



Microsoft 워크로드를 마이그레이션하기 위한 옵션, 도구 및 모범 사례 AWS

AWS 규범적 지침



AWS 규범적 지침: Microsoft 워크로드를 마이그레이션하기 위한 옵션, 도구 및 모범 사례 AWS

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon의 상표 및 브랜드 디자인은 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

소개	1
개요	1
수강 대상	1
목표 비즈니스 성과	2
Microsoft 워크로드에 AWS를 선택하는 이유는 무엇일까요?	3
기본 모범 사례	5
클라우드로 가는 길	6
마이그레이션 전략	6
주요 변환	6
마이그레이션 전략 선택	7
Windows 마이그레이션 프로세스	9
평가	9
동원	10
마이그레이션 및 현대화	10
윈도우 환경 디스커버리	12
평가	12
엔터프라이즈 아키텍처	12
표준화 및 구성 관리	12
좋은 데이터	13
자동화	13
세부 계획	13
동원	13
대규모 마이그레이션의 어려움	14
지연 시간에 민감한 종속성	14
IT 공유 서비스	14
구성 업데이트	14
애플리케이션 기능 테스트	15
애플리케이션 종속성 검색을 위한 도구	15
Microsoft 워크로드 마이그레이션	16
Active Directory 마이그레이션	16
평가	16
동원	18
마이그레이션	21
Windows Server 마이그레이션	23

평가	23
동원	24
마이그레이션	25
파일 서버 마이그레이션	25
평가	26
동원	27
마이그레이션	28
SQL Server 마이그레이션	28
평가	29
동원	30
마이그레이션	30
.NET 애플리케이션 마이그레이션	34
평가	34
동원	35
마이그레이션	35
리플랫폼	37
추가적인 리소스	39
Windows 장애 조치 클러스터 마이그레이션	40
평가	41
동원	43
마이그레이션	43
Microsoft 워크로드 모니터링	44
평가	44
동원	45
마이그레이션	45
마이그레이션 도구, 프로그램 및 교육	47
도구	47
평가 도구	47
마이그레이션 도구	50
마이그레이션 파트너 도구	52
관리 도구	52
프로그램	54
학습	55
AWS에서의 Microsoft 라이선스	56
평가	56
라이선스 포함 옵션	57

BYOL 옵션	58
Amazon EC2 전용 호스트	62
AWS의 VMware Cloud	63
동원	63
AWS License Manager	63
라이선싱 고려 사항	64
마이그레이션	64
AWS 파트너	66
AWS 컴피턴시 파트너 참여의 이점	66
계획 수립	66
비용 최적화	66
시간을 절약하세요.	67
보안 강화	68
다음 단계	69
리소스	70
문서 기록	71
용어집	73
#	73
A	74
B	76
C	78
D	81
E	85
F	87
G	88
H	89
I	90
L	92
M	93
O	97
P	99
Q	101
R	102
S	104
T	108
U	109

V	109
W	110
Z	111
.....	cxii

Microsoft 워크로드를 AWS로 마이그레이션하기 위한 옵션, 도구 및 모범 사례

제롤 헤어우드, 크리스틴 메깃, 드로 헬퍼, 다니엘 말도나도, 필 에킨스, 마니 파흐난다, 싯다르트 메타, 리치 베누아, 룬 히가레다, 살레하 하이다, 시아바시 이라니, 요기 바로트, Amazon Web Services (AWS)

2024년 4월 ([문서 기록](#))

개요

조직은 10년 이상 AWS에서 Microsoft 워크로드를 마이그레이션하고 실행해 왔으며, 이는 다른 어떤 클라우드 공급자보다도 긴 기간입니다. 이 안내서는 AWS가 수년간 마이그레이션 및 현대화 노력을 통해 얻은 지식과 전문성을 바탕으로 Microsoft 워크로드를 AWS 클라우드로 마이그레이션하는 작업을 간소화하도록 설계되었습니다. 이 안내서를 사용하여 Windows 마이그레이션의 모든 단계를 계획하고 구현할 수 있습니다. 이 가이드는 다음을 비롯한 다양한 마이그레이션 사용 사례에 적용됩니다.

- 조직의 디지털 혁신 및 현대화 여정의 일환으로 Windows 마이그레이션을 시작하고 있습니다.
- Microsoft 워크로드를 실행하는 데이터 센터의 임대 기간이 거의 만료되었습니다.
- 가용성 요구 사항이 각기 다른 다양한 Windows 애플리케이션을 사용하고 있지만 지리적으로 분산된 위치에 워크로드를 배포할 리소스가 없습니다.

이 안내서에서는 AWS Migration Hub, AWS 애플리케이션 마이그레이션 서비스 등과 같이 마이그레이션 과정을 간소화하는 데 도움이 되는 다양한 AWS 도구에 대해 알아봅니다. 이 안내서는 AWS 모범 사례에 맞춰 평가, 동원, [마이그레이션 및 현대화라는 3단계 AWS 마이그레이션 프로세스](#)를 따릅니다. 이 프로세스는 Windows 마이그레이션을 구조화하고 간소화하는 데 도움이 될 수 있는 오랜 테스트를 거친 마이그레이션 프레임워크를 기반으로 합니다. 평가 단계에서는 클라우드에서의 운영 준비 상태를 평가합니다. 동원 단계에서는 마이그레이션 계획 초안을 작성하고 평가 단계에서 파악한 준비 격차를 해소합니다. 그런 다음 자동화 도구와 템플릿을 조합하여 워크로드를 체계적으로 마이그레이션하고 비즈니스 요구 사항을 충족함으로써 마이그레이션 및 현대화 단계에서 워크로드를 마이그레이션하기 시작합니다.

수강 대상

이 안내서는 IT 아키텍트, 마이그레이션 책임자, 기술 책임자, AWS 파트너 팀 및 다음을 담당하는 기타 역할을 대상으로 합니다.

- Microsoft 워크로드를 데이터 센터에서 AWS 클라우드로 마이그레이션하기
- AWS 클라우드에서의 윈도우 환경 관리

목표 비즈니스 성과

이 안내서는 여러분과 조직이 다음 목표를 달성하는 데 도움이 될 수 있습니다.

1. Microsoft 워크로드를 AWS로 마이그레이션하는 데 사용할 수 있는 전략, 프로그램 및 서비스에 대해 알아보십시오.
2. 액티브 디렉터리, 윈도우 파일 서버, SQL 서버, .NET 워크로드와 같은 특정 Microsoft 워크로드에 대한 AWS 마이그레이션 경로를 이해하십시오.
3. 보안, 가용성 및 안정성 요구 사항을 충족하면서 AWS에서 Microsoft 워크로드를 실행하십시오.
4. AWS에서 Microsoft 워크로드를 실행하기 위한 라이선스 모범 사례를 숙지하십시오.

Microsoft 워크로드에 AWS를 선택하는 이유는 무엇일까요?

AWS는 14년 넘게 고객이 Microsoft 워크로드를 마이그레이션하고 현대화하도록 지원해 왔으며 비즈니스를 지원하는 주요 애플리케이션의 전환을 가속화하는 가장 광범위한 서비스, 프로그램 및 전문 지식 포트폴리오를 보유하고 있습니다. AWS를 사용하여 마이그레이션하고 현대화하면 다음과 같은 이점을 기대할 수 있습니다.

- 혁신을 실현하세요— 기존의 모놀리식 아키텍처에서 클라우드 기반 마이크로서비스 아키텍처로 전환하면 자유롭게 적응하고 신속하게 실험하여 조직이 혁신을 더 빠르게 실현할 수 있습니다. AWS는 아마존 엘라스틱 컨테이너 서비스 (Amazon ECS), 아마존 엘라스틱 쿠버네티스 서비스 (Amazon EKS), AWS Fargate를 포함하여 가장 광범위한 컨테이너 기술을 보유하고 있습니다. 또한 AWS는 가장 완성도가 높은 서버리스 오퍼링 (AWS Lambda) 과 긴밀하게 통합된 .NET 지원을 제공합니다. DevOps 개발 주기를 자동화하는 유틸리티, 여러 오픈 소스 통합, 모던 아키텍처를 지원하는 Amazon Aurora와 같은 특수 목적의 데이터베이스
- 비용 절감— 오픈 소스 데이터베이스 솔루션으로 전환하면 값비싼 Windows 또는 SQL Server 라이선스 비용을 지불하지 않아도 됩니다. 예를 들어 Aurora는 1/10의 비용으로 상용 데이터베이스와 동일한 기능을 제공합니다. 로 이사하는 경우 DevOps 컨테이너와 서버리스 솔루션을 사용하면 총소유 비용 (TCO) 을 줄이고 컴퓨팅 소비를 극대화할 수 있습니다.
- 보안 향상— AWS는 230개의 보안, 규정 준수, 거버넌스 서비스와 주요 기능을 제공합니다. 이는 두 번째로 큰 클라우드 공급자보다 5배 많은 서비스입니다. 사용할 수 있습니다 [AWS 디렉터리 서비스](#) AWS 관리형 Microsoft AD라고도 하며, 클라우드 보안을 개선하고 마이그레이션 중에 기존 Active Directory의 데이터를 동기화하거나 복제할 필요가 없습니다. 또한 사용할 수 있습니다 [AWS 아이덴티티 서비스](#) 대규모로 ID 및 권한을 관리하는 동시에 직원, 파트너 및 고객 정보를 관리하는 위치와 방법에 대한 유연한 옵션을 제공합니다.
- 신뢰할 수 있는 전문가와 함께 기술 개발— AWS는 수백만 개의 조직이 고유한 도구와 서비스를 통해 마이그레이션 목표를 더 빨리 달성할 수 있도록 지원한 타의 추종을 불허하는 경험을 가지고 있습니다. 더 [원도우용 AWS 마이그레이션 가속화 프로그램 \(MAP\)](#) AWS 파트너 및 AWS 전문 서비스의 지원을 받아 클라우드로 마이그레이션하는 데 따르는 복잡성과 비용을 줄일 수 있는 모범 사례, 도구 및 인센티브를 제공합니다. 더 [Windows 서버용 지원 종료 마이그레이션 프로그램](#) 레거시 Windows Server 애플리케이션을 AWS에서 지원되는 최신 버전의 Windows Server로 마이그레이션하는 데 도움을 줄 수 있습니다. Fortune 100대 기업의 90% 와 Fortune 500대 기업의 대다수가 AWS 파트너 솔루션 및 서비스를 사용합니다.
- 프로세싱 파워의 가격 및 성능을 개선하세요— AWS는 Intel x86 기반 인스턴스보다 시간당 20% 저렴하고 성능이 최대 40% 향상된 Graviton2 기반 인스턴스를 제공하는 프로세싱 혁신의 선두 주자입

니다. 또한 Aurora는 표준 MySQL의 5배, 표준 PostgreSQL의 3배 높은 처리량을 제공합니다. 이 성능은 상용 데이터베이스와 동등하며 비용은 10분의 1입니다.

- 유연한 라이선스 옵션 활용— AWS는 AWS에서 신규 및 기존 Microsoft 소프트웨어 라이선스를 사용할 수 있는 가장 많은 옵션을 클라우드에서 제공합니다. 라이선스가 포함된 Amazon Elastic Compute Cloud (Amazon EC2) 또는 Amazon 관계형 데이터베이스 서비스 (Amazon RDS) 인스턴스를 구매하면 AWS로부터 완전히 규정을 준수하는 새로운 SQL Server 라이선스를 받게 됩니다. 다음을 사용하여 기존 라이선스를 AWS로 가져올 수 있습니다. [아마존 EC2 전용 호스트](#), [아마존 EC2 전용 인스턴스](#) 또는 다음을 사용하여 기본 테넌시를 사용하는 EC2 인스턴스 [소프트웨어 어슈어런스를 통한 Microsoft 라이선스 모빌리티](#). AWS License Manager를 사용하면 소프트웨어 라이선스 사용을 쉽게 추적하고 규정 위반 위험을 줄일 수 있습니다.

자세한 내용은 [을 참조하십시오](#). [AWS 기반 윈도우](#) AWS 설명서에서 확인할 수 있습니다.

기본 모범 사례

AWS 마이그레이션을 위한 확장 가능하고 안전한 기반을 구축하면 AWS에서 Windows 환경을 쉽게 관리하고 효율적으로 실행할 수 있습니다. Microsoft 워크로드를 AWS로 마이그레이션하기 전에 다음과 같은 기본 모범 사례를 고려하는 것이 좋습니다.

- Microsoft 라이선스 비용 최적화 — 라이선스는 앞으로의 다른 모든 결정에 영향을 미치기 때문에 클라우드 마이그레이션의 중요한 요소입니다. 라이선스 옵션을 최대한 빨리 파악하는 것이 좋습니다. 라이선스에 대한 자세한 내용은 이 안내서의 [AWS 기반 Microsoft 라이선스](#) 섹션을 참조하십시오.
- 클라우드 아키텍처 간소화 — [AWS Well-Architected Framework](#)를 사용하면 클라우드에서 워크로드를 안정적으로 실행할 수 있습니다. 프레임워크를 따르고, 심각한 문제를 피하고, 조직의 요구 사항에 맞게 규모를 조정하는 데 도움이 되는 지침과 전략을 받게 됩니다. 이 지침에서는 청구, 액세스 제어, 보안 제어도 다룹니다.
- 통합된 easy-to-manage 클라우드 네트워크 구축 — [AWS Transit Gateway](#)를 사용하면 네트워크를 더 쉽게 관리하고, 온프레미스 또는 다른 클라우드 환경에서 CIDR (Classless Inter-Domain Routing) 범위 계획과 같은 중복 네트워크가 생성되는 것을 방지할 수 있습니다. 이렇게 하면 필요에 따라 트래픽을 각 네트워크로 라우팅할 수 있습니다. 계정이 서로 라우팅되고 온-프레미스 환경 및 인터넷으로 라우팅되는 방식을 결정해야 합니다. 이를 통해 적절한 제어를 설정하여 네트워크 트래픽을 보호할 수 있습니다. 예를 들어, AWS 계정을 기존 온-프레미스 데이터 센터의 확장으로 만들고 방화벽, 침입 탐지 시스템 (IDS), 침입 방지 시스템 (IPS) 과 같은 경계 방어 수단을 사용하거나, AWS 리소스를 보호하기 위해 이러한 경계 방어를 포함하는 AWS 네트워크 계정을 설정해야 합니다.
- 클라우드 보안 우선 순위 지정 — 최소 권한 권한을 적용하는 보안 모범 사례를 준수하면서 단일 계정에서 다중 계정 환경으로 전환하는 것이 좋습니다. 또한 [AWS 공동 책임 모델을 철저히 이해](#) 하고 조직의 민첩성을 유지하면서 [환경을 보호할](#) 수 있는 방법을 계획하는 것이 좋습니다. 보안을 개선하고 유지하기 위해 Amazon API Gateway, AWS WAF, 애플리케이션 로드 밸런서, Amazon, CloudWatch AWS GuardDuty, CloudTrail Amazon 및 기타 서비스를 사용할 수 있습니다. 다중 계정 전략에 대해 자세히 알아보려면 AWS Prerequisite Guidance 설명서에서 여러 [AWS 계정으로의 전환](#) 을 참조하십시오.
- 클라우드에서의 공유 IT 서비스 관리 — 클라우드에서 워크로드를 효율적으로 관리하려면 워크로드에 사용되는 모든 공유 서비스를 식별하고 클라우드에서 이를 제공할 방법을 계획하는 것이 중요합니다. 예를 들어 Active Directory, 파일 서버, SQL 데이터베이스, DNS (도메인 이름 시스템), VPN (가상 사설망), SMTP (단순 메일 전송 프로토콜), 백업 및 모니터링 서비스가 여기에 포함됩니다. 인벤토리를 작성한 후에는 기존 서비스를 클라우드로 확장할지, 완전히 새로운 서비스 인스턴스를 설정할지, 아니면 대체 관리형 클라우드 서비스를 사용할지 결정할 수 있습니다. 이 가이드의 후속 섹션에서는 이러한 고려 사항을 더 자세히 다룰 것입니다.

클라우드로 가는 길

이 섹션에서는 Windows 애플리케이션을 AWS로 마이그레이션하기 위한 모범 사례를 구현하기 위한 상위 수준의 접근 방식을 설명합니다. 이러한 마이그레이션 전략 및 단계에 대한 자세한 내용은 이 가이드의 후속 섹션에 설명되어 있습니다.

마이그레이션 전략

마이그레이션 전략은 워크로드를 AWS 클라우드로 마이그레이션하는 데 사용되는 접근 방식입니다. 애플리케이션을 클라우드로 이전하기 위한 7가지 마이그레이션 전략이 있습니다. 이러한 전략은 7R로 알려져 있으며 다음을 기반으로 합니다. [7 Rs](#)가트너가 2019년에 확인한 내용입니다.

- 리호스트 (리프트 앤 시프트)— 변경 없이 애플리케이션을 클라우드로 이동하여 클라우드 기능을 활용할 수 있습니다.
- 재배포 (하이퍼바이저 수준의 리프트 앤 시프트)— 새 하드웨어를 구입하거나, 애플리케이션을 재작성하거나, 기존 운영을 수정하지 않고도 인프라를 클라우드로 이전할 수 있습니다.
- 플랫폼 변경 (리프트 및 모양 변경)— 애플리케이션을 클라우드로 이전하고 일정 수준의 최적화를 도입하여 클라우드 기능을 활용할 수 있습니다.
- 재구매 (드롭 앤 샵)— 일반적으로 기존 라이선스에서 SaaS (Software as a Service) 모델로 전환하여 다른 제품으로 전환합니다.
- 리팩터링/리아키텍처— 클라우드 네이티브 기능을 최대한 활용하여 애플리케이션을 이동하고 아키텍처를 수정하여 민첩성, 성능 및 확장성을 개선할 수 있습니다.
- 유지 (재방문)— 애플리케이션을 소스 환경에 보관합니다. 여기에는 대대적인 리팩토링이 필요하고 해당 작업을 나중에 연기하려는 애플리케이션과 마이그레이션할 비즈니스 근거가 없으므로 보존하려는 레거시 애플리케이션이 포함될 수 있습니다.
- 은퇴— 소스 환경에서 더 이상 필요하지 않은 애플리케이션을 폐기하거나 제거할 수 있습니다.

주요 변환

레거시 Windows 애플리케이션 및 데이터베이스를 현대화하면 다음과 같은 주요 변환이 이루어집니다.

- 호스트 변경— 첫 번째 단계는 온프레미스 인프라를 클라우드 인프라로 이전하는 것입니다. 이 전략을 흔히 “리프트 앤 시프트” 또는 리호스팅이라고 합니다. 재호스팅이란 기존 애플리케이션과 데이터

베이스를 클라우드 서버 인스턴스로 마이그레이션하는 것을 의미합니다. 코드를 변경할 필요가 없으며 사용자는 인스턴스 구성, 소프트웨어 이미지 및 기타 리소스를 관리할 책임이 있습니다.

- 플랫폼 변경— 클라우드 환경으로 마이그레이션한 후 다음 혁신은 애플리케이션과 데이터베이스를 더욱 자동화되고 관리되는 환경으로 재배포하는 것입니다. 애플리케이션 관점에서 보면 이는 가상 머신 (VM) 에서 컨테이너로 이동하는 것을 의미합니다. 애플리케이션을 컨테이너화하면 애플리케이션을 더 빠르게 개발, 유지 관리 및 배포하고 이식성을 개선할 수 있습니다. AWS에는 다음과 같은 도구가 있습니다. [AWS 앱2 컨테이너](#), 레거시 애플리케이션을 컨테이너화하는 프로세스를 자동화하는 데 도움이 됩니다. 데이터베이스 측면에서는 셀프 서비스 모델에서 Amazon RDS for SQL Server와 같은 관리형 데이터베이스 서비스로 전환하면 프로비저닝, 패치 적용 및 백업이 필요하지 않습니다. 이를 통해 궁극적으로 조직에 더 많은 가치를 더할 수 있는 활동에 리소스를 확보할 수 있습니다.
- 리팩터링/리아키텍처— 전환의 세 번째 영역은 상용 소프트웨어 라이선싱에서 오픈 소스 옵션으로 전환하는 것입니다. 많은 기존 상용 소프트웨어 공급업체들은 소프트웨어 라이선스 계약을 기반으로 비즈니스를 구축해 왔습니다. 이 계약은 고객을 고정시키고 징벌적인 라이선스 조건을 사용하여 업그레이드와 마이그레이션을 강요하는 것을 목표로 합니다. 상용 소프트웨어 라이선스 수수료는 일반적으로 동등한 오픈 소스 옵션 외에 20-50% 의 비용을 추가합니다. 비용을 절감하고 성능을 개선하며 최신 혁신 기술에 액세스할 수 있도록 애플리케이션과 데이터베이스를 리팩토링하여 오픈 소스 옵션을 활용하는 것이 좋습니다.

애플리케이션과 현대화를 위한 전반적인 준비 상태에 따라 이러한 주요 혁신 영역을 단계적으로 또는 한 번에 모두 완료할 수 있습니다.

마이그레이션 전략 선택

선택할 마이그레이션 전략은 조직의 비즈니스 및 IT 목표에 따라 달라집니다. 가장 일반적인 비즈니스 동인으로는 비용 절감, 위험 감소, 효율성 향상, 기술 격차 해소, 혁신 가속화가 있습니다. 어떤 드라이버가 중요한지 평가한 후 다음 지침을 사용하여 해당 요인을 기반으로 마이그레이션 전략을 선택하는 것이 좋습니다. 또한 세 가지 접근 방식 모두 여정의 각 단계에서의 우선 순위에 따라 클라우드 현대화 여정에서 가능한 길이라는 점을 기억하십시오.

재호스팅 시기

애플리케이션에서 코드나 아키텍처를 변경할 필요가 없기 때문에 일반적으로 재호스팅 (또는 리프트 앤 시프트) 이 더 빠르고 쉽습니다. 재호스팅은 또한 비즈니스에 대한 위험과 중단을 최소화합니다. 애플리케이션이 변경되지 않기 때문에 운영 팀은 평소처럼 비즈니스를 계속 운영할 수 있습니다. 이는 많은 워크로드가 수반되기 때문에 작은 변화라도 중요해지는 대규모 마이그레이션의 경우 특히 그렇습니다. 그러나 재호스팅이 클라우드의 이점을 충분히 활용하지는 못한다는 점을 고려하는 것이 중요합니다. 예를 들어 기존 플랫폼 문제가 있는 애플리케이션을 마이그레이션하는 경우 해당 문제는 마이그

레이션 후에도 그대로 유지됩니다. 마지막으로, 재호스팅을 위한 총 소유 비용 (TCO) 과 투자 수익률 (ROI) 이 다른 마이그레이션 방식에 비해 낮은 점을 고려해 볼 가치가 있습니다.

플랫폼 재구성/재설계 시기

일반적으로 플랫폼 재구축은 재호스팅보다 비용 효율적입니다. 플랫폼 재구성을 통해 자동화를 강화하고 애플리케이션이 자동 크기 조정, 모니터링 및 백업 수행과 같은 클라우드 기능을 더 잘 사용하도록 할 수 있습니다. 플랫폼 재구축은 클라우드 운영 팀의 운영 오버헤드를 줄이고 기존 플랫폼 문제로 인한 위험을 최소화합니다. 하지만 플랫폼 재구축은 재호스팅 마이그레이션보다 시간이 더 오래 걸립니다. 또한 플랫폼 재구축에는 애플리케이션에서 코드 변경을 수행하는 자동화를 구성하고 새 플랫폼을 운영하기 위한 추가 기술이 필요합니다.

리팩토링 시기

리팩토링은 일반적으로 가장 비용 효율적인 마이그레이션 접근 방식입니다. 리팩토링은 애플리케이션 구성 요소를 분리하여 애플리케이션 레질리언스를 개선함으로써 애플리케이션이 새로운 요구 사항에 빠르게 적응할 수 있도록 하는 클라우드 네이티브 접근 방식입니다. 그러나 리팩토링에는 고급 코딩 및 자동화 기술이 필요합니다. 또한 리팩토링은 애플리케이션을 재구축해야 하기 때문에 구현 시간이 더 오래 걸립니다.

Windows 마이그레이션 프로세스

기존 Windows 환경을 AWS로 마이그레이션하려면 신중한 계획과 구현이 필요합니다. 이 프로세스에는 현재 리소스 사용량을 파악하고, AWS로 마이그레이션할 때 얻을 수 있는 비용 절감 가능성을 평가하고, 보안 요구 사항을 파악하고, 조직의 모든 요구 사항을 충족하는 잘 정의된 클라우드 아키텍처를 구축하는 작업이 포함됩니다. AWS를 사용하여 현재 Windows 서버 인프라를 빠르고 쉽게 마이그레이션하여 운영 비용을 절감하는 동시에 시스템 효율성을 극대화할 수 있습니다. 또한 AWS는 전체 프로세스에 대한 제어를 유지하고 클라우드의 Windows 환경이 최대 성능을 발휘하도록 최적으로 구성되도록 하는 데 도움이 되는 다양한 강력한 도구 및 서비스를 제공합니다.

이 섹션에서는 조직이 AWS 클라우드로 여러 애플리케이션을 성공적으로 마이그레이션할 수 있도록 지원하기 위해 AWS가 개발한 3단계 마이그레이션 프로세스(평가, 이동, 마이그레이션 및 현대화)의 개요를 제공합니다.

평가

평가 단계는 클라우드로 전환하려는 조직의 준비 상태를 이해하는 데 도움이 됩니다. AWS 도구를 사용하면 온프레미스 컴퓨팅 리소스를 평가하고 AWS에서 애플리케이션을 실행하기 위한 비용 예측을 구현하여 평가 단계를 지원할 수 있습니다. 다음 도구를 고려하는 것이 좋습니다.

- [마이그레이션 준비 상태 평가](#)를 사용하면 클라우드 여정의 어느 단계에 있는지 파악할 수 있습니다.
- [AWS Optimization and Licensing Assessment\(AWS OLA\)](#)를 사용하여 실제 리소스 사용률, 타사 라이선스, 애플리케이션 종속성을 바탕으로 현재 온프레미스 및 클라우드 환경을 평가하고 최적화할 수 있습니다.
- [마이그레이션 평가기](#)를 사용하면 AWS로 마이그레이션하기 위한 데이터 기반 비즈니스 사례를 구축하는 데 도움이 됩니다.
- [Cloud Economics Center](#)로 신뢰성 향상, 비용 최적화, 확장성과 같은 목표를 정의하여 마이그레이션에 대한 비즈니스 사례를 구축할 수 있습니다.
- [AWS Migration Hub](#)를 사용하여 마이그레이션 평가, 계획 및 추적을 위한 서버 및 애플리케이션 인벤토리 데이터를 수집할 수 있습니다.
- [Migration Validator Toolkit PowerShell 모듈](#)을 사용하여 Microsoft 워크로드를 검색하고 AWS로 마이그레이션할 수 있습니다.

동원

동원 단계에서는 마이그레이션 계획을 개발하고 비즈니스 계획을 반복하여 평가 단계에서 드러난 준비 상태의 격차를 해소합니다. 기본 환경을 구축하고, 운영 준비를 강화하고, 클라우드 기술을 개발하는 데 집중하는 것이 중요합니다. 대규모 애플리케이션 포트폴리오를 마이그레이션하는 것은 복잡한 작업일 수 있습니다. 이 프로세스를 쉽게 하기 위해 AWS는 파일럿 워크로드 세트를 클라우드로 빠르고 안전하고 비용 효율적으로 마이그레이션하는 데 도움이 되는 다양한 도구와 서비스를 제공합니다. 애플리케이션 포트폴리오에서 데이터를 수집하고 7가지 일반적인 마이그레이션 전략(리호스팅, 재배치, 리플랫폼, 재구매, 리팩터링/리아키텍트, 유지, 사용 중지) 중 하나 이상을 사용하여 애플리케이션을 합리화하면 의사 결정을 위한 더 나은 기반을 제공할 수 있습니다. AWS는 Windows 기반 애플리케이션과 워크로드를 클라우드로 마이그레이션하는 데 사용할 수 있는 다음과 같은 서비스 제품군을 제공합니다.

- [AWS Application Discovery Service](#)
- [AWS Application Migration Service](#)
- [AWS Database Migration Service](#)
- [AWS 마이그레이션 컴피턴시 파트너](#)
- [AWS에서의 관리 및 거버넌스](#)
- [AWS Control Tower](#)

마이그레이션 및 현대화

마이그레이션 및 현대화 단계에서는 마이그레이션 범위 내에 있는 각 애플리케이션을 신중하게 설계, 마이그레이션 및 검증해야 합니다. 애플리케이션 마이그레이션 서비스를 사용하면 많은 수의 서버를 물리적, 가상 또는 클라우드 인프라에서 AWS로 쉽게 마이그레이션할 수 있습니다. 애플리케이션 마이그레이션 서비스를 사용하면 다양한 애플리케이션에 동일한 자동 프로세스를 사용하고 기존 환경에서 클라우드로 신속하게 애플리케이션을 리프트 앤드 시프트할 수 있습니다.

[AWS의 클라우드 마이그레이션 팩토리](#) 솔루션은 상당한 수의 서버를 포함하는 대규모 마이그레이션의 수동 프로세스를 조정하고 자동화하도록 설계되었습니다. 이 솔루션은 워크로드를 AWS로 대규모 마이그레이션할 수 있는 오케스트레이션 플랫폼을 제공하여 성능을 개선하고 긴 전환 기간을 방지하는 데 도움이 됩니다. [AWS Professional Services](#), [AWS 파트너](#) 및 기타 기업에서는 이미 이 솔루션을 사용하여 고객이 수천 대의 서버를 AWS 클라우드로 마이그레이션하도록 지원했습니다.

마이그레이션이 완료되면 [AWS Migration Hub Refactor Spaces](#)를 사용하여 AWS용 애플리케이션을 리팩터링할 때 획일적인 작업을 줄일 수 있습니다. Refactor Spaces는 개발자가 오버헤드나 중단을 최

소화하면서 기존 애플리케이션을 최신 아키텍처로 점진적으로 리팩터링할 수 있는 사용하기 쉬운 작업 공간을 제공합니다. Refactor Spaces를 사용하여 애플리케이션에 최적화된 전체 AWS 서비스를 빠르게 활용할 수 있습니다.

귀사의 팀은 온프레미스에서 Microsoft 워크로드를 구축하고 실행하는 전문가입니다. 클라우드에서 이러한 경험을 개선할 수 있습니다. AWS로 마이그레이션하면 신뢰할 수 있는 Windows 환경에서 훨씬 더 효율적이고 안정적인 경험을 제공할 수 있습니다. AWS를 사용하면 기존 Microsoft 워크로드를 더 쉽고 빠르게 마이그레이션할 수 있도록 설계된 다양한 클라우드 서비스에 액세스할 수 있습니다. 더 확장 가능한 용량, 개선된 스토리지 옵션, 향상된 보안을 활용할 수 있습니다.

윈도우 환경 디스커버리

애플리케이션 마이그레이션 서비스와 같은 현재 사용 가능한 기술을 사용하면 Windows Server, Linux 및 기타 x86 기반 운영 체제와 해당 워크로드를 AWS로 이전하는 것이 매우 간단합니다. 그러나 이러한 워크로드가 제대로 작동하도록 하고 대규모로 수행하는 것은 다른 과제를 제시합니다. 이 섹션은 Microsoft 워크로드를 빠르고 안전하며 원활하게 마이그레이션할 수 있는 마이그레이션 고려 사항을 파악하기 위한 것입니다.

평가

최소한의 계획과 자동화로 소규모 마이그레이션 (예: 서버 100대 포함) 을 “강제 실행”할 수는 있지만 이 방법으로는 500대 이상의 서버를 이동할 수 없습니다. 다음은 성공적인 대규모 마이그레이션을 위한 주요 고려 사항이며 다음과 같은 고려 사항을 사용할 수 있습니다. [마이그레이션 준비 평가 \(MRA\)](#) 중 점적으로 고려하고 싶은 분야를 파악하기 위해서입니다.

엔터프라이즈 아키텍처

환경에 기술 부채가 많을수록 마이그레이션이 더 어려워집니다. 건전한 엔터프라이즈 아키텍처 프로그램을 보유한 조직은 환경을 현재 및 최신 버전의 소프트웨어 및 시스템 (주요 릴리스의 N 및 N-1 버전이라고도 함) 으로 제한하려고 노력합니다. 이렇게 하면 고려해야 하는 시나리오 수가 줄어들 뿐만 아니라 최신 릴리스의 발전된 이점을 활용할 수 있습니다. 예를 들어 Windows Server 2012, Windows Server 2008 및 이전 버전의 Windows Server는 최신 버전에 비해 Windows Server 환경에서 자동화하기가 점점 더 어려워지고 있습니다. 이전 버전 및 지원되지 않는 버전의 경우 라이선싱도 더 어렵습니다.

표준화 및 구성 관리

환경의 표준화는 고려해야 할 또 다른 요소입니다. 수작업으로 구축하고 유지 관리하는 환경을 갖춘 조직은 애완 동물에 가까운 것으로 간주됩니다. 각 시스템은 고유하며 표준화된 이미지, 코드형 인프라 (IAC) 또는 지속적 통합 및 지속적 전달 (CI/CD) 파이프라인을 사용하여 구축한 경우보다 훨씬 더 많은 구성 조합이 가능합니다.

예를 들어 개별 서버를 수동으로 마이그레이션하는 대신 IAC 또는 CI/CD를 사용하여 일반적인 웹 서버를 재구축하는 것이 가장 좋습니다. 또한 모든 영구 데이터를 데이터베이스, 파일 공유 또는 리포지토리와 같은 데이터스토어에 저장하는 것이 가장 좋습니다. IaC 또는 CI/CD를 사용하여 시스템을 재구축하지 않는 경우 최소한 구성 관리 도구 (예: Puppet, Chef 또는 Ansible) 를 사용하여 보유한 서버를 표준화해야 합니다.

좋은 데이터

좋은 데이터는 성공적인 마이그레이션을 위한 핵심 요소이기도 합니다. 현재 서버 및 해당 메타데이터에 대한 정확한 데이터는 자동화 및 계획에 필수적입니다. 좋은 데이터가 부족하면 마이그레이션을 계획할 때 어려움이 가중됩니다. 좋은 데이터의 예로는 정확한 서버 인벤토리, 서버의 응용 프로그램, 버전이 있는 서버의 소프트웨어, CPU 수, 메모리 양 및 디스크 수 등이 있습니다. 웨이브 플래너가 계획에 필요한 데이터나 마이그레이션 프로세스 자동화의 일부로 사용할 데이터를 캡처하는 것이 좋습니다.

