책을 수립하고 유지하는 것이 중요합니다. AWS Backup을 사용하여 버킷, 볼륨, 데이터베이스, 파일 시스템과 같은 데이터 저장소의 데이터 보호를 중앙에서 관리하고 자동화할 수 있습니다. 또한 Amazon Data Lifecycle Manager를 사용하여 EBS 스냅샷 및 EBS 지원 AMI의 생성, 보존 및 삭제를 자동화할 수 있습니다.
- 불필요한 리소스 삭제: 샌드박스 또는 개발 AWS 계정에서 미사용 리소스를 쌓아놓는 경우는 매우 흔합니다. 개발자는 일반적인 개발 주기의 일부로 다양한 서비스와 리소스를 만들고 실험한 다음 더

이상 필요하지 않게 된 리소스를 삭제하지 않습니다. 미사용 리소스는 조직에 불필요하고 때로는 높은 비용을 초래할 수 있습니다. 이러한 리소스를 삭제하면 이러한 환경을 운영하는 데 드는 비용을 절감할 수 있습니다. 확실하지 않은 경우 데이터가 필요하지 않거나 백업되지 않음을 확인하세요. AWS CloudFormation을 사용하여 배포된 스택을 정리할 수 있습니다. 이렇게 하면 템플릿에 정의된 대부분의 리소스가 자동으로 삭제됩니다. 또는 aws-nuke와 같은 도구를 사용하여 AWS 리소스 삭제를 자동화할 수 있습니다.

리소스

관련 문서:

- Modernizing operations in the AWS 클라우드
- AWS Services for Automation
- Infrastructure and automation
- AWS Systems Manager Automation
- 자동 및 수동 모니터링
- AWS automations for SAP administration and operations
- AWS Managed Services
- AWS Professional Services

관련 비디오:

- Automate Continuous Compliance at Scale in AWS
- AWS Backup Demo: Cross-Account & Cross-Region Backup
- Patching for your Amazon EC2 Instances

관련 예제:

- Reinventing automated operations (Part I)
- Reinventing automated operations (Part II)
- Automate deletion of AWS resources by using aws-nuke
- Delete unused Amazon EBS volumes by using AWS Config and AWS SSM
- Automate continuous compliance at scale in AWS
- IT Automations with AWS Lambda

결론

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

AWS에서는 복원력과 응답성이 뛰어난 적응형 배포를 구축하면서 비용을 최소화하도록 지원합니다. 이 백서에서 설명하는 도구, 기술 및 모범 사례를 활용하면 배포의 비용을 최대한 최적화할 수 있습니다.

기여자

다음은 이 문서의 기여자입니다.

- Fatih(Ben) Mergen, Amazon Web Services의 Well-Architected, Cost Optimization Pillar Lead
- Keith Jarrett, Amazon Web Services의 Business Development Lead - Cost Optimization
- Arthur Basbaum, Amazon Web Services의 Business Developer Manager
- Jarman Hauser, Amazon Web Services의 Commercial Architect

참조 자료

자세한 내용은 섹션을 참조하세요.

- AWS Well-Architected Framework
- AWS 아키텍처 센터

문서 수정

이 백서에 대한 업데이트 알림을 받으려면 RSS 피드를 구독하면 됩니다.

변경 사항	설명	날짜
모범 사례 지침 업데이트됨	여러 모범 사례 업데이트. 새로운 모범 사례 COST06-BP04.	2024년 6월 27일
모범 사례 지침 업데이트됨	전반적인 소규모 모범 사례 업데이트.	2023년 12월 6일
모범 사례 지침 업데이트됨	핵심 분야 전반에 걸쳐 새로운 지침으로 모범 사례가 업데이트되었습니다.	2023년 10월 3일
모범 사례 지침 업데이트됨	거버넌스, 비용 및 사용량 모니터링, 최적의 가격 책정 모델 선택, 수요 및 공급 리소스 관리와 같은 영역에 대한 새로운 지침으로 모범 사례가 업데이트되었습니다.	2023년 7월 13일
새 프레임워크 관련 업데이트	권장 가이드 및 새로운 모범 사례를 추가하여 모범 사례를 업데이트했습니다. 새로운 COST11-BP01 모범 사례와 함께 COST 11 질문을 추가했습니다.	2023년 4월 10일
마이너 업데이트	가격 책정 모델 섹션에 누락된 지침이 복원되었습니다.	2023년 1월 13일
백서 업데이트	새로운 구현 가이드와 함께 모범 사례를 업데이트했습니다.	2022년 12월 15일
백서 업데이트	모범 사례를 확대하고 개선 계획을 추가했습니다.	2022년 10월 20일

마이너 업데이트	소개에 지속 가능성 원칙을 추가했습니다.	2021년 12월 2일
마이너 업데이트	링크를 업데이트했습니다.	2021년 4월 25일
마이너 업데이트	링크를 업데이트했습니다.	2021년 3월 10일
새 프레임워크 관련 업데이트	CFM, 새로운 서비스 및 Well-Architected와의 통합도 포함하도록 업데이트되었습니다.	2020년 7월 8일
백서 업데이트	AWS에 대한 변경 사항을 반영하고 고객 검토에서 파악한 내용을 포함하도록 업데이트했습니다.	2018년 7월 1일
백서 업데이트	AWS에 대한 변경 사항을 반영하고 고객 검토에서 파악한 내용을 포함하도록 업데이트했습니다.	2017년 11월 1일
최초 게시	비용 최적화 원칙 - AWS Well-Architected Framework를 게시했습니다.	2016년 11월 1일

고지 사항

고객은 본 문서의 정보를 독립적으로 평가할 책임이 있습니다. 본 문서는 (a) 정보 제공의 목적으로만 제공되고, (b) 사전 통지 없이 변경될 수 있는 현재 AWS 제품 및 관행을 나타내고, (c) AWS 및 그 계열사, 공급업체 또는 라이선스 제공자로부터 어떠한 약속이나 보증도 하지 않습니다. AWS 제품 또는 서비스는 명시적이든 묵시적이든 어떠한 종류의 보증, 진술 또는 조건 없이 '있는 그대로' 제공됩니다. 고객에 대한 AWS의 책임 및 채무는 AWS 계약에 준거합니다. 본 문서는 AWS와 고객 간의 어떠한 계약도 구성하지 않으며 이를 변경하지도 않습니다.

© 2023 Amazon Web Services, Inc. 또는 계열사. All rights reserved.

AWS 용어집

최신 AWS 용어는 AWS 용어집 참조서의 AWS 용어집을 참조하십시오.