자동화

자동화는 대규모 마이그레이션에 필수적입니다. 자동화의 예로는 에이전트 설치, 자동화에 필요한 유틸리티의 소프트웨어 버전 업데이트 (예: .NET 또는 PowerShell, AWS 시스템 관리자 에이전트 (SSM 에이전트), Amazon과 같은 AWS용 소프트웨어 로드 또는 업데이트 CloudWatch AWS에서 실행하는 데 필요한 에이전트 또는 기타 백업 또는 관리 소프트웨어

세부 계획

대규모 마이그레이션을 위해서는 세부 계획을 개발하고 관리하는 것도 필수적입니다. 몇 주 동안 일주일에 50대의 서버를 마이그레이션하려면 잘 정의된 계획이 있어야 합니다. 효과적인 계획에는 다음이 포함됩니다.

- 용도웨이브 플래닝종속성과 우선 순위에 따라 서버를 웨이브로 구성할 수 있습니다.
- 용도주간 계획(컷오버까지 이어짐) 애플리케이션 팀과 통신하고 네트워크, DNS, 방화벽 및 컷오버 중에 해결해야 하는 기타 세부 정보를 식별합니다.
- 상세 사용, hour-to-hour계획(실제 컷오버 주변) 컷오버 유지 관리 기간을 설명합니다.
- 용도진행/불참 기준어떤 상황에서 애플리케이션이 AWS로 넘어간 것으로 간주되거나 소스 위치로 페일백되어야 하는지 설명합니다.
- 용도청소 활동반드시 완료해야 하는 후속 활동으로 이러한 활동은 컷오버 유지 관리 기간 외에 또는 완료 후에 발생할 수 있습니다.[하이퍼케어](#). 정리 작업에는 백업 및 다양한 에이전트 확인, 서버에서 Application Migration Service 에이전트 제거 또는 소스 서버 및 관련 리소스 제거 등이 포함됩니다.

동원

모바일 단계에서는 조직의 복잡성과 변이를 최대한 많이 발견하여 마이그레이션 계획 중에 고려할 수 있도록 하는 것이 중요합니다. 이상적으로는 컷오버 유지 관리 기간 동안 이러한 복잡성과 변형을 다루지 않고 장애 복구를 방지할 수 있습니다.

대규모 마이그레이션의 어려움

마이그레이션 실패는 애플리케이션 또는 애플리케이션이 새 환경으로 전환되어 마이그레이션 유지 관리 기간 내에 성능 또는 기능 요구 사항을 충족할 수 없을 때 발생합니다. 이로 인해 애플리케이션 또는 애플리케이션이 원래 위치로 페일백됩니다. 또한 해당 애플리케이션 또는 애플리케이션에 종속된 다른 모든 애플리케이션도 페일백해야 합니다. 애플리케이션 일정을 조정해야 하기 때문에 마이그레이션 실패는 현재의 파도뿐만 아니라 미래의 웨이브에도 영향을 미치는 경향이 있습니다.

지연 시간에 민감한 종속성

마이그레이션 실패의 주요 원인은 지연 시간에 민감한 종속성입니다. 지연 시간에 민감한 종속성을 식별하지 못하면 성능 문제가 발생하여 응답 시간이나 트랜잭션 시간이 지나치게 길어질 수 있습니다. 예를 들어, 일반적으로 애플리케이션은 데이터베이스와 애플리케이션 서버를 동시에 클라우드로 이동합니다. 서로 자주 통신하고 동일한 데이터 센터에 있을 때 1밀리초 미만의 응답 시간이 필요하기 때문입니다. 데이터베이스만 클라우드로 이동하면 이러한 트랜잭션에 몇 초의 지연 시간이 발생하여 애플리케이션 성능에 상당한 영향을 미칠 수 있습니다. 이는 서로 의존도가 높고 제대로 작동하려면 동일한 데이터 센터에 있어야 하는 애플리케이션에도 적용됩니다.

따라서 마이그레이션을 계획할 때는 애플리케이션 종속성을 이해하고 해결하는 것이 가장 중요합니다. 서로 의존하는 애플리케이션과 서비스를 식별해야 함께 마이그레이션할 수 있습니다.

IT 공유 서비스

워크로드가 클라우드로 이동한 후에는 적절하고 안전하게 작동하고 유지 관리하려면 다양한 서비스가 필요합니다. 여기에는 랜딩 존, 네트워크 및 보안 경계, 인증, 패치, 보안 스캐너, IT 서비스 관리 도구, 백업, 배스천 호스트 및 기타 리소스가 포함됩니다. 이러한 서비스가 없으면 워크로드가 제대로 작동하지 않을 수 있으며 원래 위치로 페일백될 수 밖에 없습니다.

구성 업데이트

대부분의 경우 워크로드가 클라우드로 이동된 후 워크로드가 제대로 작동하려면 몇 가지 구성을 변경해야 합니다. 이러한 구성 변경은 종종 다음과 같은 워크로드 종속성과 연관됩니다.

- 방화벽 규칙
- 허용 목록
- DNS 레코드
- 연결 문자열

구성을 제대로 업데이트하지 않으면 워크로드, 해당 사용자 및 종속 시스템이 서로 통신하지 못할 수 있습니다. 운영 중단 기간 내에 이러한 문제를 해결하는 것은 가능하지만, 현재 변경에는 시간이 많이 걸리거나 시간 내에 충족되지 않는 변경 기록이 필요할 수 있습니다.

애플리케이션 기능 테스트

대규모 마이그레이션의 또 다른 과제는 애플리케이션 기능 테스트의 필요성입니다. 많은 조직에서 지연 시간에 민감한 종속성, IT 공유 서비스 또는 필요한 구성 업데이트를 식별하기 위해 애플리케이션 팀에 의존하기 때문에 이는 특히 중요합니다. 이상적으로는 애플리케이션 팀이 컷오버 유지 관리 기간 동안 실행할 수 있는 서면 또는 자동화된 테스트 계획을 제공하여 애플리케이션이 적절한 성능으로 완벽하게 작동하는지 검증하는 것입니다. 컷오버 유지 관리 기간을 최소한으로 유지하려면 테스트를 30분 이내에 완료할 수 있어야 합니다.

애플리케이션 종속성 검색을 위한 도구

애플리케이션 간 종속성을 결정하는 것은 성공적인 마이그레이션을 위해 매우 중요하며, 지연 시간에 민감한 종속성과 연결 구성 항목을 감지하는 데 모두 중요합니다. 시장에는 다음과 같이 종속성을 검색하는 데 사용할 수 있는 몇 가지 도구가 있습니다. [애플리케이션 검색 서비스](#)(에이전트 및 에이전트리스 툴) 및 [클라우드마이즈](#)(에이전트 기반 도구)

애플리케이션 종속성 검색을 위한 도구를 선택할 때는 다음 사항을 고려하십시오.

- **소요 시간**— 알려진 피크, 월말 및 기타 이벤트와 같은 애플리케이션별 이벤트를 캡처할 수 있을 만큼 오랫동안 검색 도구를 실행하는 것이 좋습니다. 권장 최소 기간은 30일입니다.
- **액티브 (에이전트 기반)**— 액티브 종속성 검색 툴은 운영 체제의 커널에 내장되어 모든 트랜잭션을 캡처하는 경우가 많습니다. 그러나 이 방법은 일반적으로 가장 비용이 많이 들고 시간이 많이 걸리는 방법입니다.
- **패시브 (에이전트 없음)**— 패시브 종속성 검색 툴은 훨씬 저렴하고 빠르게 구현할 수 있지만 덜 사용되는 연결을 놓칠 위험이 있습니다.
- **제도적 지식**— 애플리케이션 검색 툴은 보다 상세하고 정확한 정보를 제공하지만 대부분의 조직에서는 애플리케이션 팀과 해당 기관의 지식을 활용하여 애플리케이션 상관 관계를 파악합니다. 애플리케이션 팀은 대기 시간에 민감한 종속성에 대해 잘 알고 있는 경우가 많지만 연결 구성 설정, 방화벽 규칙 또는 파트너의 허용 목록 요구 사항과 같은 일부 세부 정보를 놓치는 경우가 드물지 않습니다. 기관 지식을 활용하여 애플리케이션 종속성 검색을 개선할 수 있지만 관련된 위험도 고려하고 완화하는 것이 좋습니다. 예를 들어 애플리케이션 팀의 지식에만 의존한다면 연결 구성 항목이나 지연 시간에 민감한 종속성을 놓칠 위험이 있습니다. 이로 인해 중단이 발생하거나 마이그레이션이 실패할 수 있습니다. 이러한 위험을 줄이려면 상세한 애플리케이션 기능 테스트를 수행하는 것이 좋습니다.

Microsoft 워크로드 마이그레이션

이 섹션에서는 특정 Microsoft 워크로드에 대한 권장 가이드를 다룹니다. 다음의 모든 워크로드별 접근 방식은 평가, 동원, 마이그레이션 및 현대화 프레임워크를 준수합니다.

Active Directory 마이그레이션

Active Directory는 많은 기업 환경을 위한 일반적인 ID 및 액세스 관리 솔루션입니다. DNS, 사용자 및 시스템 관리가 결합되어 있는 Active Directory는 Microsoft 및 Linux 워크로드 모두에서 중앙 집중식 사용자 인증을 위한 이상적인 선택입니다. 클라우드나 AWS로의 여정을 계획할 때는 Active Directory를 AWS로 확장할지 아니면 관리형 서비스를 사용하여 디렉터리 서비스 인프라 관리 부담을 덜어줄지 선택해야 합니다. 조직에 적합한 접근 방식을 결정할 때는 각 옵션의 위험과 이점을 파악하는 것이 좋습니다.

Active Directory 마이그레이션을 위한 올바른 전략은 조직의 요구 사항에 맞고 AWS 클라우드를 활용할 수 있게 합니다. 여기에는 디렉터리 서비스 자체뿐만 아니라 다른 AWS 서비스와 상호 작용하는 방식도 고려하는 작업이 포함됩니다. 또한 Active Directory를 관리하는 팀의 장기적인 목표를 고려해야 합니다.

Active Directory 마이그레이션 외에도 Active Directory가 위치할 계정 구조, AWS 계정의 네트워크 토폴로지, Active Directory가 필요하며 사용할 계획인 DNS 통합 및 기타 잠재적 AWS 서비스를 결정해야 합니다. 계정 토폴로지 설계 및 기타 마이그레이션 전략 고려 사항에 대한 자세한 내용은 이 가이드의 [기본 모범 사례](#) 섹션을 참조하세요.

평가

마이그레이션을 성공적으로 구현하려면 기존 인프라를 평가하고 환경에 필요한 주요 기능을 이해하는 것이 중요합니다. 마이그레이션 방법을 선택하기 전에 다음 영역을 검토하는 것이 좋습니다.

- 기존 AWS 인프라 설계 검토 - 이 가이드의 [Windows 환경 검색](#) 섹션에 나와 있는 지침에 따라 기존 Active Directory 인프라의 설치 공간 및 인프라 요구 사항을 아직 모르는 경우 평가 방법을 사용하여 검토할 수 있습니다. AWS의 Active Directory 인프라에 대해 Microsoft에서 규정한 크기 조정을 사용하는 것이 좋습니다. Active Directory 인프라를 AWS로 확장하는 경우 AWS에서 Active Directory 인증 공간의 일부만 필요할 수 있습니다. 따라서 Active Directory 공간을 AWS로 완전히 옮기지 않는 한 환경을 과도하게 확장하지 않습니다. 자세한 내용은 Microsoft 설명서의 [Active Directory 도메인 서비스에 대한 용량 계획](#)을 참조하세요.
- 기존 온프레미스 Active Directory 디자인 검토 - 온프레미스(자체 관리형) Active Directory의 현재 사용률을 검토합니다. Active Directory 환경을 AWS로 확장하는 경우 온프레미스 환경에 대한 확

장으로도 AWS의 여러 도메인 컨트롤러에서 Active Directory를 실행하는 것이 좋습니다. 이는 여러 가용 영역에 인스턴스를 배포하여 잠재적인 장애에 대비하도록 설계하는 [AWS Well-Architected Framework](#)를 준수합니다.

- 애플리케이션 및 네트워크의 종속성 파악 - 가장 적합한 마이그레이션 전략을 선택하기 전에 조직에서 기능에 필요한 Active Directory의 모든 기능을 완전히 이해해야 합니다. 즉, 관리형 서비스와 셀프 호스팅 중에서 선택할 때는 각각의 옵션을 이해하는 것이 중요합니다. 어떤 마이그레이션이 적합한지 결정할 때 다음 사항을 고려하세요.
 - 액세스 요구 사항 - Active Directory를 제어하기 위한 액세스 요구 사항에 따라 적합한 마이그레이션 경로가 지정됩니다. 규정 준수를 위해 모든 유형의 에이전트를 설치하기 위해 Active Directory 도메인 컨트롤러에 대한 전체 액세스 권한이 필요한 경우 AWS Managed Microsoft AD는 적합한 솔루션이 아닐 수 있습니다. 대신 도메인 컨트롤러에서 AWS 계정 내 Amazon EC2로 Active Directory의 확장을 조사합니다.
 - 마이그레이션 일정 - 완료 날짜가 명확하지 않은 연장된 마이그레이션 타임라인이 있는 경우 클라우드 및 온프레미스 환경에서 인스턴스 관리를 위한 대책이 마련되어 있는지 확인합니다. 인증은 Microsoft 워크로드에서 관리 문제를 방지하기 위해 갖추어야 할 핵심 구성 요소입니다. 마이그레이션 초기에 Active Directory 이동 계획을 세우는 것이 좋습니다.
- 백업 전략 - Active Directory 도메인 컨트롤러의 시스템 상태를 캡처하기 위해 기존 Windows 백업을 사용하는 경우 AWS에서 기존 백업 전략을 계속 사용할 수 있습니다. 또한 AWS는 인스턴스를 백업하는 데 도움이 되는 기술 옵션을 제공합니다. 예를 들어, [AWS Data Lifecycle Manager](#), [AWS Backup](#) 및 [AWS Elastic Disaster Recovery](#)는 Active Directory 도메인 컨트롤러 백업을 위해 지원되는 기술입니다. 문제를 방지하려면 Active Directory의 복원에 의존하지 않는 것이 가장 좋습니다. 권장되는 모범 사례는 복원력이 뛰어난 아키텍처를 구축하는 것이지만 복구가 필요한 경우 백업 방법을 마련하는 것이 중요합니다.
- 재해 복구(DR) 요구 사항 - AWS로 Active Directory를 마이그레이션하는 경우 재해 발생 시 복원력을 갖추도록 설계해야 합니다. AWS로 기존 Active Directory를 이동하는 경우 보조 AWS 리전을 사용하고 Transit Gateway를 사용하여 두 리전을 연결하여 복제가 이루어지도록 할 수 있습니다. 이것이 일반적으로 선호되는 방법입니다. 일부 조직은 신뢰성을 테스트하기 위해 며칠 동안 기본 사이트와 보조 사이트 간의 연결을 제공하는 격리된 환경에서 장애 조치 테스트에 대한 다양한 요구 사항을 가지고 있습니다. 이것이 조직의 요구 사항인 경우 Active Directory에서 브레인 분할 문제를 정리하는 데 시간이 걸릴 수 있습니다. DR 사이트를 장애 조치 환경으로 남겨두고 DR 전략을 격리하여 정기적으로 테스트해야 하는 액티브/패시브 구현으로 [AWS Elastic Disaster Recovery](#)를 사용할 수 있습니다. 조직의 Recovery Time Objective(RTO) 및 Recovery Point Objective(RPO) 요구 사항을 계획하는 것은 AWS로의 마이그레이션을 평가할 때 중요한 요소입니다. 구현을 검증하기 위한 테스트 및 장애 조치 계획과 함께 요구 사항을 정의했는지 확인합니다.

동원

조직 및 운영 요구 사항을 충족하는 적절한 전략은 AWS로 Active Directory를 마이그레이션하거나 확장하는 데 있어 중요한 요소입니다. AWS 서비스와 통합할 방법을 선택하는 것은 AWS를 도입하는 데 있어 매우 중요합니다. 비즈니스 요구 사항에 맞는 Active Directory 또는 AWS Managed Microsoft AD의 메서드 확장을 선택해야 합니다. Amazon RDS와 같은 서비스에는 AWS Managed Microsoft AD 사용에 의존하는 일부 기능이 있습니다. AWS 서비스 제한을 평가하여 Amazon EC2 및 AWS Managed Microsoft AD의 Active Directory에 대한 호환성 제약이 있는지 확인해야 합니다. 계획 프로세스의 일부로 다음과 같은 통합 지점을 고려하는 것이 좋습니다.

AWS에서 Active Directory를 사용하는 이유는 다음과 같습니다.

- AWS 애플리케이션이 Active Directory와 연동되도록 지원
- Active Directory를 사용하여 AWS Management Console에 로그인

AWS 애플리케이션이 Active Directory와 연동되도록 지원

[AWS 클라이언트 VPN](#), [AWS 관리 콘솔](#), [AWS IAM ID 센터 \(AWSSingle Sign-On의 후속\)](#), [아마존 차임](#), [아마존 커넥트](#), [윈도우 파일 서버용 아마존 FSx](#), [아마존](#), [SQL Server용 아마존 RDS \(디렉터리 서비스에만 해당\)](#) 등 여러 AWS 애플리케이션 및 서비스를 활성화할 수 있습니다. (AWS Managed), [QuickSight 아마존](#), [아마존 \(Amazon\)](#), [아마존 \(Amazon WorkDocs WorkMail\)](#) 에서 [AWS 매니지드 서비스를 이용하세요 WorkSpaces](#) 마이크로소프트 AD 디렉터리. 디렉터리에서 AWS 애플리케이션 또는 서비스를 활성화하면 사용자는 Active Directory 보안 인증을 사용하여 애플리케이션 또는 서비스에 액세스할 수 있습니다. 익숙한 Active Directory 관리 도구를 사용하면 Active Directory 그룹 정책 객체 (GPO)를 적용하여 인스턴스를 [AWS Managed Microsoft AD 디렉터리](#)에 조인함으로써 Windows 또는 Linux용 Amazon EC2 인스턴스를 중앙에서 관리할 수 있습니다.

사용자는 자신의 Active Directory 보안 인증으로 인스턴스에 로그인할 수 있습니다. 따라서 개별 인스턴스 자격 증명을 사용하거나 프라이빗 키(PEM) 파일을 배포할 필요가 없습니다. 그로 인해 이미 사용 중인 Active Directory 사용자 관리 도구를 사용하여 사용자에게 액세스 권한을 즉시 부여하거나 취소하는 일이 더 간편해집니다.

Active Directory를 사용하여 AWS Management Console에 로그인

AWS Managed Microsoft AD를 사용하면 디렉터리 구성원에게 AWS Management Console에 대한 액세스 권한을 부여할 수 있습니다. 기본적으로 디렉터리 멤버는 AWS 리소스에 액세스할 수 있는 권한이 없습니다. 디렉터리 멤버에게 AWS Identity and Access Management(IAM) 역할을 할당하여 다양한

AWS 서비스 및 리소스에 대한 액세스 권한을 제공합니다. IAM 역할은 디렉터리 멤버가 보유하고 있는 서비스, 리소스, 액세스 수준을 정의합니다.

예를 들면, 사용자가 자신의 [Active Directory 보안 인증](#)을 사용해 AWS Management Console에 로그인하도록 활성화할 수 있습니다. 이렇게 하려면 디렉터리에서 AWS Management Console을 애플리케이션으로 사용할 수 있게 활성화한 다음 Active Directory 사용자와 그룹을 IAM 역할에 할당해야 합니다. 사용자는 AWS Management Console에 로그인할 때 AWS 리소스를 관리할 수 있는 IAM 역할을 갖습니다. 이로써 별도의 SAML 인프라를 구성 및 관리하지 않고도 사용자에게 AWS Management Console에 액세스 권한을 손쉽게 부여할 수 있습니다. 자세한 내용은 AWS Security Blog의 [How AWS IAM Identity Center Active Directory sync enhances AWS application experiences](#)를 참조하세요. 디렉터리 또는 온프레미스 Active Directory의 사용자 계정에 액세스 권한을 부여할 수 있습니다. 이를 통해 사용자는 기존 사용자 계정에 직접 IAM 역할을 할당하여 기존 보안 인증과 권한으로 AWS 리소스를 관리하여 AWS Management Console 또는 AWS Command Line Interface(AWS CLI)를 통해 로그인할 수 있습니다.

디렉터리 멤버에게 콘솔 액세스 권한을 부여할 수 있으려면 디렉터리가 액세스 URL을 가지고 있어야 합니다. 디렉터리 세부 정보를 보고 액세스 URL을 얻는 방법에 대한 자세한 내용은 AWS Directory Service 관리 안내서의 [View directory information](#)을 참조하세요. 액세스 URL을 생성하는 방법에 대한 자세한 내용은 AWS Directory Service 관리 안내서의 [Creating an access URL](#)을 참조하세요. IAM 역할을 생성하고 디렉터리 구성원에게 할당하는 방법에 대한 자세한 내용은 AWS Directory Service 관리 안내서의 [Grant users and groups access to AWS resources](#)를 참조하세요.

Active Directory에 대한 다음 마이그레이션 옵션을 고려하세요.

- Active Directory 확장
- AWS Managed Microsoft AD로 마이그레이션
- 신뢰 관계를 사용하여 AWS Managed Microsoft AD와 Active Directory 연결
- Amazon Route 53와 Active Directory DNS 통합

Active Directory 확장

Active Directory 인프라가 이미 있고 이를 사용해 Active Directory 인식 워크로드를 AWS 클라우드로 마이그레이션하고 싶다면 AWS Managed Microsoft AD가 도움이 될 수 있습니다. [신뢰 관계](#)를 사용하여 AWS Managed Microsoft AD를 기존 Active Directory에 연결할 수 있습니다. 이로써 사용자, 그룹 또는 암호를 동기화하지 않고도 사용자는 온프레미스 Active Directory 보안 인증을 사용해 Active Directory 인식 및 AWS 애플리케이션에 액세스할 수 있습니다. 예를 들어, 사용자는 기존 Active Directory 사용자 이름과 암호를 사용하여 AWS 관리 콘솔에 로그인할 수 있습니다. WorkSpaces 또한

SharePoint AWS 관리형 Microsoft AD와 같은 Active Directory 인식 애플리케이션을 사용하는 경우 로그인한 Windows 사용자는 자격 증명을 다시 입력할 필요 없이 이러한 애플리케이션에 액세스할 수 있습니다.

신뢰를 사용하는 것 외에도 AWS의 EC2 인스턴스에서 실행되도록 Active Directory를 배포하여 Active Directory를 확장할 수 있습니다. 직접 수행하거나 [AWS와 협력](#)하여 프로세스를 도울 수 있습니다. Active Directory를 AWS로 확장할 때는 서로 다른 가용 영역에 2개 이상의 도메인 컨트롤러를 배포하는 것이 좋습니다. AWS에 있는 사용자 및 컴퓨터 수에 따라 2개 이상의 도메인 컨트롤러를 배포해야 할 수도 있지만 복원성을 위해 권장하는 최소 수는 2개입니다. 또한 [Active Directory Migration Toolkit\(ADMT\)](#) 및 [Password Export Server\(PES\)](#)를 사용하여 온프레미스 Active Directory 도메인을 AWS로 마이그레이션하여 Active Directory 인프라의 운영 부담 없이 마이그레이션을 수행할 수 있습니다. [Active Directory Launch Wizard](#)를 사용하여 AWS에 Active Directory를 배포할 수도 있습니다.

AWS Managed Microsoft AD로 마이그레이션

AWS에서 Active Directory를 사용하기 위해 두 가지 메커니즘을 적용할 수 있습니다. 한 가지 방법은 AWS Managed Microsoft AD를 도입하여 Active Directory 객체를 AWS로 마이그레이션하는 것입니다. 여기에는 사용자, 컴퓨터, 그룹 정책 등이 포함됩니다. 두 번째 메커니즘은 모든 사용자와 객체를 내보낸 다음, [Active Directory Migration Tool](#)을 사용하여 사용자와 객체를 수동으로 가져오는 수동 접근 방식입니다.

AWS Managed Microsoft Active Directory로 이동해야 하는 또 다른 이유가 있습니다.

- [AWS Managed Microsoft AD는 Microsoft 원격 데스크톱 라이선싱 관리자, Microsoft SharePoint 및 Microsoft SQL Server Always On과 같은 기존의 액티브 디렉터리 인식 워크로드를 AWS 클라우드에서 실행할 수 있는 실제 Microsoft Active Directory 도메인입니다.](#)
- AWS Managed Microsoft AD는 그룹 관리형 서비스 계정(gMSA) 및 Kerberos 제한 위임(KCD)을 사용하여 Active Directory 통합 .NET 애플리케이션의 보안을 단순화하고 개선하는 데 도움이 됩니다. 자세한 내용은 AWS 설명서의 [Simplify Migration and Improve Security of Active Directory-Integrated .NET Applications by Using AWS Microsoft AD](#)를 참조하세요.

여러 AWS 계정에서 AWS Managed Microsoft AD를 공유할 수 있습니다. 이를 통해 각 계정 및 Amazon Virtual Private Cloud(VPC)에 대한 디렉터리를 운영할 필요 없이 [Amazon EC2](#)와 같은 AWS 서비스를 관리할 수 있습니다. AWS 리전 내 모든 [Amazon VPC](#)와 모든 AWS 계정의 디렉터리를 사용할 수 있습니다. 모든 계정과 VPC에서 단일 디렉터리로 디렉터리 인식 워크로드를 더 쉽게, 그리고 더 비용 효과적으로 관리할 수 있는 기능입니다. 예를 들어, 이제 단일 AWS Managed Microsoft AD 디렉터리를 사용하여 여러 계정과 Amazon VPC에 걸쳐 EC2 인스턴스에 배포된 [Microsoft 워크로드](#)를 쉽게 관리할 수 있습니다. AWS Managed Microsoft AD 디렉터리를 다른 AWS 계정과 공유하는 경우,

Amazon EC2 콘솔이나 [AWS Systems Manager](#)를 사용하여 자신의 인스턴스를 계정과 AWS 리전 내 모든 Amazon VPC 인스턴스에 원활하게 조인할 수 있습니다.

또한 수동으로 인스턴스를 도메인에 조인하거나 각 계정과 Amazon VPC의 디렉터리에 배포할 필요 없이, 신속하게 디렉터리 인식 워크로드를 EC2 인스턴스에 배포할 수 있습니다. 자세한 내용은 AWS Directory Service 관리 안내서의 [Share your directory](#)를 참조하세요. AWS Managed Microsoft AD 환경을 공유하려면 비용이 발생한다는 점에 유의하세요. Amazon VPC 피어 또는 Transit Gateway 피어를 사용하여 다른 네트워크 또는 계정에서 AWS Managed Microsoft AD 환경과 통신할 수 있으므로 공유가 필요하지 않을 수 있습니다. 디렉터리를 Amazon Aurora MySQL, Amazon Aurora PostgreSQL, Amazon FSx, Amazon RDS for MariaDB, Amazon RDS for MySQL, Amazon RDS for Oracle, Amazon RDS for PostgreSQL 및 Amazon RDS for SQL Server와 함께 사용하려는 경우 도메인을 공유해야 합니다.

AWS Managed Microsoft AD에 신뢰 사용

기존 디렉터리의 사용자에게 AWS 리소스에 대한 액세스 권한을 부여하려면 AWS Managed Microsoft AD 구현에 신뢰를 사용합니다. AWS Managed Microsoft AD 환경 간에 신뢰를 생성하는 것도 가능합니다. 자세한 내용은 AWS Security Blog의 [Everything you wanted to know about trusts with AWS Managed Microsoft AD](#) 게시물을 참조하세요.

Amazon Route 53와 Active Directory DNS 통합

AWS로 마이그레이션할 때 Route 53 Resolver를 사용하여 DNS 이름으로 서버에 대한 액세스를 허용하는 방식으로 환경에 DNS를 통합할 수 있습니다. DHCP 옵션 세트를 수정하는 대신 Route 53 Resolver 엔드포인트를 사용하여 이 작업을 수행하는 것이 좋습니다. 이는 DHCP 옵션 세트를 수정하는 것보다 DNS 구성을 관리하기 위한 보다 중앙화된 접근 방식입니다. 또한 다양한 해석기 규칙을 활용할 수 있습니다. 자세한 내용은 Networking & Content Delivery Blog의 [Integrating your Directory Service's DNS resolution with Amazon Route 53 Resolvers](#) 게시물과 AWS 권장 가이드 설명서의 [Set up DNS resolution for hybrid networks in a multi-account AWS environment](#)를 참조하세요.

마이그레이션

AWS로 마이그레이션을 시작할 때 마이그레이션에 도움이 되는 구성 및 도구 옵션을 고려해 보는 것이 좋습니다. 환경의 장기적인 보안 및 운영 측면을 고려하는 것도 중요합니다.

다음 옵션을 고려해보세요.

- 클라우드 네이티브 보안
- AWS로 Active Directory를 마이그레이션하는 도구

클라우드 네이티브 보안

- Active Directory 컨트롤러에 대한 보안 그룹 구성 - AWS Managed Microsoft AD를 사용하는 경우 도메인 컨트롤러에는 도메인 컨트롤러에 대한 액세스를 제한하기 위한 VPC 보안 구성이 함께 제공됩니다. 일부 잠재적 사용 사례에 대한 액세스를 허용하도록 보안 그룹 규칙을 수정해야 할 수도 있습니다. 보안 그룹 구성에 대한 자세한 내용은 AWS Directory Service 관리 안내서의 [Enhance your AWS Managed Microsoft AD network security configuration](#)을 참조하세요. 사용자가 이러한 그룹을 수정하거나 다른 AWS 서비스에 사용하는 것을 허용하지 않는 것이 좋습니다. 다른 사용자가 이러한 그룹을 사용하도록 허용하면 사용자가 필요한 통신을 차단하도록 Active Directory 환경을 수정하는 경우 Active Directory 환경에 대한 서비스가 중단될 수 있습니다.
 - Active Directory 이벤트 CloudWatch 로그용 Amazon Logs와 통합 — AWS 관리형 Microsoft AD를 실행하거나 자체 관리형 Active Directory를 사용하는 경우 Amazon Logs를 활용하여 Active Directory CloudWatch 로깅을 중앙 집중화할 수 있습니다. CloudWatch 로그를 사용하여 인증, 보안 및 기타 로그를 에 복사할 수 있습니다. CloudWatch 이렇게 하면 한 곳에서 로그를 쉽게 검색할 수 있으며 일부 규정 준수 요구 사항을 충족하는 데 도움이 될 수 있습니다. CloudWatch Logs와 통합하면 사용자 환경에서 발생할 향후 사고에 더 잘 대응할 수 있으므로 통합하는 것이 좋습니다. 자세한 내용은 AWS 디렉토리 서비스 관리 안내서의 [AWS 관리형 Active Directory용 Amazon CloudWatch Logs 활성화 및 AWS 지식 센터의 Windows 이벤트 로그용 Amazon Logs 사용](#)을 참조하십시오.
- CloudWatch

AWS로 Active Directory를 마이그레이션하는 도구

마이그레이션을 수행하려면 Active Directory Migration Tool(ADMT)과 Password Export Server(PES)를 사용하는 것이 좋습니다. 이렇게 하면 사용자와 컴퓨터를 도메인 간에 쉽게 이동할 수 있습니다. PES를 사용하거나 관리형 Active Directory 도메인 간에 마이그레이션하는 경우 다음 고려 사항에 유의하세요.

- 사용자, 그룹 및 컴퓨터를 위한 Active Directory Migration Tool(ADMT) - [ADMT](#)를 사용하여 자체 관리형 Active Directory에서 AWS Managed Microsoft AD로 사용자를 마이그레이션할 수 있습니다. 중요한 고려 사항은 마이그레이션 타임라인과 보안 식별자(SID) 기록의 중요성입니다. 마이그레이션 중 SID 기록이 이전되지 않습니다. SID 기록 지원이 꼭 필요한 경우 SID 기록을 유지 관리할 수 있도록 ADMT 대신 Amazon EC2에서 자체 관리형 Active Directory를 사용하는 것이 좋습니다.
- Password Export Server(PES) - PES를 사용하여 암호를 AWS Managed Microsoft AD 내부로 마이그레이션할 수 있지만 외부로 마이그레이션할 수는 없습니다. 디렉터리에서 사용자와 암호를 마이그레이션하는 방법에 대한 자세한 내용은 AWS Security Blog의 [How to migrate your on-premises domain to AWS Managed Microsoft AD using ADMT](#)와 Microsoft 설명서의 [PES v3.1 \(x64\)](#)을 참조하세요.

- LDIF - LDAP Data Interchange Format(LDIF)은 AWS Managed Microsoft AD Directory 디렉터리의 스키마를 확장하는 데 사용되는 파일 형식입니다. LDIF 파일에는 디렉터리에 새 객체와 속성을 추가하는 데 필요한 정보가 들어 있습니다. 파일은 구문에 대한 LDAP 표준을 충족해야 하며 파일이 추가하는 각 객체에 대한 유효한 개체 정의를 포함해야 합니다. LDIF 파일을 생성한 후에는 파일을 디렉터리에 업로드하여 해당 스키마를 확장해야 합니다. LDIF 파일을 사용하여 AWS Managed Microsoft AD 디렉터리의 스키마를 확장하는 방법에 대한 자세한 내용은 AWS Directory Service 관리 안내서의 [Extending the schema of AWS Managed AD](#)를 참조하세요.
- CSVDE - 트러스트를 생성하고 ADMT를 사용하지 않고 디렉터리로 사용자를 내보내고 가져와야 하는 경우도 있습니다. 이상적이지는 않지만 [Csvde](#)(명령줄 도구)를 사용하여 도메인 간에 Active Directory 사용자를 마이그레이션할 수 있습니다. Csvde를 사용하려면 사용자 이름, 암호 및 그룹 멤버십과 같은 사용자 정보가 포함된 CSV 파일을 생성해야 합니다. 그런 다음 csvde 명령을 사용하여 사용자를 새 도메인으로 가져올 수 있습니다. 이 명령을 사용하여 소스 도메인에서 기존 사용자를 내보낼 수도 있습니다. 이는 SAMBA Domain Services 등의 다른 디렉터리 소스에서 Microsoft Active Directory로 마이그레이션하는 경우 유용할 수 있습니다. 자세한 내용은 AWS Security Blog의 [How to Migrate Your Microsoft Active Directory Users to Simple AD or AWS Managed Microsoft AD](#)를 참조하세요.

추가적인 리소스

- [Everything you wanted to know about trusts with AWS Managed Microsoft AD](#)(AWS Security Blog)
- [How to migrate your on-premises domain to AWS Managed Microsoft AD using ADMT](#)(AWS Security Blog)
- [AWS 이머전 데이의 액티브 디렉터리](#) (AWS 워크숍 스튜디오)

Windows Server 마이그레이션

이 섹션에서는 AWS로 Windows Server를 마이그레이션하는 데 사용할 수 있는 다양한 옵션에 중점을 둡니다.

평가

먼저 AWS로 마이그레이션해야 하는 애플리케이션과 워크로드를 식별합니다. [AWS Application Discovery Service](#)를 사용하여 온프레미스 인프라 및 애플리케이션 간 종속성의 맵을 생성할 수 있습니다. 이를 통해 AWS로 마이그레이션하는 데 필요한 서버, 애플리케이션 및 서비스를 식별할 수 있습니다.

[AWS Migration Hub](#)를 사용하여 애플리케이션 인벤토리를 생성하고 AWS와의 호환성을 평가할 수 있습니다. Migration Hub는 애플리케이션 포트폴리오의 중앙 집중식 뷰를 제공하고 마이그레이션 프로젝트를 계획, 추적 및 관리하는 데 도움이 됩니다. Cloudamize 또는 Evolve와 같이 AWS를 지원하는 타사 평가 도구를 사용할 수도 있습니다.

동원

대규모 인프라를 리호스팅(리프트 앤드 시프트)하기 위한 올바른 경로를 찾는 것은 중요한 과제일 수 있습니다. 유용한 [모범 사례](#)는 많지만 도구 선택은 워크로드 유형, 적절한 가동 중지 시간, 운영 체제 요구 사항 등 여러 요인에 따라 달라집니다. [AWS Application Migration Service](#)를 사용하여 리호스팅하는 것이 좋습니다.

AWS Application Migration Service

Application Migration Service를 사용하여 호환성 문제, 성능 영향 또는 긴 전환 기간 없이 빠르게 물리적, 가상 또는 클라우드 서버를 리프트 앤드 시프트할 수 있습니다. 애플리케이션 마이그레이션 서비스는 소스 서버를 AWS 계정에 지속적으로 복제합니다. 그런 다음 마이그레이션할 준비가 되면 Application Migration Service가 가동 중지 시간을 최소화하면서 AWS에서 서버를 자동으로 변환하고 시작합니다. 자세한 내용은 [What Is AWS Application Migration Service?](#)를 참조하세요. Application Migration Service 사용 설명서

AWS Migration Hub Orchestrator

[AWS Migration Hub Orchestrator](#)는 Application Migration Service를 사용하여 AWS로 서버 및 엔터프라이즈 애플리케이션 마이그레이션하는 작업을 간소화하고 자동화합니다. 또한 마이그레이션을 실행하고 추적할 수 있는 단일 위치를 제공합니다. Migration Hub Orchestrator를 사용하여 S/4HANA, BW/4HANA, SAP ECC on HANA 등과 같은 SAP NetWeaver 기반 애플리케이션을 AWS로 마이그레이션하고 지원되는 사용자 지정 애플리케이션을 Amazon EC2에 재호스팅할 수 있습니다. Migration Hub Orchestrator는 고유한 마이그레이션 요구 사항에 맞게 사용자 지정할 수 있는 마이그레이션 워크플로를 생성할 수 있는 템플릿을 제공합니다. 또한 Migration Hub Orchestrator는 선택한 워크플로의 단계를 자동화하고 마이그레이션 상태를 표시합니다.

VM Import

[AWS VM Import](#)를 사용하면 VM 이미지를 기존 가상화 환경에서 Amazon EC2로 가져온 다음, 다시 내보낼 수 있습니다. 이 경우 애플리케이션과 워크로드를 Amazon EC2로 마이그레이션하고, VM 이미지 카탈로그를 Amazon EC2로 복사하거나 백업 및 재해 복구를 위해 VM 이미지 리포지토리를 만들 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [VM Import/Export란 무엇입니까?](#)를 참조하세요.

마이그레이션할 워크로드를 평가한 후 마이그레이션 전략, 타임라인 및 마이그레이션 프로세스 관련 비용을 약속한 마이그레이션 계획을 세웁니다. [AWS 요금/TCO 도구](#)를 사용하여 AWS에서 애플리케이션을 실행할 때 절감되는 비용을 추정할 수 있습니다. 또한 [Application Discovery Service](#)를 사용하여 마이그레이션된 워크로드를 호스팅하는 데 적합한 AWS 서비스를 식별할 수 있습니다.

마이그레이션

AWS로 Windows 워크로드를 마이그레이션하려면 마이그레이션 계획, 준비 상태 평가 및 마이그레이션 구현 단계를 비롯한 여러 단계가 포함됩니다. 마이그레이션 단계는 AWS로 Windows 워크로드를 마이그레이션하는 마지막 단계입니다. 다음은 마이그레이션 단계에서 고려해야 할 몇 가지 단계입니다.

- AWS 환경 준비 - 마이그레이션 프로세스를 시작하기 전에 Amazon Machine Image(AMI)를 생성하고 워크로드를 마이그레이션할 VPC를 설정하여 AWS 환경을 준비해야 합니다.
- 마이그레이션 도구 선택 - Migration Hub, Application Migration Service, VM Import 등의 다양한 마이그레이션 방법 중에서 선택할 수 있습니다. 필요에 맞는 방법을 선택하세요.
- 마이그레이션 구성 - 소스 서버를 선택하고 대상 인스턴스 유형, 스토리지 및 네트워크 설정을 지정하여 마이그레이션을 구성합니다.
- 마이그레이션 수행 - 구성이 완료된 후 마이그레이션을 수행합니다. 이 프로세스에는 데이터 복제, 마이그레이션된 워크로드 테스트, 마이그레이션된 워크로드로 전환하기 위한 최종 전환 수행이 포함됩니다. 위에서 선택한 마이그레이션 도구가 이러한 단계를 안내합니다.
- 마이그레이션 검증 - 마이그레이션이 완료된 후 마이그레이션된 워크로드가 예상대로 검증하는지 확인합니다. 테스트를 수행하고 보안 및 규정 준수 요구 사항이 충족되는지 확인합니다.
- 마이그레이션된 워크로드 최적화 - 인스턴스 크기를 조정하고, 자동 크기 조정을 구성하고, 예약 인스턴스, 스팟 인스턴스 등의 비용 절감 전략을 구현하여 마이그레이션된 워크로드를 최적화합니다.
- 마이그레이션된 워크로드 모니터링 및 관리 - 마이그레이션된 워크로드를 지속적으로 모니터링하고 관리하여 최적의 성능과 보안을 보장합니다. [CloudWatchAmazon](#)을 사용하여 모니터링할 수 있습니다.

파일 서버 마이그레이션

스토리지는 실행하는 모든 워크로드의 필수 구성 요소입니다. AWS에는 블록, 파일 및 객체 스토리지를 포함하여 클라우드에 파일을 저장할 수 있는 다양한 옵션이 있습니다. Microsoft 워크로드의 경우 가장 일반적인 옵션은 블록 및 파일 스토리지 옵션입니다. 이 섹션에서는 Microsoft 워크로드용 스토리지를 AWS 클라우드로 마이그레이션하는 데 도움이 되는 전략을 제공하고 파일 서버 마이그레이션 과정을 안내합니다.

평가

주요 스토리지 유형으로 객체, 블록, 파일 스토리지의 3가지가 있습니다. AWS는 각 항목으로 분류할 수 있는 광범위한 스토리지 서비스 포트폴리오를 제공합니다. 성공적인 마이그레이션은 현재 요구 사항을 이해한 다음, 이를 다양한 AWS 스토리지 서비스와 [비교](#)하여 자신에게 가장 적합한 것이 무엇인지 판단하는 데 달려 있습니다. 워크로드에 적합한 기술을 선택하는 것이 장기적인 성공의 열쇠입니다. 스토리지에 현재 사용하는 것과 정확히 일치시키려고 하지 않는 것이 좋습니다. 대신 사용 가능한 모든 옵션을 살펴보고 Microsoft 워크로드의 비용 및 성능을 최적화하는 데 가장 적합한 옵션을 선택하는 것이 좋습니다. 로컬 블록 스토리지가 필요한 대규모 온프레미스 파일 서버를 예로 들 수 있습니다. AWS에서 최적의 선택은 [Amazon FSx](#)로 이동하여 파일 서버와 동일한 성능을 얻으면서 획일적이고 과중한 파일 서버 및 백엔드 스토리지 관리 작업을 없애는 것입니다.

TCO는 어떤 스토리지 옵션이 가장 적합한지 판단할 때 평가해야 할 핵심 항목입니다. AWS 관리형 서비스를 사용하면 운영 비용을 절감할 수 있으므로 AWS에서 올바른 전체 스토리지 솔루션을 선택하는 데 도움이 됩니다. 스토리지 평가를 요청하려면 migration-evaluator@amazon.com으로 문의하세요. 스토리지 전문가가 워크로드 평가하고, 가장 적합한 AWS 스토리지 서비스에 워크로드 매핑, 방향별 예상 비용 제공을 도와드립니다. 스토리지 평가에는 세 단계가 있습니다.

- 에이전트 없는 수집기를 설치하거나 기존 도구 세트의 출력을 플랫폼 파일로 수신하여 검색 프로세스를 시작합니다.
- 7~60일 동안 검색 프로세스가 실행되게 둡니다.
- 스토리지 수집기는 검색 도구의 데이터를 분석한 다음, 대상 스토리지 솔루션을 제안하고 솔루션에 대한 방향별 예상 비용을 제공합니다.

스토리지 옵션의 비용이 약간 더 높은 경우 해당 스토리지 옵션이 장기적으로 전체 비용을 줄이는지 고려하고 스토리지의 보안과 신뢰성을 유지하기 위해 팀이 수행해야 할 작업을 알아봅니다. 이는 워크로드에 적합한 장기 솔루션일 수 있습니다.

적합한 솔루션을 평가할 때는 성능과 비용을 살펴보는 것이 중요합니다. [Windows Performance Monitor](#)와 같은 도구를 사용하여 워크로드의 IOPS, 처리량 및 기타 성능 요구 사항을 식별한 다음 워크로드에 대해 선택한 AWS 솔루션에서 동일한 테스트를 실행할 수 있습니다. 또한 CloudWatch 에이전트를 사용하여 [Windows 서버에서 Performance Monitor의 지표를 보고 해당 워크로드를 프로덕션에 적용하기 전에 워크로드의 지표를 분석할 수 있습니다.](#)

사용자의 요구 사항에 가장 잘 맞는 AWS 스토리지 서비스 식별

스토리지 서비스의 선택은 일반적으로 사용 사례, 애플리케이션 요구 사항, 친숙도, 성능 프로파일 및 데이터 관리 기능에 따라 달라집니다. 다음을 고려하세요.

- Amazon Simple Storage Service(S3) – [Amazon S3](#)는 어디서든 원하는 양의 데이터를 저장하고 검색할 수 있도록 구축된 객체 스토리지입니다. Amazon S3는 워크로드의 데이터 액세스, 복원력 및 비용 요구 사항에 따라 선택할 수 있는 다양한 스토리지 클래스를 제공합니다. [AWS Storage Gateway](#)를 사용하여 Amazon S3에 대한 파일 기반 액세스를 구현할 수 있습니다. 이를 통해 SMB(Server Message Block)를 사용하는 애플리케이션을 완전히 다시 작성할 필요 없이 Amazon S3의 저렴한 스토리지를 활용할 수 있습니다.
- Amazon Elastic Block Store(Amazon EBS) – [Amazon EBS](#)는 EC2 인스턴스에 사용할 수 있는 블록 수준 스토리지 볼륨을 제공합니다. EBS 볼륨은 형식이 지정되지 않은 원시 블록 디바이스처럼 동작합니다. 이러한 볼륨을 인스턴스에 디바이스로 마운트할 수 있습니다. 인스턴스에 연결된 EBS 볼륨은 스토리지 볼륨으로 표시되며, 인스턴스 수명에 관계없이 지속됩니다.
- Amazon FSx — Amazon FSx는 ONTAP, OpenZFS, 윈도우 파일 서버, 러스터 등 네 가지 파일 NetApp 시스템을 제공합니다. 적합한 시스템을 선택하는 방법에 대한 지침은 Amazon FSx 설명서의 [Amazon FSx 파일 시스템 선택](#)을 참조하세요. Amazon FSx는 Microsoft 워크로드를 AWS로 마이그레이션하고 IT 직원의 운영 오버헤드를 일부 제거할 수 있도록 다양한 파일 시스템 유형의 관리 파일 스토리지 솔루션을 제공합니다. 이를 통해 IT는 다른 중요한 비즈니스 동인에 집중할 수 있습니다.
- AWS Snow Family - AWS로 이동해야 할 페타바이트 규모의 데이터가 있는 경우 [AWS Snow Family](#)의 스토리지 솔루션 사용을 고려해 보세요. 스토리지는 데이터의 장기적인 수명 동안 AWS Snow Family 디바이스에 의존하지 않지만 AWS Snowcone, AWS Snowball 또는 AWS Snowmobile 디바이스를 사용하여 대규모 데이터 세트를 AWS 오프라인으로 시드하는 데 도움이 될 수 있습니다. 자세한 내용은 AWS [Snowball을 사용하여 대규모 SQL 데이터베이스를 원활하게 마이그레이션하는 방법과 AWS 스토리지 DataSync 블로그의 AWS](#) 게시물을 참조하십시오.

워크로드에 맞는 스토리지 서비스를 식별한 후 운영 데이터를 이동하기 전에 스트레스/로드 테스트 도구를 사용하여 테스트를 수행하는 것이 좋습니다. 예를 들어, Amazon FSx for Windows File Server에서 SQL 데이터베이스를 이동하는 경우 [Microsoft SQL Server Distributed Replay](#)를 사용할 수 있습니다. 마찬가지로 일반 IOPS 및 처리량에 [DISKSPD](#)를 사용할 수 있습니다.

동원

스토리지 서비스를 식별한 후 다음 단계는 데이터 전송을 위한 도구를 선택하는 것입니다. [Robocopy](#)와 같은 이전 솔루션과 [DataSync](#)와 같은 최신 도구를 비롯한 여러 도구를 사용할 수 있습니다. DataSync Robocopy와 같은 도구에서는 사용할 수 없는 여러 제어 기능이 포함되어 있습니다. 여기에는 전체 네트워크 트래픽에 영향을 주지 않으면서 데이터를 마이그레이션하는 데 도움이 되는 예약 전송, 간편한 네트워크 제한 제어 등이 포함됩니다. Data Sync로 완료한 성공적인 마이그레이션에 대한 자세한 내용은 AWS [고객의 고객 추천사](#)를 참조하십시오. DataSync

Robocopy를 사용하는 것이 더 편하다면 Robocopy를 사용하여 데이터를 AWS로 마이그레이션할 수 있습니다. [파일 전송 성능](#)을 최적화하는 방법에 대해 이 가이드를 검토하는 것이 좋습니다. 이 가이드는 마이그레이션 중에 문제가 발생하는 것을 방지하는 데 도움이 될 수 있습니다. 중복 제거가 활성화된 파일 시스템에서 Robocopy를 사용하는 경우 데이터 손상 문제를 방지하려면 Amazon FSx Windows 사용 설명서의 [Data deduplication](#)과 Microsoft 설명서의 [Troubleshooting Data Deduplication Corruptions](#)를 참조하세요.

[AWS Storage Gateway](#)는 파일, 볼륨 및 가상 테이프의 3가지 방법으로 데이터를 AWS로 마이그레이션할 수 있습니다. 온프레미스에서 실행되는 VMware 또는 Hyper-V 하이퍼바이저, Amazon VPC의 Amazon EC2 인스턴스 또는 전용 하드웨어 어플라이언스에 Storage Gateway를 설치할 수 있습니다.

Storage Gateway를 통해 온프레미스와 AWS 간의 격차를 해소하고 비용을 절감할 수 있습니다. Storage Gateway를 사용하여 마이그레이션을 단계적으로 구현하고 이를 통해 온프레미스 백업 디바이스와 테이프를 가상 테이프 라이브러리(VTL)로 교체할 수 있습니다. 또한 Storage Gateway를 아카이브 스토리지 솔루션으로 사용하여 마이그레이션의 첫 번째 단계로 사용하지 않는 로컬 파일만 AWS로 마이그레이션할 수도 있습니다. Storage Gateway를 사용하여 AWS에서 Microsoft 워크로드를 호스팅하는 데는 여러 가지 옵션이 있습니다.

마이그레이션

DataSync 그리고 Robocopy는 모두 네트워크 액세스 제어 목록 (ACL, Windows ACL이라고도 함)을 보존할 수 있는 기능을 갖추고 있습니다. 마이그레이션을 시작하기 전에 [icacls](#)를 사용하여 ACL의 백업 사본을 만들고 다음 리소스를 검토하는 것이 좋습니다.

- [온프레미스 파일 공유를 NetApp ONTAP용 Amazon FSx로 마이그레이션](#) (AWS 스토리지 블로그)
- [Migrating existing file storage to Amazon FSx](#)(Amazon FSx Windows 사용 설명서)
- [AWS를 사용하여 VPC를 떠나지 않고 온프레미스에서 AWS로 파일을 전송하고 그 반대로 파일 전송](#) ([DataSyncAWS](#) 스토리지 블로그)
- [Migrate small sets of data from on premises to Amazon S3 using AWS SFTP](#)(AWS 권장 가이드)

SQL Server 마이그레이션

클라우드로의 여정에는 SQL Server 환경을 AWS로 마이그레이션하기 위한 여러 옵션이 있습니다. 성공적인 [마이그레이션](#)은 SQL Server 워크로드 및 해당 종속성에 대한 자세한 인벤토리 생성, 인증 체계 식별, 고가용성 및 재해 복구(HADR) 요구 사항 캡처, 성능 목표 평가, [라이선스 옵션](#) 평가를 기반으로 합니다. 이 인벤토리는 대상 데이터베이스 플랫폼을 결정하고 마이그레이션 옵션을 정의하는 데 도움이 됩니다.

SQL Server 워크로드를 AWS로 마이그레이션할 때 고려해야 할 옵션이 많이 있으며, 각각의 옵션은 최적화된 가격/성능, 보다 직관적인 사용자 경험 및 낮은 TCO를 제공합니다. [Amazon EC2](#), [Amazon RDS for SQL Server](#) 또는 [Amazon RDS Custom for SQL Server](#)에 SQL Server를 배포하도록 선택할 수 있습니다.

평가

마이그레이션을 성공적으로 구현하려면 기존 인프라를 평가하고 환경에 필요한 주요 기능을 이해하는 것이 중요합니다. 마이그레이션 계획을 선택하기 전에 다음 주요 영역을 검토하는 것이 좋습니다.

- 기존 인프라 검토 - 마이그레이션의 검색 단계에서 수집한 데이터를 사용하여 기존 SQL Server 인프라를 검토합니다([Windows 환경 검색](#) 참조). AWS의 SQL Server 인프라에는 Microsoft에서 규정한 크기 조정을 사용하는 것이 좋습니다. 메모리, CPU, IOPS 및 처리량을 포함한 온프레미스 SQL Server 인스턴스의 현재 사용률을 파악하는 것은 AWS에서 SQL Server 인스턴스의 크기를 적절히 조정하는 데 매우 중요합니다.
- 기존 라이선스 검토 - 보완적인 [AWS Optimizing and Licensing Assessment\(AWS OLA\)](#)를 활용하여 AWS에서 마이그레이션 및 라이선스 전략을 구축할 수 있습니다. AWS OLA는 기존 라이선스 자격을 사용하여 배포 옵션을 모델링하는 보고서를 제공합니다. 이 결과는 유연한 AWS 라이선스 옵션 전반에서 사용 가능한 비용 절감 효과를 탐색하는 데 도움이 될 수 있습니다.
- 기존 SQL Server 아키텍처 검토 - 공유 스토리지 또는 SQL Server Always On 가용성 그룹 아키텍처와 함께 SQL Server 장애 조치 클러스터를 사용하는 경우 현재고가용성 아키텍처 요구 사항을 이해하면 AWS에서 [SQL Server 배포 옵션](#)을 정의하는 데 도움이 됩니다.
- 백업 전략 개발 - SQL Server에서 네이티브 백업을 사용하여 데이터베이스를 클라우드에 백업할 수 있습니다. Storage Gateway를 사용하여 Amazon EBS, Amazon FSx for Windows File Server, Amazon FSx NETAPP ONTAP 및 Amazon S3에 데이터베이스를 백업하는 다양한 옵션이 있습니다. 또한 스냅샷 접근 방식을 사용하여 SQL Server 인스턴스를 백업할 수 있습니다. SQL Server 백업에 대한 자세한 내용은 AWS 권장 가이드 설명서의 [SQL Server on Amazon EC2의 백업 및 복원 옵션](#)을 참조하세요.
- 재해 복구(DR) 요구 사항 이해 - 기존 SQL Server 워크로드를 AWS로 이동하는 경우 보조 리전을 사용하고 복제가 가능한 Transit Gateway를 사용하여 두 리전을 연결할 수 있습니다. SQL Enterprise 에디션 내에서 SQL Server 분산 가용성 그룹 아키텍처를 사용하여 DR을 설정하거나 RTO 및 RPO 요구 사항에 따라 로그 전달을 사용할 수 있습니다. 또한 DR을 장애 조치 환경으로 남겨두는 액티브/패시브 구현으로 AWS Elastic Disaster Recovery(AWS DRS)를 사용할 수 있습니다. 자세한 내용은 AWS Database Blog의 [Architect a disaster recovery for SQL Server on AWS: Part 1](#)을 참조하세요.

동원

SQL Server 워크로드에 대해 고려해야 할 [3가지 주요 마이그레이션 옵션](#)이 있습니다.

- 리호스팅(리프트 앤드 시프트) - 여기에는 온프레미스 SQL Server 데이터베이스를 AWS 클라우드의 EC2 인스턴스에 있는 SQL Server로 마이그레이션하는 작업이 포함됩니다. 이 접근 방식은 AWS로 더 빠르게 마이그레이션하는 것이 우선인 경우 유용합니다.
- 리플랫폼(리프트 앤드 리세이프) - 여기에는 온프레미스 SQL Server 데이터베이스를 AWS 클라우드의 [Amazon RDS for SQL Server](#)로 마이그레이션하는 작업이 포함됩니다. 리플랫폼은 SQL Server를 계속 사용하고 싶지만 설치, 구성, 패치, 업그레이드 및고가용성 설정과 같은 확실적이고 과중한 작업을 없애려는 경우에 가장 적합합니다. Amazon EC2 기반 SQL Server, Amazon RDS 및 Amazon RDS Custom의 기능 비교는 AWS 권장 가이드 설명서의 [Amazon EC2와 Amazon RDS 중 선택](#)을 참조하세요.
- 리팩터링(리아키텍트) - 여기에는 일반적으로 오픈 소스 데이터베이스 또는 클라우드용으로 구축된 데이터베이스를 사용하여 애플리케이션을 변경하고 현대화하는 작업이 포함됩니다. 이 시나리오에서는 [Amazon RDS for MySQL](#), [Amazon RDS for PostgreSQL](#) 또는 [Amazon Aurora](#)를 사용하도록 온프레미스 SQL Server 데이터베이스를 현대화합니다. 오픈 소스 데이터베이스로 전환하면 라이선스 비용을 줄이고 불필요한 공급업체 종속 기간과 라이선스 감사를 방지할 수 있습니다.

마이그레이션

SQL Server 워크로드를 AWS로 마이그레이션할 때는 구성과 도구에 대한 다음 항목을 고려합니다.

리호스팅

[리호스팅은 동종](#) 방식입니다. 데이터베이스 소프트웨어나 구성을 변경하지 않고 SQL Server 데이터베이스를 있는 그대로 마이그레이션하려면 이 접근 방법을 선택합니다. 예를 들어 대규모 레거시 마이그레이션에서는 비즈니스 목표를 달성하기 위해 신속하게 이동하고 대부분의 애플리케이션을 리호스팅하도록 선택할 수 있습니다.

Amazon EC2를 사용하여 SQL Server 마이그레이션

Amazon EC2로 마이그레이션하는 경우 기존 SQL Server 라이선스를 사용할 수 있습니다. 이를 기존 보유 라이선스 사용(BYOL) 모델이라고 합니다. 또는 AWS에서 라이선스가 포함된 인스턴스를 구매할 수 있습니다. 자세한 내용은 AWS Cloud Operations & Migrations Blog의 [Cost optimization with SQL BYOL using license included Windows instance on Amazon EC2 Dedicated Hosts](#) 게시물을 참조하세요. BYOL 옵션을 사용하면 기존 SQL Server 라이선스를 사용하여 비용을 절감할 수 있습니다. [AWS License Manager](#)는 Amazon EC2에서 SQL Server로 VM을 인스턴스화할 때 사용 가능한 라이선스의

할당을 제어하는 데 도움이 됩니다. License Manager는 사용자가 지정한 라이선스 규칙을 준수하도록 도와줍니다.

Microsoft Software Assurance(SA)가 있는 경우에만 BYOL을 사용하여 SQL Server를 공유 테넌시(기본값) EC2 인스턴스로 리호스팅할 수 있습니다. SQL 라이선스에 SA가 없는 경우 라이선스를 2019년 10월 1일 이전에 구입했거나 2019년 10월 1일 이전에 유효한 활성 엔터프라이즈 등록에 따라 트루업으로 추가된 라이선스만 있으면 [Amazon EC2 전용 호스트](#)에 리호스팅할 수 있습니다.

백업 및 복원, 로그 전달, Always On 가용성 그룹과 같은 SQL Server 기능을 사용하여 SQL Server 데이터베이스를 Amazon EC2 인스턴스로 마이그레이션하는 방법이 있습니다. 이러한 옵션은 단일 데이터베이스 또는 데이터베이스 세트를 Amazon EC2에서 실행되는 새 SQL Server 인스턴스로 마이그레이션하는 경우에 적합합니다. 이러한 옵션은 데이터베이스 네이티브이며 특정 SQL Server 버전 및 에디션에 따라 다릅니다. 데이터베이스 마이그레이션 외에도 로그인, 작업, 데이터베이스 메일, 연결된 서버 등의 객체를 마이그레이션하는 단계를 수행해야 할 수도 있습니다.

AWS에서 SQL Server 데이터베이스를 리호스팅하는 데 다음 접근 방식을 사용할 수 있습니다.

- [Application Migration Service](#) 또는 [AWS Database Migration Service\(AWS DMS\)](#)를 사용하여 서버 리호스팅
- [SQL Server 백업 및 복원](#)
- [SQL Server 트랜잭션 복제](#)
- [클라우드 가용성 그룹 확장](#)
- [DMS](#)
- [로그 전달](#)

또한 [SQL 서버용 AWS Launch Wizard](#)를 사용하여 Amazon EC2에서 Microsoft SQL Server의 크기 조정, 구성 및 배포 과정을 안내할 수 있습니다. SQL Server 단일 인스턴스와 Amazon EC2에서의 HA 배포를 모두 지원합니다. 자세한 내용은 [SQL 서버용 AWS Launch Wizard](#)를 참조하세요.

Application Migration Service를 사용하여 SQL Server 마이그레이션

Application Migration Service는 가동 중지 시간을 거의 없애거나 최소화하면서 데이터베이스의 SQL Server 버전, 운영 체제 또는 코드를 변경하지 않고 하나 이상의 대규모 시스템을 온프레미스 환경에서 AWS로 리프트 앤드 시프트하려는 경우에 적합한 옵션입니다. [AWS Application Migration Service](#)를 사용하여 호환성 문제, 성능 영향 또는 긴 전환 기간 없이 빠르게 물리적, 가상 또는 클라우드 서버를 리프트 앤드 시프트할 수 있습니다. Application Migration Service를 사용하여 온프레미스 환경에서 Amazon EC2 인스턴스로 SQL Server 데이터베이스를 마이그레이션하는 방법에 대한 지침은

[Microsoft SQL Server 데이터베이스를 AWS 클라우드로 마이그레이션](#)을 참조하세요. 또한 Application Migration Service를 사용하여 Microsoft SQL Server 데이터베이스 워크로드를 AWS로 마이그레이션 할 때 [모범 사례](#)를 참조할 수 있습니다.

Linux의 SQL Server

SQL Server 데이터베이스 엔진은 기본적으로 Windows Server와 Linux 모두에서 비슷한 방식으로 실행됩니다. 그러나 Linux를 사용할 때 일부 작업이 변경되었습니다. [Launch Wizard](#)를 사용하면 이러한 변경 사항에 맞게 조정하고 고가용성 솔루션을 구성할 수 있습니다. 사내에 Linux 관리 전문 지식이 있는 경우 Amazon EC2 Linux로 리호스팅하는 것이 Windows Server 라이선스 비용을 절약하는 좋은 선택입니다. [Microsoft SQL Server 데이터베이스용 Windows에서 Linux로의 리플랫폼 어시스턴트](#) 도구를 사용하여 이 프로세스를 자동화하는 것을 고려해 보세요. 자세한 내용은 AWS 권장 가이드 설명서의 [Migrate an on-premises Microsoft SQL Server database to Microsoft SQL Server on Amazon EC2 running Linux](#)를 참조하세요.

플랫폼 재구성

리플랫폼은 완전관리형 데이터베이스 제품 및 서비스를 사용하여 데이터베이스 인스턴스를 관리하는 데 드는 시간을 줄이는 데 가장 적합한 [동종](#) 접근 방식입니다. Amazon RDS for SQL Server의 완전관리형 데이터베이스는 기본 운영 체제, 시스템 볼륨 또는 사용자 지정 드라이버 설치에 액세스하는 것을 제한합니다. 자세한 내용은 Amazon RDS 사용 설명서의 [Amazon RDS for Microsoft SQL Server](#)를 참조하세요. 사용 사례에 완전관리형 데이터베이스 기능이 필요하거나 기존 SQL Server 라이선스를 사용하려는 경우 SQL Server용 [Amazon RDS Custom](#)으로 리플랫폼을 고려하세요.

기존 보유 미디어 사용(BYOM) 옵션은 SQL Server용 Amazon RDS Custom에 사용할 수 있습니다. BYOM을 사용하면 자체 설치 미디어와 라이선스를 사용할 수 있지만 라이선스는 Microsoft의 라이선스 이동 약관을 준수해야 합니다. SQL Server를 Amazon RDS for SQL Server로 또는 Amazon RDS Custom for SQL Server로 리플랫폼할 수 있습니다. 선택은 기본 운영 체제에 대한 액세스가 필요한지, 데이터베이스 사용자 지정이 필요한지, BYOM을 사용하여 기존 SQL Server 라이선스를 활용하려는지에 따라 달라집니다.

다음은 Amazon RDS for SQL Server로 SQL Server를 마이그레이션하는 데 사용할 수 있는 방법입니다.

- [TSQL을 사용한 로그 전송 PowerShell 또는 TSQL을 사용한 로그 전송](#)
- [SQL Server 백업 및 복원](#)
- [트랜잭션 복제](#)
- [DMS](#)

Amazon RDS for SQL Server에서 실행되도록 SQL Server 데이터베이스를 리플랫폼하려면 [Amazon RDS for SQL Server 리소스](#)에서 제공하는 접근 방식을 사용하는 것이 좋습니다. 지원 종료 워크로드를 마이그레이션하는 방법에 대한 자세한 내용은 AWS Database Blog의 [Migrate end of support Microsoft SQL Server databases to Amazon RDS for SQL Server confidently](#) 게시물을 참조하세요. 온프레미스 데이터베이스에 대한 자세한 내용은 Amazon RDS 사용 설명서의 [온프레미스 데이터베이스를 SQL Server용 Amazon RDS Custom으로 마이그레이션](#)을 참조하세요.

리팩터링

리팩터링은 [이기종](#)입니다. 오픈 소스 및 데이터베이스 오퍼링을 활용하기 위해 데이터베이스와 애플리케이션을 재구성, 재작성 및 재설계할 준비가 되면 이 접근 방식을 선택하십시오. built-for-the-cloud 데이터베이스와 각 애플리케이션을 리팩터링할 의향이 있는 경우 SQL Server 워크로드를 Amazon RDS for MySQL, Amazon RDS for PostgreSQL, [Amazon Aurora MySQL 호환 버전](#) 또는 [Amazon Aurora PostgreSQL 호환 버전](#)으로 현대화할 수 있습니다. 다양한 현대화 타임라인과 성능 요구 사항에 따라 리팩터링할 수 있습니다.

Amazon RDS for MySQL과 Amazon RDS for PostgreSQL은 해당 오픈 소스 데이터베이스를 위한 완전관리형 데이터베이스 제품입니다. Amazon Aurora는 MySQL 및 PostgreSQL과 완벽하게 호환되는 클라우드용으로 구축된 관계형 데이터베이스 관리 시스템(RDBMS)입니다. Aurora는 내결함성 스토리지 시스템을 갖추고 있으며 1/10의 비용으로 상용 등급 데이터베이스의 성능과 가용성을 제공합니다.

또한 [Amazon Aurora Serverless](#)를 사용하면 데이터베이스 용량을 관리하지 않고도 AWS에서 데이터베이스를 실행할 수 있습니다. Amazon Aurora Serverless v2는 순식간에 수십만 건의 트랜잭션으로 즉시 확장됩니다. 애플리케이션에서 소비하는 용량에 대해서만 비용을 지불하며, 최대 로드에 대한 용량 프로비저닝 비용에 비해 데이터베이스 비용을 최대 90% 절약할 수 있습니다.

SQL Server 데이터베이스를 이러한 오퍼링 중 하나로 리팩터링하려면 AWS DMS와 함께 [AWS Schema Conversion Tool\(AWS SCT\)](#)을 사용하는 것이 좋습니다. 자세한 내용은 Microsoft SQL Server 데이터베이스를 AWS 클라우드로 마이그레이션 가이드의 [AWS SCT](#)를 참조하세요.

애플리케이션과 데이터베이스를 AWS로 더 빠르게 마이그레이션하는 것이 목표라면 [Babelfish for Aurora PostgreSQL](#)을 사용하는 것이 좋습니다. Babelfish를 사용하면 원래 SQL Server용으로 작성된 애플리케이션을 최소한의 코드 변경으로 Aurora와 연동할 수 있습니다. 따라서 SQL Server 2019 또는 이전 버전용으로 개발된 Babelfish for Aurora PostgreSQL 애플리케이션을 수정하고 이것으로 이동하는 데 필요한 노력이 줄어 더 빠르고 위험도가 낮으며 비용 효율적인 리팩터링이 가능합니다.

Babelfish를 사용하여 마이그레이션하려면 다음 리소스를 고려해 보세요.

- [Migrate from SQL Server to Amazon Aurora using Babelfish](#)(AWS Database Blog)
- [Prepare for Babelfish migration with the AWS SCT assessment report](#)(AWS Database Blog)

- [Migrate from SQL Server to Aurora PostgreSQL using SSIS and Babelfish](#)(AWS Database Blog)
- [Babelfish를 AWS Database Migration Service 대상으로 사용](#)(AWS Database Migration Service 사용 설명서)

추가적인 리소스

- [Microsoft SQL Server 데이터베이스를 AWS 클라우드로 마이그레이션](#)(AWS 권장 가이드)
- [Migration and Modernization Strategies for your SQL Server on AWS](#)(AWS Blog)

.NET 애플리케이션 마이그레이션

.NET 애플리케이션을 AWS로 마이그레이션하면 탄력적 조정 기능을 갖춘 고가용성 워크로드를 생성하고, 운영 오버헤드를 줄이고, 차별화된 가치에 집중하여 비즈니스 민첩성을 높일 수 있습니다. 이 섹션에서는 AWS에서 .NET 애플리케이션을 호스팅하기 위한 다양한 옵션에 중점을 둡니다. VM, [AWS Elastic Beanstalk](#)와 같은 관리형 솔루션 사용, 코드 컨테이너화, 마이크로서비스 또는 서버리스 기반 아키텍처로 코드 리팩터링 중에서 선택할 수 있습니다.

평가

.NET 워크로드의 마이그레이션 경로를 선택할 때는 다음과 같은 주요 요소를 고려해야 합니다.

- 사용된 .NET 버전 찾기 - Microsoft에서 지원하는 두 가지 .NET 구현인 .NET Framework(1.0~4.8)와 .NET(.NET Core 1.0~3.1 및 .NET 5 이상)이 있습니다. 둘 다 동일한 구성 요소를 많이 공유하며 서로 다른 .NET 프로그래밍 언어(예: C#, F#, VB.NET)를 사용하여 작성된 애플리케이션 코드를 실행할 수 있습니다. .NET Framework는 Windows에서 실행되지만 최신 .NET은 다중 플랫폼이므로 마이그레이션 전략과 호스팅 서비스 선택은 사용하는 런타임에 따라 달라집니다. .NET Framework의 경우 Windows OS에서 호스팅하거나 최신 .NET을 사용하도록 코드를 리팩터링할 수 있습니다. Linux OS 기반 서비스에서도 최신 .NET을 호스팅할 수 있습니다. .NET Framework 기반 워크로드를 현대화할 때 [Porting Assistant for .NET](#) 또는 [AWS Toolkit for .NET Refactoring](#)을 사용하여 코드를 스캔하고 호환성 평가 보고서를 생성할 수 있습니다. 프로젝트에서 참조하는 호환되지 않는 .NET Framework API가 있는지 확인하여 마이그레이션 프로젝트의 복잡성을 계획하고 최신 런타임을 사용하도록 코드를 리팩터링할지 여부와 시기를 결정할 수 있습니다.
- 현재 배포 검토 - 현재 마이그레이션된 워크로드에 동일한 워크로드를 클라우드에 배포하도록 업데이트할 수 있는 기존 CI/CD 파이프라인이 있는지 확인합니다. 기존 빌드 및 배포 파이프라인을 사용하면 워크로드를 구축, 구성 및 배포하는 데 필요한 단계를 자동화하여 애플리케이션을 클라우드에 배포하는 데 걸리는 시간을 줄일 수 있습니다.

- 로드맵 검토 - 프로젝트의 현재 상태에 따라 이미 애플리케이션을 리팩터링하거나 재설계할 계획을 세우고 있을 수 있습니다. 현대화를 수행할 때는 제품 로드맵을 고려해야 합니다. 예를 들어, 기존 코드를 컨테이너화하기로 결정하거나 모놀리식 아키텍처를 마이크로서비스로 리팩터링하는 것은 이상적으로 제품 로드맵의 일부이며 다른 개발 노력과 연계됩니다.

동원

.NET 워크로드를 AWS로 마이그레이션할 때 고려할 3가지 마이그레이션 경로가 있습니다. 기존 코드베이스의 복잡성, 마이그레이션에 할당된 시간, 마이그레이션 작업을 지원하기 위해 할당된 팀의 규모에 따라 다양한 옵션 중에서 선택할 수 있습니다. 마이그레이션의 일부로 현대화를 고려할 때는 제품 로드맵에 맞추는 것이 가장 좋습니다.

- 리호스팅(리프트 앤드 시프트) - 변경 사항이 거의 또는 전혀 없이 AWS로 더 빠르게 마이그레이션하는 것이 우선인 경우 이 접근 방식을 선택할 수 있습니다. Amazon EC2 인스턴스에서 실행되는 인터넷 정보 서비스(IIS)로 ASP.NET 기반 웹 사이트를 리호스팅할 수 있습니다. [데스크톱 기반 애플리케이션 \(예: Windows 프레젠테이션 파운데이션, 웹 양식, .NET MAUI\) 을 Amazon 2.0 또는 Amazon과 같은 최종 사용자 컴퓨팅 플랫폼 중 하나로 재호스팅할 수 있습니다. AppStream WorkSpaces](#)
- 리플랫폼 - 리플랫폼은 코드를 변경하지 않고 관리형 서비스를 사용하여 애플리케이션을 호스팅하고 싶지만 설치, 패치, 업그레이드, 인스턴스 관리 등의 획일적이고 과중한 업무를 없애서 운영 오버헤드를 줄이려는 경우에 가장 적합합니다. 이 전략은 컨테이너 기반 워크로드로 전환하려는 팀에도 적합합니다. 기존 애플리케이션을 [Elastic Beanstalk](#)로 리플랫폼하거나 [Amazon ECS](#), [Amazon EKS](#) 또는 [AWS App Runner](#)에서 호스팅되는 Docker 컨테이너를 사용할 수 있습니다.
- 리팩터링 - AWS 클라우드 네이티브 서비스를 사용하여 운영 오버헤드를 줄이고 더 나은 확장, 고가용성 및 재해 복구를 달성하는 코드 및 아키텍처 변경에 시간과 노력을 투자할 수 있다면 이 접근 방식을 선택합니다. 리팩터링에는 기존 .NET 프레임워크 애플리케이션을 .NET(이전의 .NET Core)으로 이식하거나 클라우드에서 더 잘 실행되도록 기존 코드베이스를 현대화하는 작업이 포함됩니다. [AWS SDK for .NET](#)을 사용하여 .NET 코드 내에서 많은 AWS 클라우드 서비스를 직접적으로 호출할 수 있습니다. [Porting Assistant for .NET](#), [AWS Microservice Extractor for .NET](#) 등의 도구를 사용하여 코드베이스를 .NET Framework에서 .NET으로 이식하고 모놀리식 애플리케이션을 마이크로서비스로 분리할 수 있습니다. [Lambda](#)에서 실행되도록 기존 .NET 워크로드를 리팩터링하면 서버리스 컴퓨팅을 사용하여 인프라를 프로비저닝하고 관리하지 않아도 됩니다.

마이그레이션

.NET 워크로드 마이그레이션 단계는 평가 단계에서 선택한 마이그레이션 경로와 애플리케이션 유형에 따라 달라집니다.

.NET 애플리케이션 리호스팅

코드를 변경하지 않고 애플리케이션을 마이그레이션하고 클라우드의 자동 조정, 로드 밸런싱, 탄력성을 활용하려는 경우 이 마이그레이션 경로를 선택합니다. Windows 기반 웹 사이트의 경우 리호스팅은 일반적으로 AWS의 인터넷 정보 서비스(IIS)에서 실행하는 것을 의미합니다. 데스크톱 기반 애플리케이션의 경우 애플리케이션을 설치하고 사용자가 외부에서 애플리케이션에 연결할 수 있도록 해야 합니다.

AWS의 인터넷 정보 서비스

인터넷 정보 서비스(IIS)는 Windows 운영 체제에서 실행되는 Microsoft 웹 서버이며 웹 사이트와 웹 서비스를 호스팅하는 데 사용됩니다. Windows Server를 실행하는 모든 EC2 인스턴스에 IIS를 설치할 수 있습니다. IIS를 사용하도록 설정하고 구성한 후에는 온프레미스 환경에 사용하는 것과 동일한 배포 메커니즘을 사용하여 ASP.NET 웹 사이트 및 서비스를 배포할 수 있습니다.

EC2 Windows 인스턴스에서 IIS를 호스팅하는 경우 워크로드 및 HADR 요구 사항에 따라 로드 밸런싱, 오토 스케일링 및 다중 AZ 배포를 사용하여 [AWS Well-Architected Framework](#)를 따르는 것이 중요합니다. AWS에서 IIS 리소스를 실행하는 Windows Server 워크로드의 크기 조정, 구성 및 배포 과정을 안내하는 [AWS Launch Wizard](#)를 사용하는 것이 좋습니다. Launch Wizard는 새로 생성된 VPC 또는 기존 VPC에 필요한 컴퓨팅, 네트워킹 및 스토리지 구성 요소와 함께 두 개의 가용 영역에 걸친고가용성 아키텍처를 배포합니다.

AWS에서 데스크톱 애플리케이션 호스팅

많은 클라이언트가 Windows 기반 싹 클라이언트 애플리케이션에 액세스해야 합니다. 다음과 같은 3가지 플랫폼 중에서 선택할 수 있습니다.

- [Amazon EC2](#) - 사용자가 Microsoft 원격 데스크톱을 사용하여 Windows Server 기반 환경에 연결하도록 하려면 이 옵션을 선택합니다. 이 옵션을 사용할 경우 사용자가 운영 체제의 패치 적용 및 유지 관리를 담당합니다. 또한 사용자와 [활성 소프트웨어 보증\(SA\)](#)을 위한 원격 데스크톱 서비스 클라이언트 액세스 라이선스(RDS CAL)도 추가로 구입해야 합니다. 자세한 내용은 AWS 설명서의 [AWS에서의 Microsoft 라이선스](#)를 참조하세요.
- [Amazon WorkSpaces](#) — 사용자를 위한 완전 관리형 가상 데스크톱 인프라 (VDI)가 필요한 경우 이 옵션을 선택하십시오. WorkSpaces 사용하여 사용자에게 지속적인 Windows 데스크톱 환경을 제공할 수 있습니다. 또한 사용자 지정 이미지를 사용하여 WorkSpaces 환경을 사용자 지정하고.NET 애플리케이션을 설치하거나, [AWS System Manager](#)를 사용하여 환경에 .NET 애플리케이션을 제공할 수 WorkSpaces 있습니다. 사용자는 브라우저 또는 [Amazon WorkSpaces 클라이언트](#)를 사용하여 연결할 수 있습니다.

- [Amazon AppStream 2.0](#) — 어느 위치에서든 애플리케이션 및 비영구 데스크톱에 대한 안전하고 안정적인 확장 가능한 액세스를 제공하려면 이 옵션을 선택하십시오. AppStream 2.0을 사용하여 사용자가 웹에서 .NET 애플리케이션에 액세스할 수 있도록 할 수 있습니다. [기존 RDS CAL과 활성 SA가 이미 있는 경우 라이선스 모빌리티를 사용하여 AppStream 2.0에서 해당 라이선스를 사용할 수 있습니다.](#)

리플랫폼

리플랫폼에는 코드를 거의 또는 전혀 변경하지 않고 호스팅 환경을 변경하는 작업이 포함됩니다. 이 전략을 선택하여 운영 오버헤드를 줄이고 클라우드 기능 및 서비스를 활용합니다.

AWS Elastic Beanstalk

[Elastic Beanstalk](#)를 사용하여 .NET Framework 워크로드를 리플랫폼할 수 있습니다. ASP.NET 기반 또는 ASP.NET Core 기반 애플리케이션을 패키징하는 경우 해당 애플리케이션을 실행하는 인프라에 대해 배울 필요 없이 AWS에서 애플리케이션을 신속하게 배포하고 관리할 수 있습니다. 이는 선택 또는 제어에 대한 제한 없이 복잡성을 줄입니다. 애플리케이션을 업로드하기만 하면 Elastic Beanstalk에서 용량 프로비저닝, 로드 밸런싱, 조정 및 애플리케이션 상태 모니터링의 세부 정보를 자동으로 처리합니다.

자세한 내용은 다음 리소스를 참조하세요.

- [Elastic Beanstalk에서 .NET 애플리케이션 생성 및 배포](#)(AWS Elastic Beanstalk 개발자 안내서)
- [Linux에서 .NET Core를 사용한 작업](#)(AWS Elastic Beanstalk 개발자 안내서)
- [Multi-App Support with Custom Domains for .NET and AWS Elastic Beanstalk](#)(AWS Developer Tools Blog)

기존 애플리케이션 컨테이너화

Amazon ECS 또는 Amazon EKS를 사용하여 Docker 기반 컨테이너식 애플리케이션을 호스팅할 수 있습니다. AWS는 두 서비스를 모두 관리합니다. 둘 중 하나를 선택하는 것은 기존 지식과 선호도에 따라 달라집니다. 두 옵션 모두 Linux 기반 컨테이너나 Windows 기반 컨테이너를 실행할 수 있습니다.

자세한 내용은 다음 리소스를 참조하세요.

- [Amazon EC2 Windows 컨테이너](#)(Amazon ECS 개발자 안내서)
- [Amazon EKS 클러스터에 대해 Windows 지원 사용 설정](#)(Amazon EKS 사용 설명서)

- [Running Windows Containers with Amazon ECS on AWS Fargate](#)(AWS 블로그)
- [Speeding up Windows container launch times with EC2 Image builder and image cache strategy](#)(AWS 블로그)
- [Quick start: CI/CD for .NET Applications on AWS Fargate](#)(AWS 설명서)

.NET 기반 애플리케이션을 컨테이너화하는 것은 사용된 .NET 런타임에 따라 달라집니다. 다음을 고려하세요.

- Windows 컨테이너에서 .NET Framework 기반 애플리케이션 실행 - 애플리케이션을 컨테이너화하는 방법을 약속하는 Docker 파일을 생성하여 기존 애플리케이션에 Docker 지원을 추가할 수 있습니다. [AWS App2Container](#)를 사용하여 기존 .NET Framework 기반 애플리케이션을 쉽게 컨테이너화하고 AWS로 마이그레이션할 수 있습니다. App2Container는 IIS 서버를 스캔하여 필요한 파일을 확인하고 대상 애플리케이션을 추출하여 도커 이미지를 생성합니다. 또한 App2Container를 사용하여 AWS 클라우드에서 애플리케이션을 호스팅하는 데 필요한 배포 아티팩트를 생성할 수 있습니다.
- .NET 또는 .NET Core - [Amazon ECS 또는 Amazon EKS에서 최신 .NET 기반 웹 애플리케이션을 실행하는 것 외에도 AWS App Runner](#)를 사용할 수 있습니다. App Runner는 코드나 컨테이너 이미지를 실행하고 부하 분산, Auto Scaling, 로깅, 인증서, 네트워킹을 관리하는 서버리스 완전관리형 솔루션입니다.

기존 코드 리팩터링/리아키텍트

애플리케이션의 현재 환경에서 달성하기 어려운 기능, 확장 또는 성능을 추가해야 하는 강력한 비즈니스 요구가 있는 경우 이 옵션을 선택합니다. 애플리케이션 로드맵에 따라 최신 프레임워크, 클라우드 네이티브 서비스를 사용하도록 코드를 변경하거나 클라우드에서 더 잘 실행되도록 리아키텍트할 수 있습니다.

사용할 수 있는 첫 번째 리팩터링 옵션은 기존 .NET Framework 애플리케이션을 .NET으로 마이그레이션하는 것입니다. .NET으로 전환하면 Windows 대신 Linux에서 실행할 수 있는 이점이 있습니다. 이렇게 하면 총 라이선스 비용이 절감되고, 최신 프레임워크와 최신 버전의 .NET 프로그래밍 언어가 제공됩니다.

AWS SDK for .NET

[AWS SDK for .NET](#)은 .NET 개발자에게 일관되고 친숙한 라이브러리 세트를 제공하여 AWS 서비스 사용을 간소화합니다. AWS SDK는 플랫폼 간 지원을 제공하며 를 사용하여 배포됩니다. NuGet 개발자는 AWS SDK를 사용하여 .NET 코드에서 클라우드 서비스를 쉽게 직접적으로 호출하여 애플리케이션의 스토리지, 대기열, 인증 및 구성 요구 사항을 충족할 수 있습니다.

.NET Framework 애플리케이션 현대화

코드 파일을 검사하고 애플리케이션 포트폴리오 마이그레이션 로드맵을 계획하는 데 도움이 되는 보고서를 생성하는 [Porting Assistant for .NET](#)을 사용하여 .NET Framework에서 마이그레이션할 수 있습니다. Porting Assistant는 호환되지 않는 .NET Core API 및 패키지를 식별하고 알려진 대체 API를 찾아 이식 오버헤드를 줄일 수도 있습니다. [AWS Toolkit for .NET 리팩터링](#)은 개발자가 레거시 .NET 애플리케이션을 AWS에서 클라우드 기반 대안으로 리팩터링하는 데 필요한 시간과 노력을 줄여주는 Visual Studio 확장입니다. 이 확장은 애플리케이션 소스 코드를 평가하여 .NET Core로 이식과 같은 가능한 현대화 경로를 추천하고, Windows 전용 IIS 및 Active Directory 종속성 구성을 식별하고, 가능한 경우 코드 수정을 수행하여 Linux 호환성을 활성화하고, AWS 서비스에서 리팩터링된 애플리케이션을 검증하는 데 도움이 됩니다. .NET Framework 애플리케이션을 .NET으로 마이그레이션하면 더 나은 가격 대비 성능으로 ARM64 기반 Graviton 프로세서에서 애플리케이션을 실행할 수 있습니다. 자세한 내용은 AWS 워크숍 스튜디오 [설명서에서 GitHub Graviton과 Graviton2에서의 .NET 및 AWS Graviton 기반 서비스를 통한 비용 최적화의 컨테이너를 참조하십시오.](#)

모노리스에서 마이크로서비스로

많은 개발 팀이 기존 모놀리식 애플리케이션을 마이크로서비스로 리아키텍트하고자 합니다. 개발 팀은 마이크로서비스 기반 아키텍처로 전환하여 개발 민첩성을 높이고, 컴퓨팅 비용을 줄이고, 서비스를 개별적으로 확장하고, 배포 시간을 단축할 수 있습니다. [AWS Microservice Extractor for .NET](#)은 이전의 모놀리식 애플리케이션을 마이크로서비스 기반 아키텍처로 리팩터링하는 프로세스를 간소화합니다. 구성 요소를 식별하고 기능을 그룹화함으로써 개발 팀은 .NET Framework 모놀리식 애플리케이션에서 .NET 서비스로 기능을 점진적으로 추출할 수 있습니다.

서버리스 애플리케이션으로 리팩터링

[AWS Lambda](#)는 서버를 프로비저닝하거나 관리하지 않고도 사실상 모든 유형의 애플리케이션이나 백엔드 서비스에 대한 코드를 실행할 수 있는 서버리스 이벤트 기반 컴퓨팅 서비스입니다. 기존 애플리케이션에서 로직을 추출하여 필요할 때 .NET과 Lambda를 사용하여 자동으로 확장되는 이벤트 기반 서버리스 워크플로를 생성할 수 있습니다. [Lambda의 일반적인 사용 사례](#)에는 파일 처리, 분석, 웹 사이트, 모바일 애플리케이션 등의 다양한 크기 조정 요구 사항에 따라 몇 초 또는 몇 분 동안 실행되는 이벤트 기반 워크로드가 포함됩니다. 자세한 내용은 Lambda 개발자 안내서의 [C#을 사용하여 Lambda 함수 빌드](#)를 참조하세요.

추가적인 리소스

- [아마존 CodeCatalyst](#) (아마존 CodeCatalyst 설명서)
- [Azure용 AWS 툴킷](#) (DevOpsAWS 설명서)

- [젠킨스를 AWS CodeBuild 및 AWS와 통합하여 CI/CD 파이프라인 설정 CodeDeploy \(AWS 블로그\) DevOps](#)
- [.NET용 AWS 배포 도구 \(AWS GitHub\) 에 대한 정보](#)
- [AWS 기반 .NET\(AWS 설명서\)](#)
- [AWS/닷넷 \(\) GitHub](#)

Windows 장애 조치 클러스터 마이그레이션

[Microsoft 장애 조치 클러스터](#)는 서버 간에 스토리지가 대부분 공유되는 서버 그룹입니다. 장애 조치 클러스터를 사용하여 애플리케이션과 서비스의고가용성을 촉진할 수 있습니다. 또한 AWS 클라우드로 장애 조치 클러스터를 마이그레이션하여 신뢰성, 성능 및 낮은 TCO의 이점을 누릴 수 있습니다.

Windows 장애 조치 클러스터는 온프레미스 환경과 클라우드에서 다르게 작동합니다. 다중 서브넷 클러스터만 클라우드에 배포할 수 있다는 점에 유의해야 합니다. 온프레미스 환경과 달리 Windows 장애 조치 클러스터의 IP 주소는 운영 체제 수준이 아닌 Elastic Network Adapter(ENA)에 할당됩니다. 온프레미스 환경에서는 운영 체제가 IP 주소 할당을 처리하지만 클라우드 제공업체(AWS)가 클라우드에서 IP 주소 할당을 처리합니다. 장애 조치 클러스터링은 운영 체제 수준의 기능이므로 IP 장애 조치를 제어할 수 없습니다. 따라서 노드 간에 동일한 IP가 장애 조치될 수 없습니다. 이 문제를 해결하려면 클러스터가 보조 IP로 장애 조치되는 다중 서브넷 클러스터를 사용합니다. 보조 IP는 다른 서브넷의 ENA에 할당되며 온라인 상태가 될 수 있습니다. 자세한 내용은 Microsoft 설명서의 [Failover Clustering Networking Basics and Fundamentals](#)를 참조하세요.

AWS로 Windows 장애 조치 클러스터를 마이그레이션하는 과정은 복잡할 수 있지만 신중한 계획과 구현을 통해 비즈니스 운영 중단을 최소화하면서 수행할 수 있습니다. 예를 들어 장애 조치 클러스터에서는 모든 애플리케이션이 다르게 구성되므로 먼저 요구 사항을 이해한 다음 클라우드에서 이러한 요구 사항을 충족할 수 있는 방법을 먼저 알아보는 것이 중요합니다. 업로드 프로세스는 다음과 같은 단계로 구성됩니다.

- 모든 클러스터 노드가 동일한 버전의 Windows와 필요한 모든 업데이트를 실행하는지 확인
- 클러스터 쿼럼 구성
- 마이그레이션 중 모든 애플리케이션 및 데이터를 백업하고 복원할 수 있는지 확인

평가

평가 단계는 AWS로 장애 조치 클러스터를 마이그레이션하는 과정에서 중요한 단계입니다. 이 단계에서는 현재 환경에 대한 정보를 수집하고, AWS로 마이그레이션의 타당성을 판단하고, 잠재적인 문제나 위험을 파악합니다. 평가 단계에서는 다음 단계를 따르는 것이 좋습니다.

- 애플리케이션의 준비 상태 평가 - 애플리케이션을 수정 없이 AWS로 마이그레이션할 수 있는지, 아니면 클라우드 네이티브 서비스 활용을 위해 업데이트하거나 다시 작성해야 하는지 결정합니다.
- 네트워킹 및 보안 요구 사항 평가 - 방화벽, 로드 밸런서 및 VPN의 구성을 비롯한 네트워크 및 보안 요구 사항을 결정합니다.
- 데이터 마이그레이션 요구 사항 평가 - 데이터의 크기 및 위치, 마이그레이션에 필요한 시간, 데이터 전송 비용을 포함하여 데이터가 AWS로 마이그레이션되는 방법을 결정합니다. 온프레미스 환경에서는 JBOD, NAS, SAN과 같은 다양한 스토리지 기술을 사용하고 있을 수 있습니다. 각 기술은 SAN Fiber Channel, iSCSI, SAS 또는 SMB/NFS 공유와 같은 다양한 액세스 방법을 통해 애플리케이션에 데이터를 제공할 수 있습니다.
- 잠재적 위험 및 과제 파악 - 가동 중지 시간, 호환성 문제 또는 데이터 손실 등 마이그레이션 프로세스에 영향을 미칠 수 있는 잠재적 위험 또는 과제를 파악합니다.
- 예상 비용 - EC2 인스턴스, 스토리지, 데이터 전송 및 필요한 기타 AWS 서비스 비용을 포함하여 AWS로 마이그레이션하는 데 드는 비용을 추정합니다.
- 마이그레이션 계획 수립 - 평가 단계에서 수집한 정보를 바탕으로 타임라인, 필요한 리소스, AWS로 마이그레이션하는 데 필요한 단계를 포함하는 상세한 마이그레이션 계획을 수립합니다.

현재 환경 평가

하드웨어 및 소프트웨어 구성을 포함한 현재 환경을 평가하여 AWS로 마이그레이션해야 할 사항을 결정합니다. 애플리케이션, 서버 및 데이터베이스 간의 종속성을 식별합니다.

마이그레이션 전략 결정

클라우드 네이티브 서비스를 활용하기 위한 lift-and-shift 접근 방식 또는 환경 재구성을 포함하여 AWS로 마이그레이션하기 위한 옵션을 고려해 보십시오.

- 기존 장애 조치 클러스터 마이그레이션 - 클라우드에서 클러스터를 처음부터 구성하는 경우 HPC 관련 단계는 건너뛰고 Amazon EC2 - Windows 인스턴스용 사용 설명서의 [자습서: Amazon EC2에 Windows HPC 클러스터 설치](#) 단계를 따를 수 있습니다. 또는 SQL 관련 단계를 거치지 않고 [SQL Server Always On 가용성 그룹 클러스터를 생성](#)할 수도 있습니다. 공유 스토리지는 장애 조치 클러

스터 마이그레이션에서 가장 중요한 고려 사항 중 하나입니다. Amazon EBS 다중 연결은 SCSI-3 영구 예약을 지원하지 않지만 [Windows File Server용 Amazon FSx와 ONTAP용 FSx NetApp 모두 공유 스토리지 옵션에서](#) 잘 작동합니다. 가장 일반적인 사용 사례 중 하나는 Amazon FSx for Windows File Server와 함께 SQL Server 클러스터용 Always On 장애 조치 클러스터 인스턴스를 사용하는 것입니다. 자세한 내용은 AWS Storage Blog의 [Simplify your Microsoft SQL Server high availability deployments using Amazon FSx for Windows File Server](#) 게시물을 참조하세요. 다음 단계로 노드를 클라우드로 가져옵니다. Application Migration Service를 사용하면 됩니다. 자세한 내용은 AWS 스토리지 블로그의 [CloudEndure마이그레이션을 사용하여 Microsoft Windows 클러스터를 AWS로 마이그레이션하기](#) 게시물을 참조하십시오. 그런 다음 고가용성을 제공하도록 애플리케이션에 대한 클러스터된 역할을 구성할 수 있습니다.

- 확장 클러스터를 사용하여 사실상 가동 중지 시간 없이 마이그레이션 – 클라우드로 마이그레이션할 비즈니스에 중요한 애플리케이션이 있고 가동 중지 시간을 감당할 수 없는 경우 확장 클러스터가 적합할 수 있습니다. [Microsoft 확장 클러스터](#)를 사용하면 사이트 A와 사이트 B가 네트워크를 통해서로 통신해야 하지만 개별 공유 스토리지를 가질 수 있습니다. 마이그레이션 시나리오에서 이를 유리하게 활용할 수 있습니다. 예를 들어, 소스(온프레미스이든 다른 제공업체의 클라우드에 있든)는 사이트 B를 배포하는 Amazon VPC와 네트워크 연결이 있는 사이트 A일 수 있습니다. 사이트 B가 실행되고 나면 사이트 B로 이동할 수 있습니다. 이 접근 방식에서는 소스 스토리지 기술이 작동할 수 있는 복제 방법에 대한 제한 요소가 있을 수 있기 때문에 데이터 복제 메커니즘이 매우 중요합니다.
- VMware 온 프레미스에 배포된 장애 조치 클러스터를 AWS 기반 클라우드의 VMware로 마이그레이션하는 경우, VMware Cloud on AWS는 SCSI-3 영구 예약을 기본적으로 지원합니다. 이를 통해 VMware Cloud on AWS의 가상 머신 디스크 (VMDK) 에서 장애 조치 클러스터를 호스팅할 수 있습니다. 자세한 내용은 VMware 설명서의 [Migrating SQL Server FCI cluster with shared disks to VMware Cloud on AWS](#)를 참조하세요.

알림

2024년 4월 30일부터 VMware Cloud AWS on은 더 이상 채널 파트너나 해당 채널 파트너에 의해 재판매되지 않습니다. AWS 이 서비스는 Broadcom을 통해 계속 제공될 예정입니다. 자세한 내용은 AWS 담당자에게 문의하시기 바랍니다.

- Amazon EBS 다중 연결 볼륨을 사용하여 SQL Server FCI 마이그레이션 — Amazon EBS 다중 연결 및 NVMe 예약을 사용하여 Amazon EBS 볼륨을 Windows Server 장애 조치 클러스터의 공유 스토리지로 사용하는 SQL 서버 장애 조치 클러스터 인스턴스 (FCI) 를 생성할 수 있습니다. io2 이러한 볼륨은 동일한 가용 영역에 있는 인스턴스에만 연결할 수 있습니다. Amazon EBS io2 볼륨을 사용하여 Windows Server 장애 조치 클러스터를 배포하려면 SCSI 예약 명령을 NVMe 예약 명령으로 변환하는 최신 Windows 드라이버가 필요합니다. 이 접근 방식을 사용하여 온프레미스 SQL Server FCI를 단일 가용 영역의 AWS로 마이그레이션하는 [방법에 대한 자세한 내용은 AWS 블로그 게시물](#)

Windows Server에 Amazon EBS Multi-Attach를 사용하여 SQL Server 장애 조치 클러스터를 배포하는 방법을 참조하십시오.

평가 단계는 장애 조치 클러스터를 AWS로 성공적으로 마이그레이션하는 데 매우 중요합니다. 시간을 내어 정보를 수집하고 잠재적 문제를 식별하면 가동 중지 시간을 최소화하고 위험을 줄이고 AWS로 원활하게 전환할 수 있는 포괄적인 마이그레이션 계획을 개발할 수 있습니다.

동원

AWS로 장애 조치 클러스터를 마이그레이션하는 동안 동원 단계에는 AWS로 마이그레이션할 클러스터를 준비하고 클러스터가 제대로 작동하는지 테스트하는 단계가 포함됩니다. 동원 단계에는 다음 단계가 포함됩니다.

1. 대상 환경 준비 - 이 단계에서는 장애 조치 클러스터를 호스팅하는 데 필요한 AWS 리소스를 생성합니다. 여기에는 Amazon VPC, 서브넷, 보안 그룹 및 기타 필요한 리소스 설정이 포함됩니다.
2. 소스 환경 준비 - 이 단계에서는 마이그레이션을 위해 기존 장애 조치 클러스터를 준비합니다. 여기에는 네트워크 구성 변경, 복제 구성 또는 필요한 소프트웨어 설치가 포함될 수 있습니다.
3. 클러스터 검증 - 소스 및 대상 환경을 모두 준비한 후 검증 테스트를 수행하여 클러스터가 제대로 작동하는지 확인할 수 있습니다. 여기에는 클러스터가 대상 환경으로 성공적으로 장애 조치할 수 있는지 확인하기 위한 일련의 테스트 실행이 포함됩니다.
4. 복제 링크 생성 - 검증 테스트 후 소스 환경과 대상 환경 간에 복제 링크를 생성할 수 있습니다. 이렇게 하면 소스 환경에 대한 모든 변경 사항이 대상 환경에 복제됩니다.
5. 복제 모니터링 - 복제 링크가 설정된 후 복제 프로세스를 모니터링하여 모든 변경 사항이 제대로 복제되고 있는지 확인합니다.
6. 클러스터 장애 조치 - 복제가 제대로 작동하는지 확인한 후 대상 환경으로 최종 장애 조치를 수행합니다. 여기에는 소스 환경에서 클러스터 서비스를 중지하고 대상 환경에서 클러스터 서비스를 시작하는 작업이 포함됩니다.
7. 장애 조치 테스트 - 장애 조치가 완료된 후 테스트를 수행하여 클러스터에서 실행되는 애플리케이션과 서비스가 새 환경에서 제대로 작동하는지 확인합니다.

마이그레이션

Microsoft 장애 조치 클러스터를 마이그레이션하는 작업은 성공적인 결과를 보장하기 위해 신중한 계획과 구현이 필요한 복잡한 프로세스일 수 있습니다. 프로덕션 환경을 변경하기 전에 기존 환경을 철저히 평가하고, 잠재적 문제를 식별하고, 테스트 및 검증이 포함된 포괄적인 마이그레이션 계획을 개발하는 것이 중요합니다. 마이그레이션 단계에서는 프로세스를 면밀히 모니터링하고 문제나 예상치 못

한 동작을 즉시 해결하는 것이 중요합니다. 원활한 마이그레이션 프로세스를 위해서는 IT 팀, 비즈니스 사용자, 공급업체 등 모든 이해관계자 간의 커뮤니케이션과 협업이 매우 중요합니다.

또한 마이그레이션이 장애 조치 클러스터에서 실행되는 타사 애플리케이션 또는 서비스에 미치는 영향을 고려하는 것도 중요합니다. 종속성을 식별하고 해당 애플리케이션을 철저하게 테스트하여 마이그레이션 후에도 예상대로 계속 작동하는지 확인합니다. 마이그레이션 단계의 또 다른 주요 측면은 마이그레이션 프로세스 중에 예상치 못한 문제나 장애가 발생할 경우에 대비하여 롤백 계획을 수립하는 것입니다. 이 계획에는 프로덕션 환경에 미치는 영향을 최소화하면서 마이그레이션을 되돌리고 원래 환경을 복원하는 단계가 포함되는 것이 이상적입니다.

마지막으로, 마이그레이션이 완료되고 장애 조치 클러스터가 새 환경에서 성공적으로 실행되고 나면 마이그레이션 후 검증 및 테스트를 수행하여 모든 것이 의도한 대로 작동하는지 확인하는 것이 중요합니다. 여기에는 성능 모니터링, 장애 조치 기능 검증, 모든 애플리케이션과 서비스의 정상 작동 여부 확인 등이 포함됩니다.

Microsoft 워크로드 모니터링

Microsoft 워크로드는 일반적으로 백엔드에서 SQL Server를 사용하여 데이터를 검색하고 유지합니다. 클라우드로 전환하는 과정에서 간단한 접근 방식을 사용하여 이러한 솔루션을 재호스팅하기로 결정하는 경우가 많습니다. lift-and-shift 이러한 애플리케이션이 Windows on Amazon EC2 플랫폼에서 호스팅되는 경우 네이티브 Windows 기반 도구를 사용하여 서버 수준에서 이러한 애플리케이션의 상태를 모니터링할 수 있습니다. 그러나 솔루션의 일부로 배포된 다양한 구성 요소와 서버를 전체적으로 파악하는 것은 어려운 일이지만 [Amazon CloudWatch Application Insights](#)는 이러한 문제점을 해결할 수 있습니다.

CloudWatch Application Insights는 AWS 워크로드용 애플리케이션 리소스를 설정하고 모니터링하는데 도움이 되는 클라우드 네이티브 모니터링 서비스입니다. 엔터프라이즈 고객은 다양한 워크로드를 처리하므로 다양한 소스의 텔레메트릭 데이터를 상호 연관시킬 수 있는 모니터링 서비스가 필요합니다. 기업 고객인 경우 CloudWatch Application Insights를 사용하면 리소스 검색을 자동화하고 다양한 리소스에서 애플리케이션을 생성할 수 있도록 지원하여 모니터링 설정의 복잡성을 피할 수 있습니다.

평가

애플리케이션의 성능과 백엔드 상태를 추적하는 것은 대부분의 조직에서 필수적입니다. 여정 중 언제 어디서 이상이 발견되었고 왜 발생했는지 알아야 합니다. 또한 시스템을 모니터링하고 유지 관리 비용을 줄여야 합니다.

CloudWatch 모니터링 요구 사항을 해결하는 데 도움이 될 수 있으며 CloudWatch Application Insights는 CloudWatch 지표, 경보 및 이벤트를 사용합니다. 를 사용하여 CloudWatch 여러 AWS

리소스에 대한 지표, 텔레메트리 및 로그의 모니터링 및 관리를 설정할 수 있습니다. [CloudWatch ServiceLensAmazon](#)은 애플리케이션 상태를 모니터링하는 데 필요한 모든 것을 제공하는 다양한 서비스를 제공합니다.

동원

CloudWatch Application Insights는 애플리케이션에 맞는 최적의 텔레메트리 지표와 로그를 빠르고 쉽게 설정하는 데 사용할 수 있는 로우클릭 사용자 인터페이스를 제공합니다. CloudWatch Application Insights는 특정 워크로드에 맞게 모니터를 조정하므로 특정 애플리케이션의 문제 징후를 지속적으로 분석할 수 있습니다. 또한 권장 워크로드 텔레메트리의 자동 구성 및 분석을 제공합니다. 몇 가지 예로 .NET CLR, 애플리케이션/웹 서버 기술에 대한 초당 요청 수, .NET 가비지 수집과 관련된 일반적인 문제 식별, SQL Server 백업 실패 등이 있습니다.

모니터링 솔루션을 온보딩하려는 경우 일반적으로 CPU, 메모리 및 기타 임계값 요구 사항을 이해하고 구성해야 합니다. 하지만 CloudWatch Application Insights는 이러한 리소스와 관련 지표를 자동으로 탐지합니다. 애플리케이션을 CloudWatch Application Insights에 추가하면 리소스를 스캔하고 애플리케이션 구성 요소에 대한 메트릭과 CloudWatch 로그온을 권장 및 구성합니다. 애플리케이션 구성 요소의 예로는 SQL Server 백엔드 데이터베이스 및 Microsoft IIS/웹 티어가 있습니다.

선택한 리소스 그룹에 따라 CloudWatch Application Insights는 각 구성 요소에 대한 모니터링을 자동으로 설정합니다. 계정 기반 애플리케이션 모니터링의 경우 계정에서 검색된 모든 리소스가 자동으로 추가됩니다. 또한 CloudWatch Application Insights의 리소스 탐지 기능을 활용할 수 있습니다.

CloudWatch Application Insights는 과거 데이터를 사용하여 지표 패턴을 분석하여 이상 징후를 탐지하고 애플리케이션, 운영 체제 및 인프라 로그에서 오류와 예외를 지속적으로 탐지합니다. 이 알고리즘은 분류 알고리즘과 기본 규칙을 조합하여 이러한 관찰 결과를 연결합니다. 그런 다음, 작업 우선 순위를 지정하는 데 도움이 되는 관련 관찰 및 문제 심각도 정보를 보여 주는 대시보드를 자동으로 생성합니다. 애플리케이션 지연 시간, SQL Server 백업 실패, 메모리 누수, 크고 잘못된 HTTP 요청, 취소된 I/O 작업 등과 같은 .NET 및 SQL 애플리케이션 스택의 일반적인 문제에 대해 CloudWatch Application Insights는 가능한 근본 원인과 해결 단계를 가리키는 추가 통찰력을 제공합니다.

[AWS Systems Manager와의](#) 통합 기능이 내장되어 OpsCenter 있으므로 관련 AWS Systems Manager Automation 문서를 실행하여 문제를 해결할 수 있습니다. CloudWatch Application Insights는 각 문제의 심각도 수준을 AWS Systems OpsCenter Manager로 전달하므로 지원 팀 내에서 작업의 우선 순위를 정하고 작업을 배정하는 데 도움이 됩니다.

마이그레이션

CloudWatch 애플리케이션 인사이트는 Amazon EC2 기반 Windows 에코시스템의 일부입니다. CloudWatch 애플리케이션 인사이트를 사용하여 모니터링하는 것은 이 서비스의 필수적인 부분입니다.

니다. AWS로 워크로드 마이그레이션을 시작한 후에는 CloudWatch 애플리케이션 인사이트를 통해 Microsoft 워크로드를 모니터링할 수 있습니다. 또한 CloudWatch Application Insights는 SAP, Java, Oracle, MySQL, PostgreSQL 및 기타 AWS 리소스 (서버리스 애플리케이션 지원 포함) 에 대한 지원을 포함하여 Microsoft 워크로드 이외의 지원을 제공합니다. CloudWatch 애플리케이션 인사이트를 시작하려면 Amazon CloudWatch 사용 설명서의 [설정하기](#)를 참조하십시오.

마이그레이션 도구, 프로그램 및 교육

이 섹션에서는 클라우드 마이그레이션을 지원하는 데 사용할 수 있는 AWS 및 파트너 도구, 팀에 클라우드로 마이그레이션하고 클라우드에서 운영하는 데 필요한 기술을 제공하는 데 사용할 수 있는 교육 기회, 마이그레이션 여정을 가속화하고 마이그레이션 비용을 줄이는 데 사용할 수 있는 주요 마이그레이션 프로그램을 간략하게 설명합니다.

도구

평가 도구

AWS Optimization and Licensing Assessment

[AWS Optimization and Licensing Assessment\(AWS OLA\)](#)를 사용하여 AWS에서 마이그레이션 및 라이선스 전략을 수립하는 것이 좋습니다. AWS OLA를 사용하여 Windows 환경을 평가할 수 있습니다. 평가를 통해 라이선스 비용의 절감 가능성을 파악하고 리소스를 보다 효율적으로 운영하는 방법을 찾을 수 있습니다.

AWS OLA는 신규 및 기존 고객을 위한 무약정 프로그램입니다. AWS OLA를 사용하여 실제 리소스 사용률, 타사 라이선스 및 애플리케이션 종속성을 기반으로 현재 온프레미스 및 클라우드 환경을 평가하고 최적화할 수 있습니다. [Enterprise Strategy Group](#)과 [Evolve Cloud Services](#)가 2022년에 실시한 제3자 연구에 따르면 AWS OLA를 통해 고객은 Microsoft SQL Server 라이선스 비용에서 평균 45%, Windows Server에서 77%를 절감할 수 있는 것으로 나타났습니다. 라이선스 비용은 AWS 클라우드에서 이러한 워크로드를 실제로 실행하는 비용의 3배에 해당하므로 잠재적인 절감 효과가 TCO에 상당한 영향을 미칠 수 있습니다.

AWS OLA는 배포 옵션을 모델링하는 보고서를 제공합니다. 이 결과는 AWS에서 제공하는 유연한 라이선스 옵션 전반에서 사용 가능한 비용 절감 효과를 탐색하는 데 도움이 될 수 있습니다. 또한 AWS OLA와 [AWS Migration Acceleration Program for Windows](#)를 함께 사용하여 클라우드 마이그레이션 중에 지원과 리소스를 받을 수 있습니다.

마이그레이션 전, 도중 또는 후에도 AWS OLA를 사용할 수 있습니다. 이 도구 기반 접근 방식은 실제 사용률 요구 사항을 결정하는 데 도움이 될 수 있습니다. AWS OLA는 각 워크로드에 대해 가장 저렴한 EC2 인스턴스 크기 및 유형을 권장합니다. 또한 온디맨드 인스턴스, 스팟 인스턴스, Amazon EC2 전용 호스트, 절감형 플랜 및 사용자 환경에 맞는 기타 옵션을 적절히 조합하여 찾는 데 도움이 됩니다. 또한 AWS OLA는 마이그레이션 계획, 방향별 비즈니스 사례 및 로드맵을 제공합니다.

라이선스 절감은 TCO의 중요한 부분이며, AWS OLA는 기존 라이선스 권한과 워크로드를 기반으로 기존 보유 라이선스 사용(BYOL) 또는 라이선스 포함 권장 사항을 제공하여 라이선스 비용을 줄이는 데 도움이 될 수 있습니다. AWS OLA는 애플리케이션에 대한 높은 성능을 유지하면서 더 적은 수의 라이선스가 필요하도록 인스턴스를 구성하여 라이선스를 최적화합니다. 또한 AWS OLA는 클라우드 라이선스와 온프레미스 라이선스의 차이점을 이해하는 데 도움이 됩니다. 이러한 지식을 바탕으로 라이선스 전략을 조정하여 향후 비용을 더욱 절감할 수 있습니다.

AWS OLA의 범위에는 다음과 같은 사용 사례가 포함됩니다.

- 방향별 비즈니스 사례, EC2 인스턴스 비용을 요약한 권장 사항, 실제 온프레미스 사용률 및 데이터를 기반으로 한 구성
- 호스트 수준 라이선스를 위한 전용 호스트 모델링
- SQL 인스턴스 최적화 및 통합을 위한 가상 CPU(vCPU) 감소
- 업계 평균을 기준으로 한 온프레미스 TCO 추정치
- AWS의 VMware Cloud 모델링

알림

2024년 4월 30일부터 VMware Cloud AWS on은 더 이상 채널 파트너에 의해 AWS 재판매되지 않습니다. 이 서비스는 Broadcom을 통해 계속 제공될 예정입니다. 자세한 내용은 AWS 담당자에게 문의하시기 바랍니다.

- Microsoft 라이선스 위치에 따른 권장 사항(라이선스 이동 및 잠재적 감소 관련)
- T3 전용 호스트의 라이선스 영향 모델링
- Amazon RDS의 SQL 및 Oracle 모델링, 에디션 최적화, Oracle Real Application Clusters(RAC) 및 Oracle Exadata 분석
- SQL 고가용성 라이선스 영향에 대한 액티브 및 패시브 모델링
- 현대화 평가

AWS는 내부 [마이그레이션 평가기](#)나 타사 공급업체 또는 자격을 갖춘 AWS OLA 마이그레이션 파트너의 신뢰할 수 있는 도구를 사용하여 광범위한 검색을 수행하거나, 기존 인벤토리가 있는 경우 내보내기를 안전하게 업로드합니다. 사용되는 도구는 특정 요구 사항에 따라 다릅니다. AWS는 검색 도구 출력을 사용하고 이를 타사 라이선스 컨설턴트의 전문가 권장 사항과 결합하여 신뢰할 수 있는 최적화된 TCO를 제공합니다.

자세한 정보는 다음 자료를 참조하십시오.

- [AWS Optimization and Licensing Assessment\(AWS 설명서\)](#)
- [AWS용 Windows 워크로드 최적화 - AWS 온라인 테크 토크 \(\) YouTube](#)
- [Run Optimization and Licensing Assessment\(AWS 설명서\)](#)

Migration Hub Strategy Recommendations

[Migration Hub Strategy Recommendations](#)는 애플리케이션의 실행 가능한 변환 경로에 대한 마이그레이션 및 현대화 전략 권장 사항을 제공하여 마이그레이션 및 현대화 이니셔티브를 계획하는 데 도움이 됩니다. Strategy Recommendations는 서버 인벤토리와 런타임 환경을 분석합니다. 또한 소스 코드와 데이터베이스 분석을 수행할 수 있습니다. Strategy Recommendations는 이 분석을 비즈니스 목표 및 제공한 애플리케이션 및 데이터베이스의 변환 기본 설정과 결합하여 다음을 권장합니다.

- 각 애플리케이션을 위한 가장 효과적인 마이그레이션 전략
- 사용할 수 있는 마이그레이션 및 현대화 도구 또는 프로그램
- 특정 옵션에 대해 해결해야 할 애플리케이션 비호환성 및 안티 패턴

Strategy Recommendations는 관련 배포 대상, 도구 및 프로그램을 사용하여 리호스팅, 리플랫폼 및 리팩터링을 위한 마이그레이션 및 현대화 전략을 권장합니다. 예를 들어, Strategy Recommendations는 Application Migration Service를 사용하여 Amazon EC2에서 리호스팅하는 등의 간단한 옵션을 권장할 수 있습니다. 보다 최적화된 권장 사항에는 AWS App2Container를 사용하여 컨테이너로 리플랫폼이나 .NET Core 및 PostgreSQL과 같은 오픈 소스 기술로 리팩터링이 포함될 수 있습니다.

Strategy Recommendations를 사용하려면 Migration Hub Strategy Recommendations 사용 설명서의 [Getting started with Strategy Recommendations](#) 지침을 따르세요.

마이그레이션 검증기 툴킷 모듈 PowerShell

[마이그레이션 검증기 툴킷 PowerShell 모듈](#)을 사용하여 Microsoft 워크로드를 검색하고 AWS로 마이그레이션하는 것이 좋습니다. 이 모듈은 Microsoft 워크로드와 관련된 일반적인 작업에 대해 여러 검사와 검증을 수행하여 작동합니다. Migration Validator Toolkit PowerShell 모듈을 사용하면 조직에서 Microsoft 워크로드에서 실행 중인 응용 프로그램 및 서비스를 검색하는 데 드는 시간과 노력을 줄일 수 있습니다. 또한 이 모듈을 사용하면 워크로드의 구성을 식별하여 해당 구성이 AWS에서 지원되는지 확인할 수 있습니다. 이 모듈에서는 마이그레이션 전, 도중, 후에 잘못된 구성을 방지할 수 있도록 다음 단계와 완화 조치에 대한 권장 사항도 제공합니다.

AWS Cloud Readiness Assessment

[AWS Cloud Readiness Assessment](#)를 사용하여 클라우드로의 이동 아이디어를 AWS Professional Services 모범 사례를 따르는 세부 계획으로 전환하는 것이 좋습니다. AWS Cloud Readiness Assessment를 사용하여 조직의 규모와 상관없이 클라우드 도입 및 엔터프라이즈 클라우드 마이그레이션을 위한 효율적이고 효과적인 계획을 개발할 수 있습니다. 16개의 질문으로 구성된 이 온라인 설문 조사 및 평가 보고서는 비즈니스, 사람, 프로세스, 플랫폼, 운영 및 보안을 포함한 6가지 관점에서 클라우드 마이그레이션 준비 상태를 자세히 설명합니다.

평가를 완료한 후 연락처 세부 정보를 제공하여 준비 상태와 이를 개선하기 위해 취할 수 있는 조치를 차트로 보여주는 사용자 지정 클라우드 마이그레이션 평가를 다운로드할 수 있습니다. 요약 보고서에는 준비 상태 점수를 높이는 데 도움이 되는 자세한 점수 정보 및 리소스가 포함된 히트맵 및 레이더 차트가 포함되어 있습니다. 이 요약 보고서는 계획을 세우고 이해관계자와 소통하는 데 도움이 될 수 있습니다. 샘플 평가 보고서는 [AWS Cloud Adoption Readiness Assessment Report](#)를 참조하세요. 평가를 받으려면 [AWS Cloud Adoption Readiness Assessment](#)로 이동하세요.

마이그레이션 도구

AWS Migration Hub

[AWS Migration Hub](#)는 AWS로의 마이그레이션을 평가, 계획 및 추적하기 위해 서버 및 애플리케이션 인벤토리 데이터를 수집할 수 있는 중앙 위치를 제공합니다. 또한 Migration Hub는 마이그레이션 후 애플리케이션 현대화를 가속화하는 데 도움이 될 수 있습니다. Migration Hub 네트워크 시각화를 사용하면 서버와 그 종속성을 신속하게 식별하고, 서버의 역할을 식별하고, 서버를 애플리케이션으로 그룹화하여 마이그레이션 계획을 가속화할 수 있습니다. 네트워크 시각화를 사용하려면 [AWS Application Discovery Agent\(Discovery Agent\)](#)를 설치한 다음, 데이터 수집을 시작합니다.

AWS Migration Hub Orchestrator

[AWS Migration Hub Orchestrator](#)는 애플리케이션 마이그레이션을 가속화하여 마이그레이션에 소요되는 시간과 노력을 줄이는 데 도움이 됩니다. 사전 정의된 워크플로 템플릿을 사용하여 마이그레이션 워크플로를 쉽게 생성하고, 특정 요구 사항에 맞게 워크플로를 사용자 지정하고, 마이그레이션 단계를 자동화하고, 처음부터 끝까지 한 곳에서 마이그레이션 진행 상황을 추적할 수 있습니다. Orchestrator는 다음을 지원합니다.

- SAP HANA 데이터베이스를 통한 SAP 기반 애플리케이션 마이그레이션 NetWeaver
- Amazon EC2에 모든 애플리케이션 리호스팅
- Amazon EC2에 SQL Server 데이터베이스 리호스팅
- Amazon RDS로 SQL Server 데이터베이스 리플랫폼

- Amazon EC2용 AMI로 OVA(Open Virtual Appliance) 또는 VMDK(가상 머신 디스크)의 VM 이미지 가져오기

AWS Migration Hub 대시보드

[Migration Hub 대시보드](#)에는 리호스팅 및 리플랫폼 마이그레이션의 최신 상태와 지표가 표시됩니다. 대시보드를 사용하여 마이그레이션 진행 상황을 빠르게 파악하고 문제를 식별하여 해결할 수 있습니다. Migration Hub를 사용하면 마이그레이션 도구가 지원하는 모든 AWS 리전으로의 마이그레이션 상태를 추적할 수 있습니다. 마이그레이션하는 리전에 관계없이 통합 도구를 사용하면 마이그레이션 상태가 Migration Hub에 표시됩니다.

AWS Application Migration Service

[AWS Application Migration Service](#)는 AWS에서 기본적으로 실행되도록 소스 서버를 자동으로 변환하여 시간이 많이 걸리고 오류가 발생하기 쉬운 수동 프로세스를 최소화합니다. 또한 기본 제공 및 사용자 지정 최적화 옵션을 통해 애플리케이션 현대화를 간소화합니다. 다음은 Application Migration Service의 사용 사례입니다.

- 물리적 서버 또는 VMware vSphere, Microsoft Hyper-V 및 기타 온프레미스 인프라에서 실행되는 SAP, Oracle 및 SQL Server와 같은 온프레미스 워크로드
- 다른 퍼블릭 클라우드에서 AWS로 실행되는 클라우드 기반 워크로드

Application Migration Service를 사용하여 비용을 절감하고, 가용성을 높이고, 혁신을 촉진하는 200개 이상의 서비스에 액세스할 수 있습니다. 또한 이를 사용하여 AWS 리전, 가용 영역 또는 계정 간에 EC2 워크로드를 더 쉽게 이동하여 비즈니스, 복원력 및 규정 준수 요구 사항을 충족할 수 있습니다.

또는 현대화 전략으로 사용자 지정 현대화 작업을 적용하거나 리전 간 재해 복구, CentOS 변환, SUSE Linux 구독 전환과 같은 내장된 작업을 선택하여 애플리케이션을 최적화할 수 있습니다.

AWS Database Migration Service

[AWS Database Migration Service\(AWS DMS\)](#)는 데이터베이스 및 분석 워크로드를 AWS로 빠르고 안전하게 이동하고 가동 중지 시간을 최소화하고 데이터 손실을 방지하는 관리형 마이그레이션 및 복제 서비스입니다. AWS DMS는 SQL Server를 비롯한 20개 이상의 데이터베이스 및 분석 엔진 간 마이그레이션을 지원합니다.

AWS DMS를 사용하면 관리형 데이터베이스 모델을 사용하여 간소화된 마이그레이션 프로세스를 통해 기존 또는 온프레미스 데이터베이스에서 관리형 클라우드 서비스로 마이그레이션할 수 있으므로 개발자에게 혁신에 필요한 시간을 확보할 수 있습니다. 또한, AWS DMS를 사용하여 라이선스 비용에

서 벗어나 비즈니스 성장을 가속화하고, 특별히 구축된 데이터베이스를 사용하여 1/10의 비용으로 모든 사용 사례에 맞게 혁신하고 더 빠르게 구축할 수 있습니다.

AWS DMS를 사용하여 다음을 수행할 수도 있습니다.

- 백업 파일 복제
- 업무상 중요한 데이터베이스와 데이터 스토어를 중복하여 가동 중지 시간과 데이터 손실을 최소화합니다.
- 데이터 레이크를 구축하여 데이터 스토어의 변경 데이터를 실시간으로 처리합니다.
- 데이터 레이크를 구축하여 데이터 마트를 통합합니다.
- 데이터 스토어의 변경 데이터를 실시간으로 처리합니다.

마이그레이션 파트너 도구

CloudBasix

[CloudBasix](#) 클라우드 네이티브 워크로드 최적화 및 데이터 통합 제품을 만듭니다. CloudBasix의 주력 제품인 [CLOUDBASIX for RDS SQL Server Read Replicas and Disaster Recovery\(DR\)](#)를 사용하여 다음을 활성화할 수 있습니다.

- 리전 내 읽기 전용 복제본
- 교차 리전 DR
- Azure에서 AWS로 클라우드 간 재해 복구
- AI 기반 데이터 레이크 및 데이터 하우스
- Amazon Redshift 및 Snowflake를 위한 통합

관리 도구

Amazon CloudWatch 애플리케이션 인사이트

[Amazon CloudWatch Application Insights](#)는 애플리케이션과 기본 AWS 리소스의 관찰성을 용이하게 합니다. 이를 통해 애플리케이션 리소스에 가장 적합한 모니터를 설정하여 데이터를 지속적으로 분석하여 애플리케이션 관련 문제의 징후가 있는지 확인할 수 있습니다. CloudWatch Amazon SageMaker 및 기타 AWS 기술을 기반으로 하는 Application Insights는 모니터링되는 애플리케이션의 잠재적 문제를 보여주는 자동화된 대시보드를 제공합니다. 이를 통해 애플리케이션 및 인프라와 관련된 지속적인 문제를 신속하게 격리할 수 있습니다.

애플리케이션을 CloudWatch Application Insights에 추가하면 애플리케이션의 리소스를 스캔하고 애플리케이션 구성 요소에 대한 메트릭과 CloudWatch 로그온을 권장 및 구성합니다. 응용 프로그램 구성 요소의 예로는 SQL Server 백엔드 데이터베이스와 Microsoft IIS 또는 웹 티어가 있습니다. CloudWatch Application Insights는 기간별 데이터를 사용하여 지표 패턴을 분석하여 이상 징후를 탐지하고 응용 프로그램, 운영 체제 및 인프라 로그에서 오류와 예외를 지속적으로 탐지합니다. 이 알고리즘은 분류 알고리즘과 기본 규칙을 조합하여 이러한 관찰 결과를 연결합니다. 그런 다음 CloudWatch Application Insights는 관련 관찰 및 문제 심각도 정보를 보여주는 대시보드를 자동으로 생성하여 조치의 우선 순위를 정하는 데 도움이 됩니다. 애플리케이션 대기 시간, SQL Server 백업 실패, 메모리 누수, 대용량 HTTP 요청, 취소된 I/O 작업과 같은 일반적인 .NET 및 SQL 애플리케이션 스택 문제에 대해 가능한 근본 원인 및 해결 단계를 나타내는 추가 인사이트를 제공합니다. [AWS Systems OpsCenter Manager와의](#) 통합 기능이 내장되어 있으므로 관련 Systems Manager 자동화 문서를 실행하여 문제를 해결할 수 있습니다.

AWS License Manager

[AWS License Manager](#)를 사용하면 AWS 및 온프레미스 환경 전반에서 Microsoft, SAP, Oracle, IBM 등 공급업체의 소프트웨어 라이선스를 보다 쉽게 관리할 수 있습니다. License Manager를 사용하면 라이선스 유형을 전환하고 기존 라이선스의 검색, 추적 및 보고를 자동화하여 라이선스 관리를 간소화할 수 있습니다. 또한 자동 할당, 릴리스 및 복구 기능을 갖춘 전용 호스트 컬렉션을 단일 엔티티로 관리하여 Windows BYOL 환경을 단순화할 수 있습니다. 최종 사용자를 위한 AWS 계정 전반의 소프트웨어 사용 권한 및 워크로드의 배포 및 활성화를 자동화하여 여러 계정의 마켓플레이스 라이선스를 처리할 수도 있습니다.

AWS Backup

[AWS Backup](#)은 대규모 데이터 보호를 간소화하는 비용 효율적인 완전관리형 정책 기반 서비스입니다. AWS Backup을 사용하여 AWS 서비스 전반의 버킷, 볼륨, 데이터베이스, 파일 시스템 등의 주요 데이터 스토어에 대한 클라우드 네이티브 백업을 만들 수 있습니다. AWS Backup은 VMware 워크로드 및 AWS Storage Gateway 볼륨과 같은 하이브리드 환경에서 실행되는 애플리케이션에 대한 데이터 보호 관리를 제공하여 데이터 보호를 중앙 집중화합니다. 또한 조직의 AWS 계정, 리소스 및 리전 전반에 걸쳐 백업 활동을 구성, 관리 및 통제하기 위한 정책을 중앙에서 관리할 수 있습니다.

AWS Systems Manager Fleet Manager

AWS Systems Manager의 기능인 [Fleet Manager](#)는 AWS 또는 온프레미스에서 실행되는 관리형 노드를 원격으로 관리하는 데 도움이 되는 통합 사용자 인터페이스(UI) 환경입니다. Fleet Manager를 사용하면 하나의 콘솔에서 전체 서버 플릿의 상태 및 성능 상태를 볼 수 있습니다. 또한 개별 노드에서 데이터를 수집하여 콘솔에서 일반적인 문제 해결 및 관리 태스크를 수행할 수 있습니다. 여기에는 원격 데스크톱 프로토콜(RDP)을 사용하여 Windows 인스턴스에 연결, 폴더 및 파일 내용 보기, Windows 레지

스트리 관리, 운영 체제 사용자 관리 등이 포함됩니다. 노드 플릿 또는 Amazon ECS 클러스터 관리를 중앙 집중화하려는 경우 Fleet Manager를 사용할 수 있습니다.

프로그램

AWS Migration Acceleration Program

[AWS Migration Acceleration Program\(MAP\)](#)은 수천 명의 기업 고객을 클라우드로 마이그레이션한 AWS의 경험을 기반으로 하는 포괄적이고 검증된 클라우드 마이그레이션 프로그램입니다. 엔터프라이즈 마이그레이션은 복잡하고 시간이 많이 소요될 수 있지만 MAP는 결과 기반 방법론을 통해 클라우드 마이그레이션 및 현대화 여정을 가속화하는 데 도움이 될 수 있습니다.

MAP는 비용을 절감하고 구현을 자동화 및 가속화하는 도구, 맞춤형 교육 방식 및 콘텐츠, AWS 파트너 네트워크 파트너의 전문 지식, 글로벌 파트너 커뮤니티, AWS 투자를 제공합니다. 또한 MAP는 검증된 3단계 프레임워크를 사용하여 마이그레이션 목표를 달성하는 데 도움이 됩니다. MAP를 통해 위험을 줄이고, 생산성을 높이고, 운영 탄력성을 개선하고, 초기 마이그레이션 비용을 상쇄하는 동시에 강력한 AWS 클라우드 기반을 구축할 수 있습니다. 클라우드의 성능, 보안 및 신뢰성을 활용할 수도 있습니다.

AWS Windows Migration Accelerator

[AWS Windows Migration Accelerator](#)는 [Application Migration Service](#)를 사용하여 Windows 서버의 마이그레이션을 가속화할 때 AWS 프로모션 크레딧으로 마이그레이션 비용을 줄이는 데 도움이 됩니다. AWS Windows Migration Accelerator 인센티브는 다른 합의된 판매 인센티브 및 프로모션 프로그램과 함께 적용할 수 있습니다. Application Migration Service를 사용하여 한 달에 서버를 40대 이상 (Windows 서버 최소 15대 포함) AWS로 마이그레이션하는 경우 2023년 12월 31일까지 Windows 서버당 200 USD의 AWS 프로모션 크레딧을 받을 수 있습니다. 한 달에 서버를 80대 이상 (Windows 서버 최소 25대 포함) 마이그레이션하는 경우 Application Migration Service를 사용하여 AWS로 마이그레이션하는 Windows 서버당 AWS 프로모션 크레딧이 250 USD로 늘어납니다. 마이그레이션된 서버는 AWS 외부 위치에서 마이그레이션하고 마이그레이션 후 최소 4주 동안 AWS에서 계속 실행해야 합니다.

AWS Migration Acceleration Program for Windows

기존 AWS MAP 프로그램의 확장인 [AWS Migration Acceleration Program \(MAP\) for Windows](#)는 조직이 AWS 서비스, 모범 사례, 도구 및 인센티브를 통해 마이그레이션 목표를 더 빨리 달성할 수 있도록 설계되었습니다. AWS는 3단계 접근 방식을 사용하여 클라우드로 마이그레이션하는 데 따른 불확실성, 복잡성 및 비용을 줄이는 데 도움이 됩니다. 또한 MAP를 사용하면 Linux, Aurora, 컨테이너 기반 서비스 및 Lambda에서 실행되는 SQL Server와 같은 클라우드 솔루션으로 Windows Server 및 SQL Server 워크로드의 현재 및 레거시 버전을 현대화하여 비용을 절감할 수 있습니다. 클라우드 네이티브 또는 오픈 소스 솔루션을 사용하면 상용 라이선스의 높은 비용 부담에서 벗어날 수 있습니다.

AWS 인프라 이벤트 관리

[AWS 인프라 이벤트 관리\(IEM\)](#)는 쇼핑 홀리데이, 제품 출시, 마이그레이션과 같은 계획된 이벤트를 준비하고 구현하는 동안 아키텍처 및 크기 조정 지침과 운영 지원을 제공합니다. 이러한 이벤트에 대해 IEM은 운영 준비 상태를 평가하고, 위험을 식별 및 완화하고, AWS 전문가와 함께 이벤트를 자신 있게 구현할 수 있도록 도와줍니다. 이 프로그램은 Enterprise Support 플랜에 포함되어 있으며 비즈니스 지원 고객은 추가 비용을 지불하고 이용할 수 있습니다.

AWS 전문가는 다음을 수행하는 데 도움이 되는 규범적이고 단계적인 접근 방식을 사용하여 계획된 이벤트에 대한 아키텍처 및 운영 지침을 제공하기 위해 매우 집중적인 참여를 주도합니다.

- 성공 기준 및 원하는 비즈니스 성과 이해
- AWS 환경의 준비 상태 평가하고, 위험 식별 및 완화 지원, 계획 문서화
- AWS 전문가와 함께 자신 있게 이벤트 주최
- 이벤트 후 결과를 분석하고 서비스를 일반 운영 수준으로 확장하여 다음 이벤트 계획에 집중

학습

자기 주도형, 대화형 및 강의실 교육

AWS는 마이그레이션 여정을 지원하기 위해 디지털 교육과 강의실 교육을 모두 제공합니다. AWS 전문가가 만든 수백 개의 자습형 디지털 교육 과정으로 학습을 시작할 수 있습니다. 그런 다음, [AWS Skill Builder](#)로 대화형 교육을 이수하여 실무 기술을 습득할 수 있습니다. 강의실 교육을 통해 질문을 하고, 직접 솔루션을 살펴보고, 심층적인 기술 지식을 갖춘 AWS 공인 강사로부터 피드백을 받을 수 있습니다. 자세한 내용은 [AWS Training and Certification](#) 서비스를 참조하세요.

AWS 파트너 교육

또한 AWS 파트너는 EdX 및 Coursera와 같은 최고의 온라인 학습 플랫폼에서 AWS 클라우드 기초부터 기계 학습에 이르기까지 다양한 주제를 다루는 자습형 과정으로 디지털 교육을 제공합니다. 자세한 내용은 [AWS 파트너 교육 및 자격증](#) 서비스를 참조하세요. 역할 및 솔루션별로 인증을 받을 수 있습니다. 예를 들어 역할에는 클라우드 전문가, 솔루션스 아키텍트, 개발자 및 SysOps 관리자가 포함됩니다. 솔루션에는 고급 네트워킹, 데이터 분석, 데이터베이스, 기계 학습, 보안, 스토리지 등이 포함됩니다.

AWS에서의 Microsoft 라이선스

이 섹션에서는 AWS에서 Microsoft 라이선싱이 어떻게 적용되는지 설명하고, AWS에 Microsoft 워크로드를 배포하기 위한 라이선스 모범 사례와 전략을 제시하며, 비용을 최적화하면서 Microsoft의 라이선스 조건을 계속 준수하는 데 도움을 줍니다. 라이선스가 마이그레이션 비용에 미치는 영향으로 인해, Microsoft 라이선스와 기존 보유 라이선스 사용(BYOL) 옵션은 AWS 고객이 이용할 수 있는 배포 옵션에 영향을 미치는 경우가 많습니다. 그렇기 때문에 마이그레이션 프로세스를 시작하기 전에 라이선스의 작동 방식을 이해하는 것이 중요합니다.

평가

AWS로 마이그레이션할 Microsoft 워크로드를 평가할 때는 라이선스 요구 사항을 고려하는 것이 중요합니다. Microsoft 워크로드의 경우 [AWS Optimization and Licensing Assessment\(AWS OLA\)](#)를 활용하여 온프레미스 또는 클라우드 워크로드를 평가하고 AWS에서 워크로드를 실행하기 위한 적절한 규모의 최적화된 로드맵을 구축하는 것이 좋습니다. AWS OLA는 워크로드에 가장 적합한 EC2 인스턴스를 제안할 뿐만 아니라 Microsoft 라이선스 상태도 검토합니다. 결과적으로, 컴퓨팅 및 라이선스 비용을 절감할 수 있는 최선의 경로가 추천됩니다. AWS OLA는 신규 및 기존 고객이 이용할 수 있으며 전액 자금 지원 대상이며 고객 의무는 없습니다. 자세한 내용은 [AWS OLA 팀](#)에 문의하시기 바랍니다.

현재 AWS OLA를 선택할 수 없더라도 AWS에서 Microsoft 라이선스가 작동하는 방식을 이해하는 것은 중요합니다. BYOL을 활용하려면 Microsoft 라이선스 구매 담당자에게 Microsoft 라이선스 정책(MLS)의 업데이트된 사본을 요청하는 것이 좋습니다. 이 정보를 사용하여 보유한 라이선스와 경우에 따라 구매 날짜 및 SA 수량을 검토할 수 있습니다. MLS와 관련하여 도움이 필요하다면 AWS 담당자에게 문의하세요. 담당자가 Microsoft 전문가와 연결해 드릴 수 있습니다.

Microsoft 제품마다 라이선스 요구 사항이 다르므로 배포한 Microsoft 제품이 무엇인지 명확하게 파악하는 것이 중요합니다. AWS는 다양한 Microsoft 제품의 요구 사항을 충족하기 위해 다양한 옵션을 제공합니다. 여기에는 라이선스 이동이 적용되는 제품을 위한 Amazon EC2의 공유/기본 테넌시와 라이선스 이동이 적용되지 않는 제품을 위한 전용 옵션이 포함됩니다. 또한 AWS에는 라이선스 비용이 Amazon EC2 컴퓨팅 비용에 포함되는 라이선스 포함 옵션도 있습니다. AWS로 마이그레이션할 때 혼합 라이선스 모델의 이점을 누릴 수 있습니다. 혼합 라이선스 모델은 공유 테넌시 EC2 인스턴스를 모든 또는 일부 라이선스 포함 옵션과 함께 사용하는 것을 말합니다. 혼합 라이선스 모델은 가변적인 워크로드에 적합하며, 안정적이고 예측 가능한 워크로드를 위해 전용 EC2 옵션을 사용하는 경우, 특히 Windows Server Datacenter 또는 SQL Server Enterprise BYOL을 사용하는 경우에 적합합니다.

Microsoft의 볼륨 라이선스 프로그램을 통해 구입한 제품의 최신 Microsoft 라이선스 약관에 대한 자세한 내용은 [Microsoft 제품 약관](#) 사이트를 참조하세요.

라이선스 포함 옵션

라이선스 포함이란 컴퓨팅 비용에 라이선스 비용이 포함된 Amazon EC2 인스턴스를 말합니다. Microsoft 서버 워크로드의 경우, AWS는 현재 Windows Server([Amazon EC2](#), [Amazon EC2 Dedicated Hosts](#), [Amazon EC2 Dedicated Instances](#), [AWS Outposts](#))와 SQL Server Enterprise, Standard 및 Web 에디션([Amazon EC2](#))을 제공합니다. 이러한 서버 라이선스는 라이선스 포함 EC2 인스턴스의 혜택으로 pay-as-you-go 모델과 함께 초당 vCPU당 제공됩니다. EC2 인스턴스가 중지될 예정이거나 수요에 따라 스케일 업 또는 스케일 다운되는 경우, 인스턴스 실행 시간에 대한 라이선스 비용만 지불하면 됩니다. 온디맨드 요금제에는 장기 약정이 없으므로 향후 현대화 계획에 이상적입니다.

라이선스 포함은 지원되는 모든 버전에서 Amazon Machine Image(AMI)를 사용할 수 있는 현재 버전과 레거시 버전에 사용 가능합니다. Windows Server 2008 또는 SQL Server 2012와 같은 End-of-support 버전에서도 라이선스가 포함된 상태로 라이선스를 받을 수 있지만 미디어를 직접 가져와야 합니다.

라이선스 포함 옵션의 경우 소프트웨어 업그레이드 요금이 부과되지 않습니다. Microsoft에서 새 버전의 제품을 출시하는 즉시 Amazon EC2 콘솔에서 현재 라이선스 포함 비용을 초과하는 추가 비용 없이 새 버전을 즉시 사용할 수 있습니다. 무엇보다, AWS가 라이선스 포함 EC2 인스턴스의 라이선스 규정 준수를 책임진다는 것이 중요합니다. 라이선스 규정 준수는 복잡하고 어려울 수 있으므로 이를 통해 많은 시간과 노력을 줄일 수 있습니다.

SQL Server 라이선스 포함 옵션은 클라이언트 액세스 라이선스(CAL)가 필요 없는 코어 기반 라이선스를 제공합니다. CAL을 계산하거나 라이선스를 취득할 필요 없이, 무제한의 사용자가 라이선스 포함 Windows Server EC2 인스턴스에 액세스할 수 있습니다. Windows Server 라이선스 포함 EC2 인스턴스에는 관리 목적으로만 사용할 수 있는 두 개의 Microsoft 원격 데스크톱 연결도 포함됩니다. 추가 Microsoft 원격 데스크톱 연결이 필요한 경우 소프트웨어 보증(SA)을 통해 Microsoft로부터 원격 데스크톱 서비스 사용자 CAL을 구매하고 라이선스 이동 혜택을 통해 AWS로 가져올 수 있습니다.

AWS는 일부 사용자 기반 라이선스 포함 옵션도 제공합니다. Visual Studio 2022 Enterprise 및 Professional 에디션([Amazon EC2](#) 및 [Lambda](#)) 및 Office LTSC Professional Plus 2021([Amazon EC2](#))은 사용자당 월 요금이 부과됩니다. 여기에는 각 사용자별 Microsoft 원격 데스크톱 연결이 포함됩니다. [Amazon은 WorkSpaces](#) 또한 오피스 프로페셔널 플러스 2016 또는 2019를 애드온으로 제공하며, 사용자당 매월 요금이 부과됩니다.

AWS는 Microsoft 워크로드에 대해 다음과 같은 라이선스 포함 옵션을 제공합니다.

제품	가용성	사용 가능한 버전
Windows Server	EC2, EC2 전용 인스턴스, EC2 전용 호스트, Outposts	모두*

SQL Server Enterprise	EC2	모두*
SQL Server Standard	EC2	모두*
SQL Server Web**	EC2	모두*
Visual Studio Enterprise	EC2, Lambda	2022
Visual Studio Professional	EC2, Lambda	2022년
Office Professional Plus	WorkSpaces	2019, 2016
Office Professional Plus LTSC	EC2	2021

*Out-of-support 및 지원되는 버전에는 자체 미디어가 필요합니다.

**SQL Server Web 에디션은 Microsoft의 라이선스 약관에 따라 사용 사례가 제한됩니다. SQL Server Web 에디션은 퍼블릭 및 인터넷으로 액세스 가능한 웹 페이지, 웹 사이트, 웹 애플리케이션 및 웹 서비스를 지원하는 데에만 사용할 수 있습니다. line-of-business 응용 프로그램 (예: 고객 관계 관리, 기업 자원 관리 및 기타 유사한 응용 프로그램) 을 지원하는 데 사용할 수 없습니다.

라이선스 포함 옵션은 다양한 워크로드에 최적의 옵션이 됩니다. 예를 들어 대부분의 경우 워크로드를 실행할 필요가 없거나, 워크로드를 자주 스케일 업 및 스케일 다운해야 하는 경우가 여기에 해당합니다.

BYOL 옵션

기존 보유 라이선스 사용(BYOL) 모델을 사용하는 것은 온프레미스 소프트웨어에 대한 기존 투자를 활용하면서 AWS 클라우드의 효율성이라는 이점까지 얻는 좋은 방법입니다. BYOL을 사용하면 이전 소프트웨어 버전 및 구매의 수명 주기를 연장하고 AWS에서 제공하지 않는 제품을 라이선스가 포함되어 있는 상태로 배포할 수 있습니다. 자체 라이선스를 사용할 때마다 자체 미디어도 사용해야 합니다. 즉, Amazon에서 제공한 AMI를 사용하는 대신 자체 미디어로 자체 AMI를 생성해야 합니다. [VM Import/Export](#) 도구는 무료로 사용할 수 있으며 자체 AMI를 생성할 수 있습니다. 또는 [Application Migration Service](#)를 사용하여 자체 미디어 및 AMI를 생성할 수 있습니다.

소프트웨어 보증을 통한 Microsoft 제품 라이선스 이동

AWS는 [공인 모빌리티 파트너](#)이므로 활성 SA의 적용을 받는 라이선스 이동 옵션이 포함된 모든 Microsoft 제품을 공유 또는 전용 테넌트 환경에서 AWS로 가져올 수 있습니다. SA를 통해 라이선스 이동이 가능한 제품에는 SQL Server, SharePoint 서버, Exchange 서버, 프로젝트 서버, 비즈니스용

Skype 서버, BizTalk 서버, 원격 데스크톱 서비스 사용자 CAL, 시스템 센터 서버 등이 있습니다. 라이선스 이동이 적용되지 않는 Microsoft 제품은 Microsoft의 2019년 10월 1일자 [라이선스 변경 사항](#)의 영향을 받지 않습니다. 따라서 라이선스 이동이 적용되는 제품에는 구매 날짜 또는 버전 제한이 없습니다. 라이선스에 활성 SA가 있는 한 BYOL에서 AWS로 전환할 수 있습니다. 예를 들어 활성 SA가 있는 SQL Server 2022 라이선스는 SA가 유지되는 한 공유 테넌시(기본) EC2 인스턴스(전용 인스턴스가 필요하지 않음)로 전환할 수 있습니다.

SA를 통한 라이선스 이동이 적용되는 제품은 System Center Server를 제외하고 가상화된 온프레미스 환경과 동일한 방식으로 AWS에서 라이선스를 받습니다. System Center Server 라이선스에는 AWS 클라우드로 가져올 때 적용되는 특수한 라이선스 계산 방식이 적용됩니다. System Center Server Datacenter 에디션의 코어 16개당 최대 10개의 EC2 인스턴스(크기 제한 없음)를 관리할 수 있습니다. System Center Server Standard 에디션의 코어 16개당 최대 2개의 EC2 인스턴스(크기 제한 없음)를 관리할 수 있습니다.

SQL Server는 AWS에서 가장 일반적으로 사용되는 라이선스 이동 적용 제품입니다. 활성 SA 또는 구독 라이선스가 포함된 SQL Server Core 라이선스(Cloud Solution Provider 또는 CSP 프로그램을 통해 구매한 라이선스 제외)는 공유 테넌시(기본) EC2 인스턴스에서 vCPU별로 라이선스가 부여되며, EC2 인스턴스당 vCPU 4개라는 최소 Microsoft 라이선스 요구 사항이 적용됩니다. 활성 SA가 포함된 SQL Server/CAL 라이선스는 EC2 인스턴스당 서버 라이선스 1개로 라이선스가 부여됩니다. 또한 액세스 권한이 있는 모든 사용자 또는 디바이스에 해당 CAL이 할당되어 있어야 합니다. SQL Server는 활성 SA 및 구독을 통해 패시브 장애 조치의 이점도 제공합니다. EC2에서 라이선스가 부여된 모든 액티브 SQL Server에 대해 패시브 인스턴스의 SQL Server 부분에 라이선스를 부여할 필요 없이 EC2의 보조 패시브 SQL Server 인스턴스를 사용할 수 있습니다. 자세한 내용은 Microsoft 웹 사이트에서 [Microsoft SQL Server 2022 라이선스 가이드](#)(다운로드 가능한 PDF)를 참조하세요.

AWS는 [공인 모빌리티 파트너](#)입니다(다운로드 가능한 PDF). 라이선스 이동이 적용되는 Microsoft 제품을 AWS에서 사용하는 경우 [라이선스 이동](#) 확인 양식을 작성하여 Microsoft에 제출해야 합니다. 이 양식은 다음과 같은 내용을 요구하는 간단한 Microsoft Word 문서입니다.

- 이름 및 연락처 정보
- Microsoft 계약 번호
- 이용 중인 클라우드 파트너
- 라이선스 이동을 통해 사용할 제품
- 사용할 라이선스 수

제품을 AWS로 가져온 후 10일 이내에 Microsoft에 직접 또는 Microsoft 리셀러를 통해 양식을 제출해야 합니다. 확인 프로세스에 대한 자세한 내용은 Microsoft 설명서에서 [License Mobility through](#)

[Software Assurance](#)를 참조하세요. 라이선스 이동 확인 양식에는 공인 모빌리티 파트너에 대한 정보를 제공하는 섹션이 있습니다. microsoft@amazon.com을 이메일 주소로, Amazon Web Services를 파트너 이름으로, aws.amazon.com을 파트너 웹 사이트로 사용할 수 있습니다. 자세한 지침은 Microsoft 설명서에서 Microsoft의 [Verification Guide for Customers](#)(다운로드 가능한 PDF)를 참조하세요. 라이선스 이동 확인 양식의 사본을 다운로드하려면 Microsoft 설명서에서 [licensing Resources and Documents](#)를 참조하세요.

참고

AWS가 Microsoft에 의해 정식 제공업체* 클라우드로 선정되었기 때문에, Microsoft에서 제공하는 유연한 가상화 프로그램은 AWS에서 사용할 수 없습니다. Microsoft는 2019년 10월 1일 자 [라이선스 변경 사항](#)의 일환으로 Alibaba, Amazon, Google Cloud를 [정식 제공업체](#)로 지정했습니다. 2019년 10월 1일부터 SA 및 라이선스 이동 옵션 없이 구매한 온프레미스 라이선스는 정식 제공업체가 제공하는 호스팅 클라우드 서비스를 배포할 수 없습니다.

라이선스 이동이 적용되지 않는 Microsoft 제품

Windows Server, Visual Studio, Microsoft Developer Network(MSDN), Windows 데스크톱 운영 체제, Microsoft Office 및 Microsoft 365 앱(이전 Office 365)은 라이선스에 SA가 포함되어 있거나 활성 구독 라이선스인 경우에도 Microsoft 제품 약관에 따라 라이선스 이동이 적용되지 않습니다. 따라서 이러한 제품에 대한 라이선스를 제공하려면 EC2 전용 호스트, EC2 전용 인스턴스, AWS의 VMware Cloud, Outposts의 전용 호스트와 같은 전용 인프라가 필요합니다. 또한 BYOL에서 AWS로 전환할 자격을 갖추려면 다른 특정 요구 사항도 준수해야 합니다. 이러한 요구 사항은 Microsoft가 2019년 10월 1일부터 정식 제공업체 클라우드에 배포할 때 라이선스 이동이 적용되지 않는 제품에 대한 라이선스 조건을 변경한 결과입니다. 자세한 내용은 Microsoft 설명서에서 [Updated Microsoft licensing terms for dedicated hosted cloud services](#)를 참조하세요.

AWS로의 BYOL 자격을 얻으려면 라이선스 이동이 적용되지 않는 제품의 라이선스가 Microsoft의 다음 요구 사항을 충족해야 합니다.

- 영구 사용 권한(구독이 아님)으로 구매한 라이선스여야 합니다.
- 라이선스 구매 날짜가 2019년 10월 1일 이전이어야 하며, 그렇지 않으면 2019년 10월 1일 이전에 시작된 Microsoft 기업계약 기간 내에 라이선스를 구매해야 합니다.
- 배포된 버전은 2019년 10월 1일 이전에 공개적으로 정식 출시된 것이어야 합니다.
- 제품은 전용 인프라에 배포해야 합니다.

라이선스 이동이 적용되지 않는 제품의 구독 라이선스를 2019년 10월 1일 이후에 구매 또는 갱신한 경우 BYOL 자격을 잃게 됩니다.

참고

라이선스 이동이 적용되지 않는 제품은 라이선스가 위의 요구 사항을 충족하는 한 AWS에서의 BYOL을 위한 활성 SA가 필요하지 않습니다.

라이선스는 복잡할 수 있으므로 [Amazon Web Services](#) 및 [Microsoft FAQ 사이트](#)를 참조하여 사용 중인 라이선스가 AWS로의 BYOL 옵션에 적격인지 확인하세요. FAQ에서 필요한 정보를 찾을 수 없거나 Microsoft 워크로드를 AWS로 마이그레이션하는 작업을 어디서부터 시작해야 할지 잘 모르겠으면 Microsoft@Amazon.com으로 문의하세요. AWS의 Microsoft 워크로드 및 라이선스 전문가가 필요한 모든 정보를 갖추도록 도와드립니다.

참고

Windows Server BYOL은 물리적 코어를 기준으로 라이선스를 받아야 하기 때문에 Windows Server BYOL에는 EC2 전용 호스트, Outpost의 전용 호스트 또는 AWS의 VMware Cloud가 필요합니다.

Services Provider License Agreement(SPLA)의 BYOL

Services Provider License Agreement(SPLA) 프로그램은 Microsoft의 2019년 10월 1일자 [라이선스 변경 사항](#)의 영향을 받지 않았습니다. 따라서 자체 SPLA 라이선스를 보유한 고객은 구매 날짜 또는 버전의 제한 없이 SPLA를 통해 신규 Windows Server 라이선스를 가져올 수 있습니다. SPLA를 통해 라이선스를 받은 모든 코어 또는 프로세서 기반 제품에는 사용자 기반 구독자 액세스 라이선스(SAL)를 공유 테넌시(기본) EC2 인스턴스로 가져올 수 있는 EC2 전용 호스트가 필요합니다. SPLA의 사용자 기반 SAL은 [서비스 공급자 사용 권한\(SPUR\)](#)에서 데이터 센터 제공업체(DCP) 자격이 부여되기 때문입니다.

참고

Microsoft는 2025년 9월 30일 이후 더 이상 AWS 또는 다른 정식 제공업체 클라우드에서 SPLA BYOL을 허용하지 않을 것이라고 [발표했습니다](#).

Amazon EC2 전용 호스트

[Amazon EC2 전용 호스트](#)의 주요 기능은 다음과 같습니다.

- 물리적 소켓 및 코어에 대한 가시성을 제공하는 사전 구성된 EC2 Nitro 및 Xen 하이퍼바이저
- 동일한 전용 호스트에서 지원되는 동일 패밀리 내 여러 인스턴스 크기(지원되는 최신 인스턴스 유형 세트는 Windows 인스턴스에 대한 Amazon EC2 사용 설명서의 [전용 호스트](#) 참조)
- 자동 관리, Auto Scaling, 인스턴스 배치 제어
- 여러 AWS 계정에서 호스트를 공유하는 기능
- [AWS License Manager](#)와 통합되어 라이선스 사용량 및 관리 현황 추적
- 호스트에 대한 인스턴스 선호도를 유지하는 기능
- 자동 호스트 복구
- AWS Config를 사용한 지속적 모니터링

Windows Server BYOL에는 전용 인프라와 물리적 코어 수가 필요하므로 EC2 전용 호스트는 다음과 같은 이점을 제공할 수 있는 훌륭한 옵션이 됩니다.

- 상당한 비용 절감 실현
- SA 또는 라이선스 이동에 관계없이 모든 Microsoft 애플리케이션을 AWS로 가져올 수 있음(2019년 10월 1일 구매 조건 및 버전 요구 사항 적용)
- Windows Server Datacenter 및 SQL Server Enterprise 에디션의 물리적 코어 라이선스 혜택 극대화
- EC2 인스턴스가 아니라 호스트별로만 요금 지불(전용 호스트를 사용하면 추가 컴퓨팅 요금 없이 호스트에서 사용 가능한 최대 인스턴스 수를 사용 가능)

BYOL 적격 Windows Server 라이선스를 EC2 전용 호스트에 제공하는 경우 호스트의 모든 물리적 코어(vCPU 제외)에 라이선스를 부여할 수 있습니다. 예를 들어 R5 EC2 전용 호스트에는 48개의 물리적 코어가 있습니다. Windows Server Datacenter 에디션의 48개의 코어를 R5 EC2 전용 호스트에서 사용하면 기술적으로 가능한 최대의 EC2 인스턴스를 호스트에 배포할 수 있습니다. Windows Server Standard 에디션의 48개의 코어를 사용하면 호스트에서 크기와 관계없이 EC2 인스턴스를 최대 2개까지 사용할 수 있습니다.

Windows Server Standard 에디션 라이선스를 중첩 적용하여 동일한 호스트에 EC2 인스턴스를 추가할 수 있습니다. 그러면 두 번째로 라이선스가 부여된 호스트의 모든 물리적 코어에 두 개의 추가 EC2 인스턴스(또는 그 이상)를 추가할 수 있습니다. 물리적 코어별로 SQL Server Enterprise에 라이선스를 부여하려면 호스트의 모든 물리적 코어에 라이선스를 부여해야 합니다. 이렇게 하면 라이선스가 부여

된 물리적 코어 수와 동일한 수의 SQL Server용 EC2 인스턴스를 호스트에 배포할 수 있습니다. 예를 들어 SQL Server Enterprise의 48개 코어 라이선스가 부여된 R5 EC2 전용 호스트를 사용하면 해당 호스트에서 SQL Server를 실행하는 EC2 인스턴스를 최대 48개까지 배포할 수 있습니다.

BYOL 적격 Windows Server Datacenter 및 SQL Server Enterprise 라이선스를 가져와 호스트의 전체 물리적 코어에 라이선스를 부여하면 동일한 수와 크기의 EC2 인스턴스에 포함된 라이선스에 비해 상당한 비용 절감 효과를 얻을 수 있습니다. 이는 워크로드가 대부분 호스트를 가득 채울 수 있고 대부분의 시간 동안 실행되고 있다고 가정했을 때를 기준으로 합니다. 예를 들어 라이선스가 포함된 Windows Server 및 SQL Server Enterprise BYOL이 포함된 공유 테넌시 인스턴스에 R5.2xlarge EC2 인스턴스 12개를 배포할 수 있으며 라이선싱하려면 총 96개의 SQL Server Enterprise 코어가 필요합니다. 하지만 R5 EC2 전용 호스트(동일한 12개의 R5.2xlarge EC2 인스턴스에 맞출 수 있음)를 배포하는 경우 Windows Server Datacenter의 48개 코어와 SQL Server Enterprise BYOL 적격 라이선스의 48개 코어를 사용할 수 있습니다. Windows Server 라이선스 포함 비용을 절약할 수 있을 뿐만 아니라, SQL Server Enterprise 코어 라이선스 수의 절반만 가져오면 됩니다.

BYOL on EC2 전용 호스트는 호스트를 70% 이상 채울 수 있고 워크로드가 대부분의 시간 동안 실행되는, 안정적이고 예측 가능한 워크로드에 가장 적합합니다. AWS 기반 Microsoft 라이선싱에 대해 자세히 알아보려면 [Microsoft 설명서에서 AWS 기반 Microsoft 라이선싱](#) YouTube 및 [Amazon Web Services](#) 및 [Microsoft 자주 묻는 질문](#)을 참조하십시오.

AWS의 VMware Cloud

AWS의 VMware Cloud로 마이그레이션하는 방법에 대해 자세히 알아보려면 AWS 권장 가이드 설명서에서 [AWS의 VMware Cloud 개요 및 운영 모델](#)을 참조하세요.

알림

2024년 4월 30일부터 VMware Cloud AWS on은 더 이상 AWS 또는 해당 채널 파트너에 의해 재판매되지 않습니다. 이 서비스는 Broadcom을 통해 계속 제공될 예정입니다. 자세한 내용은 AWS 담당자에게 문의하시기 바랍니다.

동원

AWS License Manager

Microsoft 라이선스 고려 사항을 위한 동원 단계의 일환으로 AWS의 워크로드에 할당하려는 라이선스를 [AWS License Manager](#)에 입력하는 것이 좋습니다. License Manager는 AWS뿐만 아니라 온프레미

스 또는 다른 클라우드의 워크로드에 걸쳐 Microsoft, Oracle, IBM, SAP 등 여러 공급업체의 소프트웨어 라이선스를 보다 손쉽게 관리할 수 있는 무료 도구입니다.

AWS로 가져오려는 Microsoft 라이선스를 License Manager에 입력하면 다음에 도움이 됩니다.

- 소프트웨어 라이선스가 사용되는 방식에 대한 가시성과 통제력을 높이고 오용을 미연에 방지합니다.
- 라이선스 추적 및 관리 방법을 비롯하여, 라이선스를 최대한 활용하여 비용을 절감합니다.
- 라이선스 사용 제한을 적용하고, 새로운 시작을 차단하고, 기타 통제를 통해 규정 미준수 위험을 줄입니다.
- 호스트 리소스 그룹을 사용하여 호스트의 배치, 해제 및 복구를 자동화함으로써 생산성을 높입니다.

License Manager에 대한 자세한 내용은 AWS License Manager 사용 설명서에서 [Working with AWS License Manager](#) 섹션을 참조하세요.

라이선싱 고려 사항

마이그레이션 전에 현재 워크로드에 할당된 라이선스를 중심으로 마이그레이션을 계획해 보세요. 예를 들어 여러 온프레미스 호스트를 AWS로 가져오는 경우 여러 호스트에 있는 워크로드를 그룹화하는 대신 호스트별로 마이그레이션하는 것을 고려해 보세요. 온프레미스 호스트를 사용 중지하면 해당 호스트와 관련된 라이선스를 AWS에서 사용할 수 있게 되기 때문입니다. 또는 마이그레이션 중에 Windows Server 또는 SQL Server용 라이선스가 포함된 인스턴스를 사용하고, 마이그레이션이 완료된 후 BYOL 옵션으로 전환할 수 있습니다. 하지만 이 옵션을 사용하려면 처음부터 자체 미디어와 AMI를 사용해야 합니다(라이선스 포함 옵션인 경우에도 동일). AWS License Manager에서 제공되는 [라이선스 변환 기능](#)을 사용하면 EC2 인스턴스가 원래 자체 미디어 및 AMI에서 생성된 경우에만 라이선스 포함 옵션에서 BYOL 옵션으로 전환할 수 있습니다.

마이그레이션

AWS에 Microsoft 워크로드를 배포한 후 10일 이내에 AWS로 가져오려는 라이선스 이동이 적용되는 라이선스에 대한 [라이선스 이동 확인 양식](#)을 Microsoft에 제출해야 합니다. 마이그레이션의 여러 단계에 따라 이 양식을 여러 번 제출할 수 있습니다. 이 양식에는 다음 정보가 필요합니다.

- 이름 및 연락처 정보
- Microsoft 계약 번호
- 이용 중인 클라우드 파트너
- 라이선스 이동을 통해 사용할 제품

- 사용할 라이선스 수

확인 프로세스에 대한 자세한 내용은 Microsoft 설명서에서 [License Mobility through Software Assurance](#)를 참조하세요. 자세한 지침은 Microsoft 설명서에서 Microsoft의 [Verification Guide for Customers](#)(다운로드 가능한 PDF)를 참조하세요. 라이선스 이동 확인 양식의 사본을 다운로드하려면 Microsoft 설명서에서 [icensing Resources and Documents](#)를 참조하세요.

AWS 파트너

AWS 컴피턴시 파트너 참여의 이점

Microsoft 워크로드를 클라우드로 효율적으로 마이그레이션하려면 신중한 계획과 간소화된 구현이 필요합니다. 주요 단계에는 범위 지정, 클라우드 마이그레이션 비즈니스 사례 수립, 경영진 조정, 클라우드 재무 관리 KPI 설정, 클라우드 우수 센터 구축, 마이그레이션 서비스 검증, 대규모 마이그레이션을 위한 자동화 도구 배포, 클라우드로의 보안 전략 확대 등이 포함됩니다.

검증된 [AWS 컴피턴시 파트너](#)를 고용하여 조직의 마이그레이션 과정을 이끄는 것이 좋습니다. AWS 파트너는 전략 전문가이자 경험이 풍부한 구축자로, 마이그레이션 여정의 모든 단계를 안내하여 앞서 언급한 주요 단계와 비즈니스 목표를 달성하는 데 도움을 줍니다. AWS 파트너 커뮤니티에는 클라우드 여정을 지원하고 혁신, 민첩성 향상, 비용 절감에 집중할 수 있도록 도와주는 150여 개국의 100,000개 이상의 파트너가 있습니다.

계획 수립

AWS 파트너는 준비 상태 평가를 수행하고, 마이그레이션 계획을 수립하고, 마이그레이션 도구를 제공하여 클라우드로의 여정을 가속화할 수 있습니다. 또한 기술 격차를 해소하고, 비용 최적화 전략을 추천하고, AWS로 마이그레이션하는 데 드는 비용을 보조하는 독점 마이그레이션 인센티브를 받을 자격을 갖추도록 도와줍니다.

비용 최적화

오늘날의 빠르게 진화하는 기술 환경에서 많은 조직은 디지털 혁신 여정과 관련하여 상당한 비용 문제에 직면해 있습니다. 한 가지 공통적인 우려는 클라우드가 너무 비싸서 클라우드가 제공하는 중요한 비즈니스 이점을 파악하기 어렵다는 인식입니다. 또한 기술 스택을 현대화하는 데 드는 비용으로 인해 재정적 문제가 발생할 수 있습니다.

[AWS Microsoft 워크로드 컴피턴시 파트너와 협력하면 가장 자격을 갖춘 AWS 파트너가 AWS에 Microsoft 워크로드를 배포할 수 있습니다.](#) 이러한 파트너는 기술 역량을 검증하고 고객이 Microsoft 워크로드를 AWS로 마이그레이션, 관리 또는 배포하도록 지원하는 데 있어 성과를 입증했습니다. 이러한 파트너가 지원하는 워크로드에는 Windows Server, Microsoft SQL Server, Windows 파일 서버 SharePoint, .NET 애플리케이션이 포함됩니다.

AWS 파트너는 AWS 모범 사례를 사용하여 안전하고, 가용성이 높고, 안정적이며, 성능이 뛰어나고, 비용이 최적화된 아키텍처를 구축합니다. 또한 파트너는 AWS에서 제공하는 자금을 최대한 활용하여

비용을 최적화하고 전문 지식으로 가치를 창출하는 시간을 단축할 수 있도록 지원합니다. 마지막으로, AWS 파트너는 [Windows용 AWS Migration Acceleration Program](#)을 활용하여 AWS로의 마이그레이션 비용을 상쇄할 수 있습니다.

시간을 절약하세요.

알림

2024년 4월 30일부터 VMware Cloud AWS on은 더 이상 채널 파트너에 의해 AWS 재판매되지 않습니다. 이 서비스는 Broadcom을 통해 계속 제공될 예정입니다. 자세한 내용은 AWS 담당자에게 문의하시기 바랍니다.

많은 기업이 온프레미스 인프라에 막대한 투자를 하고 있습니다. 조직에서 온프레미스 인프라를 관리하기 위해 VMware 소프트웨어에 대규모 투자를 했고 동일한 온프레미스 도구를 사용하여 AWS상의 인프라를 관리하고자 할 수 있습니다. 클라우드로 마이그레이션하기는 어렵지만 마이그레이션된 워크로드에 의존하는 특수 워크로드와 인프라가 있을 수도 있습니다. 또한 일부 인프라는 기존 온프레미스 데이터 센터에 있고 다른 일부는 클라우드로 배포되는 하이브리드 인프라 패턴을 사용할 수도 있습니다.

시간이 촉박할 때는 숙련된 인재, 정교한 프로세스 및 기술 [역량을 바탕으로 광범위한 대규모 마이그레이션을 수행한 검증된 실적을 보유한 AWS 마이그레이션 컴피턴시 파트너와](#) 상담하는 것이 좋습니다. 지원되는 워크로드 카테고리에는 Windows, SAP, 오라클, VMware on AWS, 데이터베이스, 분석, 스토리지, IoT, 기계 학습, 서비스형 소프트웨어 등이 포함됩니다.

AWS 파트너는 AWS로 이전한다고 해서 기존 all-or-nothing 투자를 없애는 것이 아니라는 점을 이해하고 있습니다. 이들은 인프라를 최적화하고 간소화하는 데 능숙하며, 온프레미스에 보관하는 것이 가장 좋은 부분과 클라우드로 가장 적합한 부분을 최적화합니다. AWS는 Amazon VPC, Direct Connect 및 Storage Gateway를 비롯한 다양한 하이브리드 클라우드 솔루션을 제공합니다.

AWS 파트너는 수천 명의 기업 고객을 클라우드로 마이그레이션한 [AWS의 경험을 바탕으로 하는 포괄적이고 검증된 클라우드 마이그레이션 프로그램인 AWS Migration Acceleration Program \(MAP\)](#)을 적격 고객에게 제공할 수 있습니다. MAP는 포괄적인 도구, 서비스, 지침, 교육 및 추가 인센티브를 통해 특화된 워크로드를 지원합니다. 메인프레임, 윈도우, 스토리지, VMware Cloud on AWS, SAP, 데이터베이스 및 Amazon Connect에 대한 전문 워크로드 지원이 제공됩니다.

보안 강화

데이터의 개인 정보 보호 및 보안이 걱정될 수 있습니다. 또한 데이터 처리 관행이 합법적인 해외 데이터 사용 명확화 (CLOUD) 법 및 일반 데이터 보호 규정 (GDPR) 을 준수하는지 확인해야 할 수도 있습니다. 특정 워크로드 및 사용 사례에 대한 [보안 중심 솔루션을 제공하기 위한 보안 전문가 팀을 제공할 수 있는 AWS 보안 컴피턴시 파트너와](#) 계약하는 것이 좋습니다. AWS 파트너 솔루션을 사용하면 워크로드에 따라 자동화와 민첩성 및 확장이 가능합니다.

게시 당시 AWS는 PCI-DSS, HIPAA/HITECH, FedRAMP, GDPR, FIPS 140-2 및 NIST 800-171과 같은 광범위한 보안 표준 및 규정 준수 인증을 지원합니다. 전 세계 대부분의 규제 기관의 규정 준수 요구 사항을 충족할 수 있도록 지원합니다.

의료, 은행, 법률, 제약 등 보안에 가장 민감한 업종의 민간 및 공공 부문 조직은 AWS를 신뢰하여 보안 태세를 개선했습니다. 중소기업이든 대기업이든 공공 부문 조직이든 관계없이 비즈니스를 발전시키는 데 도움이 되는 적절한 기술과 경험을 갖춘 AWS 파트너가 있습니다. AWS 파트너 전문가가 비즈니스 요구 사항에 맞는 적절한 클라우드 파트너를 찾고 연결하도록 도와드릴 수 있습니다. 자세한 내용은 [AWS 파트너 전문가에게](#) 문의하십시오. 전 세계 고객이 AWS 파트너 네트워크 (APN) 를 통해 클라우드 채택을 가속화하고 혁신을 촉진하는 방법을 알아보려면 [AWS 파트너와 함께하는 고객 성공을](#) 참조하십시오.

다음 단계

다음 단계를 수행하는 것이 좋습니다.

1. 특정 마이그레이션 및 현대화 시나리오에 대해 자세히 알아보십시오. 자세한 내용은 [을 참조하십시오. Microsoft SQL Server 데이터베이스를 AWS 클라우드로 마이그레이션, RDBMS에서 Amazon DynamoDB로 마이그레이션하여 애플리케이션을 현대화합니다., 및 NET 애플리케이션 현대화를 위한 접근 방식 선택.](#)
2. 대규모 마이그레이션이 조직에 미치는 영향에 대해 자세히 알아보십시오. 대규모 마이그레이션은 기술 혁신일 뿐만 아니라 조직의 역할, 프로세스 및 우선 순위에 대한 변화도 수반합니다. 자세한 내용은 [을 참조하십시오. AWS 대규모 마이그레이션을 위한 전략 및 모범 사례.](#)
3. 검토하기 [마이크로소프트 워크로드용 AWS 자율 학습 가이드.](#)
4. 완료하기 [Microsoft 워크로드를 AWS 실습 워크숍으로 마이그레이션.](#)

리소스

마이크로소프트에서 AWS로의 마이그레이션 가이드라인

- [Microsoft 워크로드를 AWS로 마이그레이션: 자율 학습 가이드](#)
- [Microsoft 워크로드를 AWS로 마이그레이션하기: 실습](#)
- [Microsoft SQL Server 데이터베이스를 AWS 클라우드로 마이그레이션](#)
- [RDBMS에서 Amazon DynamoDB로 마이그레이션하여 애플리케이션을 현대화합니다.](#)
- [.NET 애플리케이션 현대화를 위한 접근 방식 선택](#)
- [AWS 대규모 마이그레이션을 위한 전략 및 모범 사례.](#)

일반 가이드라인

- [AWS 기반 윈도우](#)
- [AWS 대규모 마이그레이션을 위한 전략 및 모범 사례](#)
- [AWS 설명서에 오신 것을 환영합니다](#)

동영상

- [AWS re:Invent 2020: 마이크로소프트 워크로드를 AWS로 마이그레이션](#)
- [AWS 애플리케이션 마이그레이션 서비스를 통한 Windows 워크로드 재호스팅 — AWS 가상 워크샵](#)

AWS 블로그 게시물

- [AWS 애플리케이션 마이그레이션 서비스로 온프레미스 워크로드를 마이그레이션하는 방법](#)
- [AWS로 Windows 워크로드를 마이그레이션해야 하는 이유 \(및 AWS가 도와드릴 수 있는 방법\)](#)

문서 기록

아래 표에 이 가이드의 주요 변경 사항이 설명되어 있습니다. 향후 업데이트에 대한 알림을 받으려면 [RSS 피드](#)를 구독하십시오.

변경 사항	설명	날짜
업데이트	Windows 장애 조치 클러스터 마이그레이션 섹션에 Amazon EBS 다중 연결에 대한 정보가 추가되었습니다.	2024년 4월 1일
업데이트	마이그레이션 검증기 툴킷 모듈에 대한 링크가 추가되었습니다. PowerShell Windows 장애 조치 클러스터 마이그레이션 페이지의 자습서: Amazon EC2에 Windows HPC 클러스터 설치 사용에 대한 지침이 명확해졌습니다.	2023년 12월 14일
업데이트	Windows 장애 조치 클러스터 마이그레이션 페이지가 업데이트되었습니다.	2023년 12월 8일
업데이트	AWS에서의 Microsoft 라이선스 페이지의 Amazon EC2 전용 호스트 섹션에서 전용 호스트에 대해 지원되는 인스턴스 유형 목록이 업데이트되었습니다.	2023년 11월 16일
업데이트	AWS에서의 Microsoft 라이선스 페이지의 Amazon EC2 전용 호스트 섹션에 지원되는 인스턴스 패밀리의 전체 목록을 추가했습니다.	2023년 7월 31일

[업데이트](#)

Migrating SQL Server 페이지
의 리플랫폼 섹션에 BYOM
지침을 추가했습니다.

2023년 6월 23일

[최초 게시](#)

—

2023년 6월 9일

AWS 규범적 지침 용어집

다음은 규범적 지침에서 제공하는 AWS 전략, 가이드 및 패턴에서 일반적으로 사용되는 용어입니다. 용어집 항목을 제안하려면 용어집 끝에 있는 피드백 제공 링크를 사용하십시오.

숫자

7가지 전략

애플리케이션을 클라우드로 이전하기 위한 7가지 일반적인 마이그레이션 전략 이러한 전략은 Gartner가 2011년에 파악한 5가지 전략을 기반으로 하며 다음으로 구성됩니다.

- 리팩터링/리아키텍트 - 클라우드 네이티브 기능을 최대한 활용하여 애플리케이션을 이동하고 해당 아키텍처를 수정함으로써 민첩성, 성능 및 확장성을 개선합니다. 여기에는 일반적으로 운영 체제와 데이터베이스 이식이 포함됩니다. 예: 온프레미스 Oracle 데이터베이스를 Amazon Aurora PostgreSQL 호환 에디션으로 마이그레이션하십시오.
- 리플랫폼(리프트 앤드 리세이프) - 애플리케이션을 클라우드로 이동하고 일정 수준의 최적화를 도입하여 클라우드 기능을 활용합니다. 예: 온프레미스 Oracle 데이터베이스를 오라클용 Amazon RDS (Amazon RDS) 로 마이그레이션합니다. AWS 클라우드
- 재구매(드롭 앤드 쇼프) - 일반적으로 기존 라이선스에서 SaaS 모델로 전환하여 다른 제품으로 전환합니다. 예: 고객 관계 관리 (CRM) 시스템을 Salesforce.com으로 마이그레이션하십시오.
- 리호스팅(리프트 앤드 시프트) - 애플리케이션을 변경하지 않고 클라우드로 이동하여 클라우드 기능을 활용합니다. 예: 온프레미스 Oracle 데이터베이스를 EC2 인스턴스에서 Oracle로 마이그레이션합니다. AWS 클라우드
- 재배포(하이퍼바이저 수준의 리프트 앤 시프트) - 새 하드웨어를 구매하거나, 애플리케이션을 다시 작성하거나, 기존 운영을 수정하지 않고도 인프라를 클라우드로 이동합니다. 온프레미스 플랫폼에서 동일한 플랫폼의 클라우드 서비스로 서버를 마이그레이션합니다. 예: Microsoft Hyper-V 애플리케이션을 다음으로 마이그레이션하십시오. AWS
- 유지(보관) - 소스 환경에 애플리케이션을 유지합니다. 대규모 리팩터링이 필요하고 해당 작업을 나중에 연기하려는 애플리케이션과 비즈니스 차원에서 마이그레이션할 이유가 없어 유지하려는 레거시 애플리케이션이 여기에 포함될 수 있습니다.
- 사용 중지 - 소스 환경에서 더 이상 필요하지 않은 애플리케이션을 폐기하거나 제거합니다.

A

ABAC

[속성 기반 액세스](#) 제어를 참조하십시오.

추상화된 서비스

[관리형 서비스를](#) 참조하십시오.

산

[원자성, 일관성, 격리성, 내구성을](#) 참조하십시오.

능동-능동 마이그레이션

양방향 복제 도구 또는 이중 쓰기 작업을 사용하여 소스 데이터베이스와 대상 데이터베이스가 동기화된 상태로 유지되고, 두 데이터베이스 모두 마이그레이션 중 연결 애플리케이션의 트랜잭션을 처리하는 데이터베이스 마이그레이션 방법입니다. 이 방법은 일회성 전환이 필요한 대신 소규모의 제어된 배치로 마이그레이션을 지원합니다. [더 유연하지만 액티브-패시브 마이그레이션보다 더 많은 작업이 필요합니다.](#)

능동-수동 마이그레이션

소스 데이터베이스와 대상 데이터베이스가 동기화된 상태로 유지되지만 소스 데이터베이스만 연결 애플리케이션의 트랜잭션을 처리하고 데이터는 대상 데이터베이스로 복제되는 데이터베이스 마이그레이션 방법입니다. 대상 데이터베이스는 마이그레이션 중 어떤 트랜잭션도 허용하지 않습니다.

집계 함수

행 그룹에서 연산을 수행하고 그룹에 대한 단일 반환값을 계산하는 SQL 함수입니다. 집계 함수의 예로는 `MAX` 및 `SUM` 등이 있습니다.

AI

[인공 지능을](#) 참조하십시오.

AIOps

[인공 지능 운영을](#) 참조하십시오.

익명화

데이터세트에서 개인 정보를 영구적으로 삭제하는 프로세스입니다. 익명화는 개인 정보 보호에 도움이 될 수 있습니다. 익명화된 데이터는 더 이상 개인 데이터로 간주되지 않습니다.

안티 패턴

솔루션이 다른 솔루션보다 비생산적이거나 비효율적이거나 덜 효과적이어서 반복되는 문제에 자주 사용되는 솔루션입니다.

애플리케이션 제어

시스템을 멀웨어로부터 보호하기 위해 승인된 애플리케이션만 사용할 수 있는 보안 접근 방식입니다.

애플리케이션 포트폴리오

애플리케이션 구축 및 유지 관리 비용과 애플리케이션의 비즈니스 가치를 비롯하여 조직에서 사용하는 각 애플리케이션에 대한 세부 정보 모음입니다. 이 정보는 [포트폴리오 검색 및 분석 프로세스](#)의 핵심이며 마이그레이션, 현대화 및 최적화할 애플리케이션을 식별하고 우선순위를 정하는 데 도움이 됩니다.

인공 지능

컴퓨터 기술을 사용하여 학습, 문제 해결, 패턴 인식 등 일반적으로 인간과 관련된 인지 기능을 수행하는 것을 전문으로 하는 컴퓨터 과학 분야입니다. 자세한 내용은 [What is Artificial Intelligence?](#)를 참조하십시오.

인공 지능 운영(AIOps)

기계 학습 기법을 사용하여 운영 문제를 해결하고, 운영 인시던트 및 사용자 개입을 줄이고, 서비스 품질을 높이는 프로세스입니다. AWS 마이그레이션 전략에서 AIOps가 사용되는 방법에 대한 자세한 내용은 [운영 통합 가이드](#)를 참조하십시오.

비대칭 암호화

한 쌍의 키, 즉 암호화를 위한 퍼블릭 키와 복호화를 위한 프라이빗 키를 사용하는 암호화 알고리즘입니다. 퍼블릭 키는 복호화에 사용되지 않으므로 공유할 수 있지만 프라이빗 키에 대한 액세스는 엄격히 제한되어야 합니다.

원자성, 일관성, 격리성, 내구성(ACID)

오류, 정전 또는 기타 문제가 발생한 경우에도 데이터베이스의 데이터 유효성과 운영 신뢰성을 보장하는 소프트웨어 속성 세트입니다.

ABAC(속성 기반 액세스 제어)

부서, 직무, 팀 이름 등의 사용자 속성을 기반으로 세분화된 권한을 생성하는 방식입니다. 자세한 내용은 AWS Identity and Access Management (IAM) [설명서의 AWS ABAC](#) for를 참조하십시오.

신뢰할 수 있는 데이터 소스

가장 신뢰할 수 있는 정보 소스로 간주되는 기본 버전의 데이터를 저장하는 위치입니다. 익명화, 편집 또는 가명화와 같은 데이터 처리 또는 수정의 목적으로 신뢰할 수 있는 데이터 소스의 데이터를 다른 위치로 복사할 수 있습니다.

가용 영역

다른 가용 영역의 장애로부터 격리되고 동일한 지역 내 다른 가용 영역에 저렴하고 지연 시간이 짧은 네트워크 연결을 제공하는 별도의 위치. AWS 리전

AWS 클라우드 채택 프레임워크 (AWS CAF)

조직이 클라우드로 성공적으로 AWS 전환하기 위한 효율적이고 효과적인 계획을 개발하는 데 도움이 되는 지침 및 모범 사례 프레임워크입니다. AWS CAF는 지침을 관점이라고 하는 6가지 중점 영역, 즉 비즈니스, 사람, 거버넌스, 플랫폼, 보안, 운영으로 분류합니다. 비즈니스, 사람 및 거버넌스 관점은 비즈니스 기술과 프로세스에 초점을 맞추고, 플랫폼, 보안 및 운영 관점은 전문 기술과 프로세스에 중점을 둡니다. 예를 들어, 사람 관점은 인사(HR), 직원 배치 기능 및 인력 관리를 담당하는 이해관계자를 대상으로 합니다. 이러한 관점에서 AWS CAF는 조직이 성공적인 클라우드 채택을 준비할 수 있도록 인력 개발, 교육 및 커뮤니케이션에 대한 지침을 제공합니다. 자세한 내용은 [AWS CAF 웹 사이트](#)와 [AWS CAF 백서](#)를 참조하십시오.

AWS 워크로드 검증 프레임워크 (AWS WQF)

데이터베이스 마이그레이션 워크로드를 평가하고 마이그레이션 전략을 권장하며 작업 예상치를 제공하는 도구입니다. AWS WQF는 () 에 포함됩니다. AWS Schema Conversion Tool AWS SCT데이터베이스 스키마 및 코드 객체, 애플리케이션 코드, 종속성 및 성능 특성을 분석하고 평가 보고서를 제공합니다.

B

배드 봇

개인이나 조직을 방해하거나 피해를 입히려는 의도를 가진 [봇입니다](#).

BCP

[비즈니스 연속성 계획을](#) 참조하십시오.

동작 그래프

리소스 동작과 시간 경과에 따른 상호 작용에 대한 통합된 대화형 뷰입니다. Amazon Detective에서 동작 그래프를 사용하여 실패한 로그인 시도, 의심스러운 API 호출 및 유사한 작업을 검사할 수 있습니다. 자세한 내용은 Detective 설명서의 [Data in a behavior graph](#)를 참조하십시오.

빅 엔디안 시스템

가장 중요한 바이트를 먼저 저장하는 시스템입니다. [엔디안도](#) 참조하십시오.

바이너리 분류

바이너리 결과(가능한 두 클래스 중 하나)를 예측하는 프로세스입니다. 예를 들어, ML 모델이 “이 이메일이 스팸인가요, 스팸이 아닌가요?”, ‘이 제품은 책임가요, 자동차인가요?’ 등의 문제를 예측해야 할 수 있습니다.

블룸 필터

요소가 세트의 멤버인지 여부를 테스트하는 데 사용되는 메모리 효율성이 높은 확률론적 데이터 구조입니다.

블루/그린(Blue/Green) 배포

서로 다르지만 동일한 환경을 두 개 만드는 배포 전략입니다. 현재 애플리케이션 버전을 한 환경 (파란색) 에서 실행하고 다른 환경 (녹색) 에서 새 애플리케이션 버전을 실행합니다. 이 전략을 사용하면 영향을 최소화하면서 신속하게 롤백할 수 있습니다.

bot

인터넷을 통해 자동화된 작업을 실행하고 사람의 활동이나 상호 작용을 시뮬레이션하는 소프트웨어 애플리케이션입니다. 인터넷에서 정보를 인덱싱하는 웹 크롤러와 같은 일부 봇은 유용하거나 유용합니다. 배드 봇으로 알려진 일부 다른 봇은 개인이나 조직을 방해하거나 피해를 입히기 위한 것입니다.

봇넷

[멀웨어에 감염되어 봇 허더 또는 봇 운영자로 알려진 단일 당사자의 통제 하에 있는 봇 네트워크](#). 봇넷은 봇과 그 영향을 확장하는 가장 잘 알려진 메커니즘입니다.

브랜치

코드 리포지토리의 포함된 영역입니다. 리포지토리에 생성되는 첫 번째 브랜치가 기본 브랜치입니다. 기존 브랜치에서 새 브랜치를 생성한 다음 새 브랜치에서 기능을 개발하거나 버그를 수정할 수 있습니다. 기능을 구축하기 위해 생성하는 브랜치를 일반적으로 기능 브랜치라고 합니다. 기능을 출시할 준비가 되면 기능 브랜치를 기본 브랜치에 다시 병합합니다. 자세한 내용은 [브랜치 정보](#) (문서) 를 참조하십시오. GitHub

브레이크 글래스 액세스

예외적인 상황에서 승인된 프로세스를 통해 사용자가 일반적으로 액세스 권한이 없는 데이터에 빠르게 액세스할 수 있는 AWS 계정 있는 수단입니다. 자세한 내용은 Well-Architected AWS 지침의 [브레이크 글래스 절차 구현](#) 표시기를 참조하십시오.

브라운필드 전략

사용자 환경의 기존 인프라 시스템 아키텍처에 브라운필드 전략을 채택할 때는 현재 시스템 및 인프라의 제약 조건을 중심으로 아키텍처를 설계합니다. 기존 인프라를 확장하는 경우 브라운필드 전략과 [그린필드](#) 전략을 혼합할 수 있습니다.

버퍼 캐시

가장 자주 액세스하는 데이터가 저장되는 메모리 영역입니다.

사업 역량

기업이 가치를 창출하기 위해 하는 일(예: 영업, 고객 서비스 또는 마케팅)입니다. 마이크로서비스 아키텍처 및 개발 결정은 비즈니스 역량에 따라 이루어질 수 있습니다. 자세한 내용은 백서의 [AWS에서 컨테이너화된 마이크로서비스 실행의 비즈니스 역량 중심의 구성화](#) 섹션을 참조하십시오.

비즈니스 연속성 계획(BCP)

대규모 마이그레이션과 같은 중단 이벤트가 운영에 미치는 잠재적 영향을 해결하고 비즈니스가 신속하게 운영을 재개할 수 있도록 지원하는 계획입니다.

C

CAF

[클라우드 채택 프레임워크를 참조하십시오AWS](#).

카나리아 배포

최종 사용자에게 버전을 느리고 점진적으로 릴리스하는 것입니다. 확신이 들면 새 버전을 배포하고 현재 버전을 완전히 교체합니다.

CCoE

[클라우드 센터 오브 엑셀런스를 참조하십시오](#).

CDC

[변경 데이터 캡처를 참조하십시오](#).

변경 데이터 캡처(CDC)

데이터베이스 테이블과 같은 데이터 소스의 변경 내용을 추적하고 변경 사항에 대한 메타데이터를 기록하는 프로세스입니다. 대상 시스템의 변경 내용을 감사하거나 복제하여 동기화를 유지하는 등의 다양한 용도로 CDC를 사용할 수 있습니다.

카오스 엔지니어링

시스템의 복원력을 테스트하기 위해 의도적으로 장애나 장애를 일으키는 이벤트를 발생시키는 행위 [AWS Fault Injection Service \(AWS FIS\)](#) 를 사용하여 AWS 워크로드에 스트레스를 주는 실험을 수행하고 응답을 평가할 수 있습니다.

CI/CD

[지속적 통합 및 지속적 전달](#)을 참조하십시오.

분류

예측을 생성하는 데 도움이 되는 분류 프로세스입니다. 분류 문제에 대한 ML 모델은 이산 값을 예측합니다. 이산 값은 항상 서로 다릅니다. 예를 들어, 모델이 이미지에 자동차가 있는지 여부를 평가해야 할 수 있습니다.

클라이언트측 암호화

대상이 데이터를 AWS 서비스 수신하기 전에 데이터를 로컬로 암호화합니다.

클라우드 혁신 센터(CCoE)

클라우드 모범 사례 개발, 리소스 동원, 마이그레이션 타임라인 설정, 대규모 혁신을 통한 조직 선도 등 조직 전체에서 클라우드 채택 노력을 추진하는 다분야 팀입니다. 자세한 내용은 AWS 클라우드 기업 전략 [블로그의 CCoE 게시물을](#) 참조하십시오.

클라우드 컴퓨팅

원격 데이터 스토리지와 IoT 디바이스 관리에 일반적으로 사용되는 클라우드 기술 클라우드 컴퓨팅은 일반적으로 [엣지 컴퓨팅 기술과](#) 연결됩니다.

클라우드 운영 모델

IT 조직에서 하나 이상의 클라우드 환경을 구축, 성숙화 및 최적화하는 데 사용되는 운영 모델입니다. 자세한 내용은 [클라우드 운영 모델 구축](#)을 참조하십시오.

클라우드 채택 단계

조직이 마이그레이션할 때 일반적으로 거치는 4단계는 다음과 같습니다. AWS 클라우드

- 프로젝트 - 개념 증명 및 학습 목적으로 몇 가지 클라우드 관련 프로젝트 실행
- 기반 - 클라우드 채택 확장을 위한 기초 투자(예: 랜딩 존 생성, CCoE 정의, 운영 모델 구축)
- 마이그레이션 - 개별 애플리케이션 마이그레이션
- Re-invention - 제품 및 서비스 최적화와 클라우드 혁신

Stephen Orban은 기업 전략 블로그의 [클라우드 우선주의를 향한 여정 및 채택 단계에](#) 대한 블로그 게시물에서 이러한 단계를 정의했습니다. AWS 클라우드 [이들이 AWS 마이그레이션 전략과 어떤 관련이 있는지에 대한 자세한 내용은 마이그레이션 준비 가이드를 참조하십시오.](#)

CMDB

[구성 관리 데이터베이스](#)를 참조하십시오.

코드 리포지토리

소스 코드와 설명서, 샘플, 스크립트 등의 기타 자산이 버전 관리 프로세스를 통해 저장되고 업데이트되는 위치입니다. 일반 클라우드 리포지토리에는 또는 이 포함됩니다 GitHub . AWS CodeCommit코드의 각 버전을 브랜치라고 합니다. 마이크로서비스 구조에서 각 리포지토리는 단일 기능 전용입니다. 단일 CI/CD 파이프라인은 여러 리포지토리를 사용할 수 있습니다.

콜드 캐시

비어 있거나, 제대로 채워지지 않았거나, 오래되었거나 관련 없는 데이터를 포함하는 버퍼 캐시입니다. 주 메모리나 디스크에서 데이터베이스 인스턴스를 읽어야 하기 때문에 성능에 영향을 미치며, 이는 버퍼 캐시에서 읽는 것보다 느립니다.

콜드 데이터

거의 액세스되지 않고 일반적으로 과거 데이터인 데이터. 이런 종류의 데이터를 쿼리할 때는 일반적으로 느린 쿼리가 허용됩니다. 이 데이터를 성능이 낮고 비용이 저렴한 스토리지 계층 또는 클래스로 옮기면 비용을 절감할 수 있습니다.

컴퓨터 비전 (CV)

기계 학습을 사용하여 디지털 이미지 및 비디오와 같은 시각적 형식에서 정보를 분석하고 추출하는 [AI](#) 분야. 예를 들어 AWS Panorama 는 온프레미스 카메라 네트워크에 CV를 추가하는 디바이스를 제공하고, SageMaker Amazon은 CV용 이미지 처리 알고리즘을 제공합니다.

구성 드리프트

워크로드의 경우 구성이 예상 상태에서 변경됩니다. 이로 인해 워크로드가 규정을 준수하지 않게 될 수 있으며, 일반적으로 점진적이고 의도하지 않은 방식으로 진행됩니다.

구성 관리 데이터베이스(CMDB)

하드웨어 및 소프트웨어 구성 요소와 해당 구성을 포함하여 데이터베이스와 해당 IT 환경에 대한 정보를 저장하고 관리하는 리포지토리입니다. 일반적으로 마이그레이션의 포트폴리오 검색 및 분석 단계에서 CMDB의 데이터를 사용합니다.

규정 준수 팩

AWS Config 규정 준수 및 보안 검사를 사용자 지정하기 위해 조합할 수 있는 규칙 및 수정 조치 모음입니다. YAML 템플릿을 사용하여 한 AWS 계정 및 지역 또는 조직 전체에 단일 엔티티로 적합성 팩을 배포할 수 있습니다. 자세한 내용은 설명서의 [적합성 팩](#)을 참조하십시오. AWS Config

지속적 통합 및 지속적 전달(CI/CD)

소프트웨어 릴리스 프로세스의 소스, 빌드, 테스트, 스테이징 및 프로덕션 단계를 자동화하는 프로세스입니다. CI/CD는 일반적으로 파이프라인으로 설명됩니다. CI/CD를 통해 프로세스를 자동화하고, 생산성을 높이고, 코드 품질을 개선하고, 더 빠르게 제공할 수 있습니다. 자세한 내용은 [지속적 전달의 이점](#)을 참조하십시오. CD는 지속적 배포를 의미하기도 합니다. 자세한 내용은 [지속적 전달\(Continuous Delivery\)](#)과 [지속적인 개발](#)을 참조하십시오.

CV

[컴퓨터 비전을 참조하십시오.](#)

D

저장 데이터

스토리지에 있는 데이터와 같이 네트워크에 고정되어 있는 데이터입니다.

데이터 분류

중요도와 민감도를 기준으로 네트워크의 데이터를 식별하고 분류하는 프로세스입니다. 이 프로세스는 데이터에 대한 적절한 보호 및 보존 제어를 결정하는 데 도움이 되므로 사이버 보안 위험 관리 전략의 중요한 구성 요소입니다. 데이터 분류는 AWS Well-Architected 프레임워크의 보안 핵심 요소입니다. 자세한 내용은 [데이터 분류](#)를 참조하십시오.

데이터 드리프트

프로덕션 데이터와 ML 모델 학습에 사용된 데이터 간의 상당한 차이 또는 시간 경과에 따른 입력 데이터의 의미 있는 변화. 데이터 드리프트는 ML 모델 예측의 전반적인 품질, 정확성 및 공정성을 저하시킬 수 있습니다.

전송 중 데이터

네트워크를 통과하고 있는 데이터입니다. 네트워크 리소스 사이를 이동 중인 데이터를 예로 들 수 있습니다.

데이터 메시

중앙 집중식 관리 및 거버넌스와 함께 분산되고 분산된 데이터 소유권을 제공하는 아키텍처 프레임워크입니다.

데이터 최소화

꼭 필요한 데이터만 수집하고 처리하는 원칙입니다. 에서 데이터 최소화를 실천하면 개인 정보 보호 위험, 비용 및 분석에 따른 탄소 발자국을 줄일 AWS 클라우드 수 있습니다.

데이터 경계

신뢰할 수 있는 ID만 예상 네트워크에서 신뢰할 수 있는 리소스에 액세스하도록 하는 데 도움이 되는 AWS 환경 내 일련의 예방 가드레일입니다. 자세한 내용은 [데이터 경계 구축을 참조하십시오](#).

AWS

데이터 사전 처리

원시 데이터를 ML 모델이 쉽게 구문 분석할 수 있는 형식으로 변환하는 것입니다. 데이터를 사전 처리한다는 것은 특정 열이나 행을 제거하고 누락된 값, 일관성이 없는 값 또는 중복 값을 처리함을 의미할 수 있습니다.

데이터 출처

라이프사이클 전반에 걸쳐 데이터의 출처와 기록을 추적하는 프로세스(예: 데이터 생성, 전송, 저장 방법).

데이터 주체

데이터를 수집 및 처리하는 개인입니다.

데이터 웨어하우스

분석과 같은 비즈니스 인텔리전스를 지원하는 데이터 관리 시스템. 데이터 웨어하우스에는 일반적으로 대량의 과거 데이터가 포함되며 일반적으로 쿼리 및 분석에 사용됩니다.

데이터 정의 언어(DDL)

데이터베이스에서 테이블 및 객체의 구조를 만들거나 수정하기 위한 명령문 또는 명령입니다.

데이터베이스 조작 언어(DML)

데이터베이스에서 정보를 수정(삽입, 업데이트 및 삭제)하기 위한 명령문 또는 명령입니다.

DDL

[데이터베이스 정의 언어](#)를 참조하십시오.

딥 앙상블

예측을 위해 여러 딥 러닝 모델을 결합하는 것입니다. 딥 앙상블을 사용하여 더 정확한 예측을 얻거나 예측의 불확실성을 추정할 수 있습니다.

딥 러닝

여러 계층의 인공 신경망을 사용하여 입력 데이터와 관심 대상 변수 간의 매핑을 식별하는 ML 하위 분야입니다.

defense-in-depth

네트워크와 그 안의 데이터 기밀성, 무결성 및 가용성을 보호하기 위해 컴퓨터 네트워크 전체에 일련의 보안 메커니즘과 제어를 신중하게 계층화하는 정보 보안 접근 방식입니다. 이 전략을 채택하면 AWS Organizations 구조의 여러 계층에 AWS 여러 컨트롤을 추가하여 리소스를 보호하는 데 도움이 됩니다. 예를 들어 다단계 인증, 네트워크 세분화, 암호화를 결합한 defense-in-depth 접근 방식을 사용할 수 있습니다.

위임된 관리자

에서 AWS Organizations 호환 가능한 서비스는 AWS 구성원 계정을 등록하여 조직의 계정을 관리하고 해당 서비스에 대한 권한을 관리할 수 있습니다. 이러한 계정을 해당 서비스의 위임된 관리자라고 합니다. 자세한 내용과 호환되는 서비스 목록은 AWS Organizations 설명서의 [AWS Organizations와 함께 사용할 수 있는 AWS 서비스](#)를 참조하십시오.

배포

대상 환경에서 애플리케이션, 새 기능 또는 코드 수정 사항을 사용할 수 있도록 하는 프로세스입니다. 배포에는 코드 베이스의 변경 사항을 구현한 다음 애플리케이션 환경에서 해당 코드베이스를 구축하고 실행하는 작업이 포함됩니다.

개발 환경

[환경](#)을 참조하십시오.

탐지 제어

이벤트 발생 후 탐지, 기록 및 알림을 수행하도록 설계된 보안 제어입니다. 이러한 제어는 기존의 예방적 제어를 우회한 보안 이벤트를 알리는 2차 방어선입니다. 자세한 내용은 Implementing security controls on AWS의 [Detective controls](#)를 참조하십시오.

개발 가치 흐름 매핑 (DVSM)

소프트웨어 개발 라이프사이클에서 속도와 품질에 부정적인 영향을 미치는 제약 조건을 식별하고 우선 순위를 지정하는 데 사용되는 프로세스입니다. DVSM은 원래 린 제조 방식을 위해 설계된 가치 흐름 매핑 프로세스를 확장합니다. 소프트웨어 개발 프로세스를 통해 가치를 창출하고 이동하는 데 필요한 단계와 팀에 중점을 둡니다.

디지털 트윈

건물, 공장, 산업 장비 또는 생산 라인과 같은 실제 시스템을 가상으로 표현한 것입니다. 디지털 트윈은 예측 유지 보수, 원격 모니터링, 생산 최적화를 지원합니다.

치수 표

[스타 스키마에서](#) 팩트 테이블의 양적 데이터에 대한 데이터 속성을 포함하는 작은 테이블입니다. 차원 테이블 속성은 일반적으로 텍스트처럼 동작하는 텍스트 필드 또는 불연속형 숫자입니다. 이러한 속성은 일반적으로 쿼리 제한, 필터링 및 결과 집합 레이블 지정에 사용됩니다.

재해

워크로드 또는 시스템이 기본 배포 위치에서 비즈니스 목표를 달성하지 못하게 방해하는 이벤트입니다. 이러한 이벤트는 자연재해, 기술적 오류, 의도하지 않은 구성 오류 또는 멀웨어 공격과 같은 사람의 행동으로 인한 결과일 수 있습니다.

재해 복구(DR)

[재해로 인한 다운타임과 데이터 손실을 최소화하기 위해 사용하는 전략과 프로세스입니다.](#) 자세한 내용은 [워크로드의 재해 복구 AWS: AWS Well-Architected 프레임워크에서의 클라우드 복구를 참조하십시오.](#)

DML

[데이터베이스](#) 조작 언어를 참조하십시오.

도메인 기반 설계

구성 요소를 각 구성 요소가 제공하는 진화하는 도메인 또는 핵심 비즈니스 목표에 연결하여 복잡한 소프트웨어 시스템을 개발하는 접근 방식입니다. 이 개념은 에릭 에반스에 의해 그의 저서인 도메인 기반 디자인: 소프트웨어 중심의 복잡성 해결(Boston: Addison-Wesley Professional, 2003)에서 소개되었습니다. Strangler Fig 패턴과 함께 도메인 기반 설계를 사용하는 방법에 대한 자세한 내용은 [컨테이너 및 Amazon API Gateway를 사용하여 기존의 Microsoft ASP.NET\(ASMX\) 웹 서비스를 점진적으로 현대화하는 방법](#)을 참조하십시오.

DR

[재해 복구](#)를 참조하십시오.

드리프트 감지

기존 구성으로부터의 편차 추적. 예를 들어 [시스템 리소스의 편차를 감지하는 AWS CloudFormation](#) 데 사용하거나 거버넌스 요구 사항 준수에 영향을 미칠 수 있는 [착륙 지대의 변경 사항을 탐지하는 AWS Control Tower](#) 데 사용할 수 있습니다.

DVSM

[개발 가치 흐름 매핑](#) 참조하십시오.

E

EDA

[탐색적 데이터 분석](#) 참조하십시오.

엣지 컴퓨팅

IoT 네트워크의 엣지에서 스마트 디바이스의 컴퓨팅 성능을 개선하는 기술 [클라우드 컴퓨팅과](#) 비교할 때 엣지 컴퓨팅은 통신 대기 시간을 줄이고 응답 시간을 개선할 수 있습니다.

암호화

사람이 읽을 수 있는 일반 텍스트 데이터를 암호문으로 변환하는 컴퓨팅 프로세스입니다.

암호화 키

암호화 알고리즘에 의해 생성되는 무작위 비트의 암호화 문자열입니다. 키의 길이는 다양할 수 있으며 각 키는 예측할 수 없고 고유하게 설계되었습니다.

엔디안

컴퓨터 메모리에 바이트가 저장되는 순서입니다. 빅 엔디안 시스템은 가장 중요한 바이트를 먼저 저장합니다. 리틀 엔디안 시스템은 가장 덜 중요한 바이트를 먼저 저장합니다.

엔드포인트

[서비스](#) 엔드포인트를 참조하십시오.

엔드포인트 서비스

Virtual Private Cloud(VPC)에서 호스팅하여 다른 사용자와 공유할 수 있는 서비스입니다. 다른 주체 AWS 계정 또는 AWS Identity and Access Management (IAM) 보안 주체에 권한을 부여하여 엔드포인트 서비스를 생성하고 권한을 부여할 수 있습니다. AWS PrivateLink 이러한 계정 또는 보안 주체는 인터페이스 VPC 엔드포인트를 생성하여 엔드포인트 서비스에 비공개로 연결할 수 있습니다.

다. 자세한 내용은 Amazon Virtual Private Cloud(VPC) 설명서의 [엔드포인트 서비스 생성](#)을 참조하십시오.

ERP (전사적 자원 관리)

기업의 주요 비즈니스 프로세스 (예: 회계, [MES](#), 프로젝트 관리) 를 자동화하고 관리하는 시스템입니다.

봉투 암호화

암호화 키를 다른 암호화 키로 암호화하는 프로세스입니다. 자세한 내용은 AWS Key Management Service (AWS KMS) [설명서의 봉투 암호화](#)를 참조하십시오.

환경

실행 중인 애플리케이션의 인스턴스입니다. 다음은 클라우드 컴퓨팅의 일반적인 환경 유형입니다.

- 개발 환경 - 애플리케이션 유지 관리를 담당하는 핵심 팀만 사용할 수 있는 실행 중인 애플리케이션의 인스턴스입니다. 개발 환경은 변경 사항을 상위 환경으로 승격하기 전에 테스트하는 데 사용됩니다. 이러한 유형의 환경을 테스트 환경이라고도 합니다.
- 하위 환경 - 초기 빌드 및 테스트에 사용되는 환경을 비롯한 애플리케이션의 모든 개발 환경입니다.
- 프로덕션 환경 - 최종 사용자가 액세스할 수 있는 실행 중인 애플리케이션의 인스턴스입니다. CI/CD 파이프라인에서 프로덕션 환경이 마지막 배포 환경입니다.
- 상위 환경 - 핵심 개발 팀 이외의 사용자가 액세스할 수 있는 모든 환경입니다. 프로덕션 환경, 프로덕션 이전 환경 및 사용자 수용 테스트를 위한 환경이 여기에 포함될 수 있습니다.

에픽

애자일 방법론에서 작업을 구성하고 우선순위를 정하는 데 도움이 되는 기능적 범주입니다. 에픽은 요구 사항 및 구현 작업에 대한 개괄적인 설명을 제공합니다. 예를 들어 AWS CAF 보안 에픽에는 ID 및 액세스 관리, 탐지 제어, 인프라 보안, 데이터 보호, 사고 대응 등이 포함됩니다. AWS 마이그레이션 전략의 에픽에 대한 자세한 내용은 [프로그램 구현 가이드](#)를 참조하십시오.

ERP

[엔터프라이즈 리소스 계획을](#) 참조하십시오.

탐색 데이터 분석(EDA)

데이터 세트를 분석하여 주요 특성을 파악하는 프로세스입니다. 데이터를 수집 또는 집계한 다음 초기 조사를 수행하여 패턴을 찾고, 이상을 탐지하고, 가정을 확인합니다. EDA는 요약 통계를 계산하고 데이터 시각화를 생성하여 수행됩니다.

F

팩트 테이블

[스타 스키마의](#) 중앙 테이블. 비즈니스 운영에 대한 정량적 데이터를 저장합니다. 일반적으로 팩트 테이블에는 측정값이 포함된 열과 차원 테이블의 외부 키가 포함된 열 등 두 가지 유형의 열이 포함됩니다.

빨리 실패하세요

빈번하고 점진적인 테스트를 통해 개발 라이프사이클을 단축하는 철학. 이는 애자일 접근 방식의 중요한 부분입니다.

장애 격리 경계

장애 영향을 제한하고 워크로드의 복원력을 개선하는 데 도움이 되는 가용 영역 AWS 리전, 컨트를 플레인 또는 데이터 플레인과 같은 경계 AWS 클라우드자세한 내용은 [AWS 장애 격리](#) 경계를 참조하십시오.

기능 브랜치

[브랜치를](#) 참조하십시오.

기능

예측에 사용하는 입력 데이터입니다. 예를 들어, 제조 환경에서 기능은 제조 라인에서 주기적으로 캡처되는 이미지일 수 있습니다.

기능 중요도

모델의 예측에 특성이 얼마나 중요한지를 나타냅니다. 이는 일반적으로 SHAP(Shapley Additive Descriptions) 및 통합 그래디언트와 같은 다양한 기법을 통해 계산할 수 있는 수치 점수로 표현됩니다. 자세한 내용은 [다음은AWS사용한 기계 학습 모델 해석 가능성을](#) 참조하십시오.

기능 변환

추가 소스로 데이터를 보강하거나, 값을 조정하거나, 단일 데이터 필드에서 여러 정보 세트를 추출하는 등 ML 프로세스를 위해 데이터를 최적화하는 것입니다. 이를 통해 ML 모델이 데이터를 활용할 수 있습니다. 예를 들어, 날짜 '2021-05-27 00:15:37'을 '2021년', '5월', '목', '15일'로 분류하면 학습 알고리즘이 다양한 데이터 구성 요소와 관련된 미묘한 패턴을 학습하는 데 도움이 됩니다.

FGAC

[세분화된 액세스 제어](#)를 참조하십시오.

세분화된 액세스 제어(FGAC)

여러 조건을 사용하여 액세스 요청을 허용하거나 거부합니다.

플래시컷 마이그레이션

단계별 접근 방식 대신 [변경 데이터 캡처를 통한 지속적인 데이터](#) 복제를 통해 최단 시간에 데이터를 마이그레이션하는 데이터베이스 마이그레이션 방법입니다. 목표는 가동 중지 시간을 최소화하는 것입니다.

G

지리적 차단

[지리적 제한](#)을 참조하십시오.

지리적 제한(지리적 차단)

CloudFrontAmazon에서는 특정 국가의 사용자가 콘텐츠 배포에 액세스하지 못하도록 하는 옵션을 제공합니다. 허용 목록 또는 차단 목록을 사용하여 승인된 국가와 차단된 국가를 지정할 수 있습니다. 자세한 내용은 [설명서의 콘텐츠의 지리적 배포 제한](#)을 참조하십시오. CloudFront

Gitflow 워크플로

하위 환경과 상위 환경이 소스 코드 리포지토리의 서로 다른 브랜치를 사용하는 방식입니다.

Gitflow 워크플로는 레거시로 간주되며 [트렁크 기반 워크플로는](#) 현대적이고 선호되는 접근 방식입니다.

브라운필드 전략

새로운 환경에서 기존 인프라의 부재 시스템 아키텍처에 대한 그린필드 전략을 채택할 때 [브라운필드](#)라고도 하는 기존 인프라와의 호환성 제한 없이 모든 새로운 기술을 선택할 수 있습니다. 기존 인프라를 확장하는 경우 브라운필드 전략과 그린필드 전략을 혼합할 수 있습니다.

가드레일

조직 단위(OU) 전체에서 리소스, 정책 및 규정 준수를 관리하는 데 도움이 되는 중요 규칙입니다. 예방 가드레일은 규정 준수 표준에 부합하도록 정책을 시행하며, 서비스 제어 정책과 IAM 권한 경계를 사용하여 구현됩니다. 탐지 가드레일은 정책 위반 및 규정 준수 문제를 감지하고 해결을 위한 알림을 생성하며, 이들은, Amazon AWS Config AWS Security Hub GuardDuty AWS Trusted Advisor, Amazon Inspector 및 사용자 지정 AWS Lambda 검사를 사용하여 구현됩니다.

H

하

[고가용성을](#) 확인하세요.

이기종 데이터베이스 마이그레이션

다른 데이터베이스 엔진을 사용하는 대상 데이터베이스로 소스 데이터베이스 마이그레이션(예: Oracle에서 Amazon Aurora로) 이기종 마이그레이션은 일반적으로 리아키텍트 작업의 일부이며 스키마를 변환하는 것은 복잡한 작업일 수 있습니다. AWS 는 스키마 변환에 도움이 되는 [AWS SCT](#)를 제공합니다.

높은 가용성(HA)

문제나 재해 발생 시 개입 없이 지속적으로 운영할 수 있는 워크로드의 능력. HA 시스템은 자동으로 장애 조치되고, 지속적으로 고품질 성능을 제공하고, 성능에 미치는 영향을 최소화하면서 다양한 부하와 장애를 처리하도록 설계되었습니다.

히스토리언 현대화

제조 산업의 요구 사항을 더 잘 충족하도록 운영 기술(OT) 시스템을 현대화하고 업그레이드하는 데 사용되는 접근 방식입니다. 히스토리언은 공장의 다양한 출처에서 데이터를 수집하고 저장하는 데 사용되는 일종의 데이터베이스입니다.

동종 데이터베이스 마이그레이션

동일한 데이터베이스 엔진을 공유하는 대상 데이터베이스로 소스 데이터베이스 마이그레이션(예: Microsoft SQL Server에서 Amazon RDS for SQL Server로) 동종 마이그레이션은 일반적으로 리호스팅 또는 리플랫폼 작업의 일부입니다. 네이티브 데이터베이스 유틸리티를 사용하여 스키마를 마이그레이션할 수 있습니다.

핫 데이터

자주 액세스하는 데이터(예: 실시간 데이터 또는 최근 번역 데이터). 일반적으로 이 데이터에는 빠른 쿼리 응답을 제공하기 위한 고성능 스토리지 계층 또는 클래스가 필요합니다.

핫픽스

프로덕션 환경의 중요한 문제를 해결하기 위한 긴급 수정입니다. 긴급성 때문에 핫픽스는 일반적으로 일반적인 DevOps 릴리스 워크플로 외부에서 만들어집니다.

하이퍼케어 기간

전환 직후 마이그레이션 팀이 문제를 해결하기 위해 클라우드에서 마이그레이션된 애플리케이션을 관리하고 모니터링하는 기간입니다. 일반적으로 이 기간은 1~4일입니다. 하이퍼케어 기간이 끝나면 마이그레이션 팀은 일반적으로 애플리케이션에 대한 책임을 클라우드 운영 팀에 넘깁니다.

I

IaC

[인프라를 코드로 보세요.](#)

자격 증명 기반 정책

환경 내에서 권한을 정의하는 하나 이상의 IAM 보안 주체에 연결된 정책입니다. AWS 클라우드 유휴 애플리케이션

90일 동안 평균 CPU 및 메모리 사용량이 5~20%인 애플리케이션입니다. 마이그레이션 프로젝트에서는 이러한 애플리케이션을 사용 중지하거나 온프레미스에 유지하는 것이 일반적입니다.

IIoT

[산업용 사물 인터넷을 참조하십시오.](#)

불변의 인프라

기존 인프라를 업데이트, 패치 또는 수정하는 대신 프로덕션 워크로드용 새 인프라를 배포하는 모델입니다. [변경 불가능한 인프라는 기본적으로 변경 가능한 인프라보다 더 일관되고 안정적이며 예측 가능합니다.](#) 자세한 내용은 Well-Architected AWS 프레임워크의 [변경 불가능한 인프라를 사용한 배포](#) 모범 사례를 참조하십시오.

인바운드(수신) VPC

AWS 다중 계정 아키텍처에서 VPC는 애플리케이션 외부에서 네트워크 연결을 허용, 검사 및 라우팅합니다. [AWS Security Reference Architecture](#)에서는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPC로 네트워크 계정을 설정할 것을 권장합니다.

중분 마이그레이션

한 번에 전체 전환을 수행하는 대신 애플리케이션을 조금씩 마이그레이션하는 전환 전략입니다. 예를 들어, 처음에는 소수의 마이크로서비스나 사용자만 새 시스템으로 이동할 수 있습니다. 모든 것

이 제대로 작동하는지 확인한 후에는 레거시 시스템을 폐기할 수 있을 때까지 추가 마이크로서비스 또는 사용자를 점진적으로 이동할 수 있습니다. 이 전략을 사용하면 대규모 마이그레이션과 관련된 위험을 줄일 수 있습니다.

Industry 4.0

[Klaus Schwab](#)이 연결성, 실시간 데이터, 자동화, 분석 및 AI/ML의 발전을 통한 제조 프로세스의 현대화를 지칭하기 위해 2016년 도입한 용어입니다.

인프라

애플리케이션의 환경 내에 포함된 모든 리소스와 자산입니다.

코드형 인프라(IaC)

구성 파일 세트를 통해 애플리케이션의 인프라를 프로비저닝하고 관리하는 프로세스입니다. IaC는 새로운 환경의 반복 가능성, 신뢰성 및 일관성을 위해 인프라 관리를 중앙 집중화하고, 리소스를 표준화하고, 빠르게 확장할 수 있도록 설계되었습니다.

산업용 사물 인터넷(IIoT)

제조, 에너지, 자동차, 의료, 생명과학, 농업 등의 산업 부문에서 인터넷에 연결된 센서 및 디바이스의 사용 자세한 내용은 [산업용 사물 인터넷\(IoT\) 디지털 트랜스포메이션 전략 구축](#)을 참조하십시오.

검사 VPC

AWS 다중 계정 아키텍처에서 VPC (동일하거나 AWS 리전다른), 인터넷 및 온프레미스 네트워크 간의 네트워크 트래픽 검사를 관리하는 중앙 집중식 VPC입니다. [AWS Security Reference Architecture](#)에서는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPC로 네트워크 계정을 설정할 것을 권장합니다.

사물 인터넷(IoT)

인터넷이나 로컬 통신 네트워크를 통해 다른 디바이스 및 시스템과 통신하는 센서 또는 프로세서가 내장된 연결된 물리적 객체의 네트워크 자세한 내용은 [IoT란?](#)을 참조하십시오.

해석력

모델의 예측이 입력에 따라 어떻게 달라지는지를 사람이 이해할 수 있는 정도를 설명하는 기계 학습 모델의 특성입니다. 자세한 내용은 [Machine learning model interpretability with AWS](#)를 참조하십시오.

IoT

[사물 인터넷을 참조하십시오.](#)

IT 정보 라이브러리(TIL)

IT 서비스를 제공하고 이러한 서비스를 비즈니스 요구 사항에 맞게 조정하기 위한 일련의 모범 사례 ITIL은 ITSM의 기반을 제공합니다.

IT 서비스 관리(TSM)

조직의 IT 서비스 설계, 구현, 관리 및 지원과 관련된 활동 클라우드 운영을 ITSM 도구와 통합하는 방법에 대한 자세한 내용은 [운영 통합 가이드](#)를 참조하십시오.

ITIL

[IT 정보 라이브러리를](#) 참조하십시오.

ITSM

[IT 서비스 관리를](#) 참조하십시오.

L

레이블 기반 액세스 제어(LBAC)

사용자 및 데이터 자체에 각각 보안 레이블 값을 명시적으로 할당하는 필수 액세스 제어(MAC)를 구현한 것입니다. 사용자 보안 레이블과 데이터 보안 레이블 간의 교차 부분에 따라 사용자가 볼 수 있는 행과 열이 결정됩니다.

랜딩 존

Landing Zone은 확장 가능하고 안전한 잘 설계된 다중 계정 AWS 환경입니다. 조직은 여기에서부터 보안 및 인프라 환경에 대한 확신을 가지고 워크로드와 애플리케이션을 신속하게 시작하고 배포할 수 있습니다. 랜딩 존에 대한 자세한 내용은 [안전하고 확장 가능한 다중 계정 AWS 환경 설정](#)을 참조하십시오.

대규모 마이그레이션

300대 이상의 서버 마이그레이션입니다.

LBAC

[레이블 기반 액세스 제어를](#) 참조하십시오.

최소 권한

작업을 수행하는 데 필요한 최소 권한을 부여하는 보안 모범 사례입니다. 자세한 내용은 IAM 설명서의 [최소 권한 적용](#)을 참조하십시오.

리프트 앤드 시프트

[7 R](#)을 참조하십시오.

리틀 엔디안 시스템

가장 덜 중요한 바이트를 먼저 저장하는 시스템입니다. [엔디안](#) 참조.

하위 환경

[환경 참조](#).

M

기계 학습(ML)

패턴 인식 및 학습에 알고리즘과 기법을 사용하는 인공지능의 한 유형입니다. ML은 사물 인터넷 (IoT) 데이터와 같은 기록된 데이터를 분석하고 학습하여 패턴을 기반으로 통계 모델을 생성합니다. 자세한 내용은 [기계 학습](#)을 참조하십시오.

기본 브랜치

[브랜치](#) 참조.

악성 코드

컴퓨터 보안 또는 개인 정보를 침해하도록 설계된 소프트웨어 멀웨어는 컴퓨터 시스템을 방해하거나, 민감한 정보를 유출하거나, 무단 액세스를 얻을 수 있습니다. 멀웨어의 예로는 바이러스, 웜, 랜섬웨어, 트로이 목마, 스파이웨어, 키로거 등이 있습니다.

매니지드 서비스

AWS 서비스 인프라 계층, 운영 체제 및 플랫폼을 AWS 운영하며 사용자는 엔드포인트에 액세스하여 데이터를 저장하고 검색합니다. 관리형 서비스의 예로는 아마존 심플 스토리지 서비스 (Amazon S3) 와 아마존 DynamoDB가 있습니다. 이러한 서비스를 추상화된 서비스라고도 합니다.

제조 실행 시스템 (MES)

제조 현장에서 원자재를 완제품으로 전환하는 생산 프로세스를 추적, 모니터링, 문서화 및 제어하기 위한 소프트웨어 시스템입니다.

MAP

[Migration Acceleration 프로그램](#)을 참조하십시오.

기구

도구를 만들고 도구 채택을 유도한 다음 결과를 검토하여 조정하는 전체 프로세스입니다. 메커니즘은 작동하면서 자체적으로 강화되고 개선되는 사이클입니다. 자세한 내용은 [AWS Well-Architected 프레임워크에서의 메커니즘 구축을](#) 참조하십시오.

멤버 계정

조직의 일부인 관리 계정을 AWS 계정 제외한 모든 계정 AWS Organizations 하나의 계정은 한 번에 하나의 조직 멤버만 될 수 있습니다.

MES

[제조 실행 시스템을](#) 참조하십시오.

메시지 큐 텔레메트리 전송 (MQTT)

[퍼블리시/구독 패턴을 기반으로 하는 리소스가 제한된 IoT 디바이스를 위한 경량 machine-to-machine \(M2M\) 통신 프로토콜입니다.](#)

마이크로서비스

잘 정의된 API를 통해 통신하고 일반적으로 소규모 자체 팀이 소유하는 소규모 독립 서비스입니다. 예를 들어, 보험 시스템에는 영업, 마케팅 등의 비즈니스 역량이나 구매, 청구, 분석 등의 하위 영역에 매핑되는 마이크로 서비스가 포함될 수 있습니다. 마이크로서비스의 이점으로 민첩성, 유연한 확장, 손쉬운 배포, 재사용 가능한 코드, 복원력 등이 있습니다. [자세한 내용은 서버리스 서비스를 사용하여 마이크로서비스 통합을](#) 참조하십시오. [AWS](#)

마이크로서비스 아키텍처

각 애플리케이션 프로세스를 마이크로서비스로 실행하는 독립 구성 요소를 사용하여 애플리케이션을 구축하는 접근 방식입니다. 이러한 마이크로서비스는 경량 API를 사용하여 잘 정의된 인터페이스를 통해 통신합니다. 애플리케이션의 특정 기능에 대한 수요에 맞게 이 아키텍처의 각 마이크로서비스를 업데이트, 배포 및 조정할 수 있습니다. 자세한 내용은 마이크로서비스 [구현을](#) 참조하십시오. [AWS](#)

Migration Acceleration Program(MAP)

조직이 클라우드로 전환하기 위한 강력한 운영 기반을 구축하고 초기 마이그레이션 비용을 상쇄할 수 있도록 컨설팅 지원, 교육 및 서비스를 제공하는 AWS 프로그램입니다. MAP에는 레거시 마이그레이션을 체계적인 방식으로 실행하기 위한 마이그레이션 방법론과 일반적인 마이그레이션 시나리오를 자동화하고 가속화하는 도구 세트가 포함되어 있습니다.

대규모 마이그레이션

애플리케이션 포트폴리오의 대다수를 웨이브를 통해 클라우드로 이동하는 프로세스로, 각 웨이브에서 더 많은 애플리케이션이 더 빠른 속도로 이동합니다. 이 단계에서는 이전 단계에서 배운 모범 사례와 교훈을 사용하여 팀, 도구 및 프로세스의 마이그레이션 팩토리를 구현하여 자동화 및 민첩한 제공을 통해 워크로드 마이그레이션을 간소화합니다. 이것은 [AWS 마이그레이션 전략](#)의 세 번째 단계입니다.

마이그레이션 팩토리

자동화되고 민첩한 접근 방식을 통해 워크로드 마이그레이션을 간소화하는 다기능 팀입니다. 마이그레이션 팩토리 팀에는 일반적으로 운영, 비즈니스 분석가 및 소유자, 마이그레이션 엔지니어, 개발자 및 스프린트에서 일하는 DevOps 전문가가 포함됩니다. 엔터프라이즈 애플리케이션 포트폴리오의 20~50%는 공장 접근 방식으로 최적화할 수 있는 반복되는 패턴으로 구성되어 있습니다. 자세한 내용은 이 콘텐츠 세트의 [클라우드 마이그레이션 팩토리 가이드](#)와 [마이그레이션 팩토리에 대한 설명](#)을 참조하십시오.

마이그레이션 메타데이터

마이그레이션을 완료하는 데 필요한 애플리케이션 및 서버에 대한 정보 각 마이그레이션 패턴에는 서로 다른 마이그레이션 메타데이터 세트가 필요합니다. 마이그레이션 메타데이터의 예로는 대상 서브넷, 보안 그룹, 계정 등이 있습니다. AWS

마이그레이션 패턴

사용되는 마이그레이션 전략, 마이그레이션 대상, 마이그레이션 애플리케이션 또는 서비스를 자세히 설명하는 반복 가능한 마이그레이션 작업입니다. 예: 애플리케이션 마이그레이션 서비스를 사용하여 Amazon EC2로 AWS 마이그레이션을 재호스팅합니다.

Migration Portfolio Assessment(MPA)

로 마이그레이션하기 위한 비즈니스 사례를 검증하기 위한 정보를 제공하는 온라인 도구입니다. AWS 클라우드 MPA는 상세한 포트폴리오 평가(서버 적정 규모 조정, 가격 책정, TCO 비교, 마이그레이션 비용 분석)와 마이그레이션 계획(애플리케이션 데이터 분석 및 데이터 수집, 애플리케이션 그룹화, 마이그레이션 우선순위 지정, 웨이브 계획)을 제공합니다. [MPA 도구](#) (로그인 필요) 는 모든 컨설턴트와 APN 파트너 AWS 컨설턴트에게 무료로 제공됩니다.

마이그레이션 준비 상태 평가(MRA)

CAF를 사용하여 조직의 클라우드 준비 상태에 대한 통찰력을 얻고, 강점과 약점을 파악하고, 식별된 격차를 해소하기 위한 실행 계획을 수립하는 프로세스입니다. AWS 자세한 내용은 [마이그레이션 준비 가이드](#)를 참조하십시오. MRA는 [AWS 마이그레이션 전략](#)의 첫 번째 단계입니다.

마이그레이션 전략

워크로드를 로 마이그레이션하는 데 사용된 접근 방식. AWS 클라우드자세한 내용은 이 용어집의 [7R 항목 및 대규모 마이그레이션 가속화를 위한 조직 동원을 참조하십시오.](#)

ML

[기계 학습을 참조하십시오.](#)

현대화

비용을 절감하고 효율성을 높이고 혁신을 활용하기 위해 구식(레거시 또는 모놀리식) 애플리케이션과 해당 인프라를 클라우드의 민첩하고 탄력적이고 가용성이 높은 시스템으로 전환하는 것입니다. 자세한 내용은 [의 AWS 클라우드애플리케이션 현대화 전략을 참조하십시오.](#)

현대화 준비 상태 평가

조직 애플리케이션의 현대화 준비 상태를 파악하고, 이점, 위험 및 종속성을 식별하고, 조직이 해당 애플리케이션의 향후 상태를 얼마나 잘 지원할 수 있는지를 확인하는 데 도움이 되는 평가입니다. 평가 결과는 대상 아키텍처의 청사진, 현대화 프로세스의 개발 단계와 마일스톤을 자세히 설명하는 로드맵 및 파악된 격차를 해소하기 위한 실행 계획입니다. 자세한 내용은 [에서 애플리케이션의 현대화 준비 상태 평가를 참조하십시오.](#) AWS 클라우드

모놀리식 애플리케이션(모놀리식 유형)

긴밀하게 연결된 프로세스를 사용하여 단일 서비스로 실행되는 애플리케이션입니다. 모놀리식 애플리케이션에는 몇 가지 단점이 있습니다. 한 애플리케이션 기능에 대한 수요가 급증하면 전체 아키텍처 규모를 조정해야 합니다. 코드 베이스가 커지면 모놀리식 애플리케이션의 기능을 추가하거나 개선하는 것도 더 복잡해집니다. 이러한 문제를 해결하기 위해 마이크로서비스 아키텍처를 사용할 수 있습니다. 자세한 내용은 [마이크로서비스로 모놀리식 유형 분해](#)를 참조하십시오.

MPA

[마이그레이션 포트폴리오 평가를 참조하십시오.](#)

MQTT

[메시지 큐 원격 분석 전송을 참조하십시오.](#)

멀티클래스 분류

여러 클래스에 대한 예측(2개 이상의 결과 중 하나 예측)을 생성하는 데 도움이 되는 프로세스입니다. 예를 들어, ML 모델이 '이 제품은 책인가요, 자동차인가요, 휴대폰인가요?' 또는 '이 고객이 가장 관심을 갖는 제품 범주는 무엇인가요?'라고 물을 수 있습니다.

변경 가능한 인프라

프로덕션 워크로드를 위해 기존 인프라를 업데이트하고 수정하는 모델입니다. 일관성, 안정성 및 예측 가능성을 개선하기 위해 AWS Well-Architected Framework는 [변경 불가능한](#) 인프라를 모범 사례로 사용할 것을 권장합니다.

O

OAC

[원본 액세스 제어를 참조하십시오.](#)

좋아요

[원본 액세스 ID를 참조하십시오.](#)

OCM

[조직 변경 관리를 참조하십시오.](#)

오프라인 마이그레이션

마이그레이션 프로세스 중 소스 워크로드가 중단되는 마이그레이션 방법입니다. 이 방법은 가동 중지 증가를 수반하며 일반적으로 작고 중요하지 않은 워크로드에 사용됩니다.

O

[운영 통합을 참조하십시오.](#)

안녕하세요.

[운영 수준 계약을 참조하십시오.](#)

온라인 마이그레이션

소스 워크로드를 오프라인 상태로 전환하지 않고 대상 시스템에 복사하는 마이그레이션 방법입니다. 워크로드에 연결된 애플리케이션은 마이그레이션 중에도 계속 작동할 수 있습니다. 이 방법은 가동 중지 차단 또는 최소화를 수반하며 일반적으로 중요한 프로덕션 워크로드에 사용됩니다.

OPC-UA

[오픈 프로세스 커뮤니케이션 - 통합](#) 아키텍처를 참조하십시오.

오픈 프로세스 커뮤니케이션 - 통합 아키텍처 (OPC-UA)

산업 machine-to-machine 자동화를 위한 (M2M) 통신 프로토콜. OPC-UA는 데이터 암호화, 인증 및 권한 부여 체계와 함께 상호 운용성 표준을 제공합니다.

운영 수준 협약(OLA)

서비스 수준에 관한 계약(SLA)을 지원하기 위해 직무 IT 그룹이 서로에게 제공하기로 약속한 내용을 명확히 하는 계약입니다.

운영 준비 검토 (ORR)

인시던트 및 발생 가능한 실패의 범위를 이해, 평가, 예방 또는 줄이는 데 도움이 되는 질문 및 관련 모범 사례로 구성된 체크리스트입니다. 자세한 내용은 Well-Architected AWS 프레임워크의 [운영 준비 상태 검토 \(ORR\)](#) 를 참조하십시오.

운영 기술 (OT)

물리적 환경과 함께 작동하여 산업 운영, 장비 및 인프라를 제어하는 하드웨어 및 소프트웨어 시스템. 제조 분야에서는 OT와 정보 기술 (IT) 시스템의 통합이 [인더스트리 4.0](#) 혁신의 핵심 초점입니다.

운영 통합(OI)

클라우드에서 운영을 현대화하는 프로세스로 준비 계획, 자동화 및 통합을 수반합니다. 자세한 내용은 [운영 통합 가이드](#)를 참조하십시오.

조직 트레일

이를 통해 AWS CloudTrail 생성되는 트레일은 조직 AWS 계정 내 모든 사용자의 모든 이벤트를 기록합니다. AWS Organizations이 트레일은 조직에 속한 각 AWS 계정에 생성되고 각 계정의 활동을 추적합니다. 자세한 내용은 CloudTrail 설명서에서 [조직을 위한 트레일 만들기를](#) 참조하십시오.

조직 변경 관리(OCM)

사람, 문화 및 리더십 관점에서 중대하고 파괴적인 비즈니스 혁신을 관리하기 위한 프레임워크입니다. OCM은 변화 채택을 가속화하고, 과도기적 문제를 해결하고, 문화 및 조직적 변화를 주도함으로써 조직이 새로운 시스템 및 전략을 준비하고 전환할 수 있도록 지원합니다. 클라우드 채택 프로젝트에 필요한 변화 속도 때문에 AWS 마이그레이션 전략에서는 이 프레임워크를 사용자 가속화라고 합니다. 자세한 내용은 [사용 가이드](#)를 참조하십시오.

오리진 액세스 제어(OAC)

CloudFront에서는 Amazon Simple Storage Service (Amazon S3) 콘텐츠의 보안을 위해 액세스를 제한하는 향상된 옵션을 제공합니다. OAC는 모든 S3 버킷 AWS 리전, AWS KMS (SSE-KMS) 를 사용한 서버 측 암호화, S3 버킷에 대한 동적 및 요청을 모두 지원합니다. PUT DELETE

오리진 액세스 ID(OAI)

CloudFront에서는 Amazon S3 콘텐츠 보안을 위해 액세스를 제한하는 옵션입니다. OAI를 사용하면 Amazon S3가 인증할 수 있는 보안 주체를 CloudFront 생성합니다. 인증된 보안 주체는 특정 배

포를 통해서만 S3 버킷의 콘텐츠에 액세스할 수 있습니다. CloudFront 더 세분화되고 향상된 액세스 제어를 제공하는 [OAC](#)도 참조하십시오.

또는

[운영 준비 상태](#) 검토를 참조하십시오.

아니요

[운영 기술을](#) 참조하십시오.

아웃바운드(송신) VPC

AWS 다중 계정 아키텍처에서 애플리케이션 내에서 시작되는 네트워크 연결을 처리하는 VPC입니다. [AWS Security Reference Architecture](#)에서는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPC로 네트워크 계정을 설정할 것을 권장합니다.

P

권한 경계

사용자나 역할이 가질 수 있는 최대 권한을 설정하기 위해 IAM 보안 주체에 연결되는 IAM 관리 정책입니다. 자세한 내용은 IAM 설명서의 [권한 경계](#)를 참조하십시오.

개인 식별 정보(PII)

직접 보거나 다른 관련 데이터와 함께 짝을 지을 때 개인의 신원을 합리적으로 추론하는 데 사용할 수 있는 정보입니다. PII의 예로는 이름, 주소, 연락처 정보 등이 있습니다.

PII

[개인 식별](#) 정보를 참조하십시오.

플레이북

클라우드에서 핵심 운영 기능을 제공하는 등 마이그레이션과 관련된 작업을 캡처하는 일련의 사전 정의된 단계입니다. 플레이북은 스크립트, 자동화된 런북 또는 현대화된 환경을 운영하는 데 필요한 프로세스나 단계 요약의 형태를 취할 수 있습니다.

PLC

[프로그래머블 로직 컨트롤러](#)를 참조하십시오.

PLM

[제품 라이프사이클 관리](#)를 참조하십시오.

정책

권한을 정의 ([ID 기반 정책 참조](#)) 하거나, 액세스 조건을 지정 ([리소스 기반 정책 참조](#)) 하거나, 조직 내 모든 계정에 대한 최대 권한을 정의 AWS Organizations ([서비스 제어 정책 참조](#)) 할 수 있는 개체입니다.

다국어 지속성

데이터 액세스 패턴 및 기타 요구 사항을 기반으로 독립적으로 마이크로서비스의 데이터 스토리지 기술 선택. 마이크로서비스가 동일한 데이터 스토리지 기술을 사용하는 경우 구현 문제가 발생하거나 성능이 저하될 수 있습니다. 요구 사항에 가장 적합한 데이터 스토어를 사용하면 마이크로서비스를 더 쉽게 구현하고 성능과 확장성을 높일 수 있습니다. 자세한 내용은 [마이크로서비스에서 데이터 지속성 활성화](#)를 참조하십시오.

포트폴리오 평가

마이그레이션을 계획하기 위해 애플리케이션 포트폴리오를 검색 및 분석하고 우선순위를 정하는 프로세스입니다. 자세한 내용은 [마이그레이션 준비 상태 평가](#)를 참조하십시오.

조건자

일반적으로 조항에 있는 true false OR를 반환하는 쿼리 조건입니다. WHERE

조건부 푸시다운

전송하기 전에 쿼리의 데이터를 필터링하는 데이터베이스 쿼리 최적화 기법입니다. 이렇게 하면 관계형 데이터베이스에서 검색하고 처리해야 하는 데이터의 양이 줄어들고 쿼리 성능이 향상됩니다.

예방적 제어

이벤트 발생을 방지하도록 설계된 보안 제어입니다. 이 제어는 네트워크에 대한 무단 액세스나 원치 않는 변경을 방지하는 데 도움이 되는 1차 방어선입니다. 자세한 내용은 Implementing security controls on AWS의 [Preventative controls](#)를 참조하십시오.

보안 주체

작업을 수행하고 리소스에 액세스할 수 있는 AWS 있는 엔티티 이 엔티티는 일반적으로 IAM 역할의 루트 사용자 또는 사용자입니다. AWS 계정자세한 내용은 IAM 설명서의 [역할 용어 및 개념](#)의 보안 주체를 참조하십시오.

개인 정보 보호 중심 설계

전체 엔지니어링 프로세스에서 개인 정보를 고려하는 시스템 엔지니어링에서의 접근 방식입니다.

프라이빗 호스팅 영역

Amazon Route 53에서 하나 이상의 VPC 내 도메인과 하위 도메인에 대한 DNS 쿼리에 응답하는 방법에 대한 정보가 담긴 컨테이너입니다. 자세한 내용은 Route 53 설명서의 [프라이빗 호스팅 영역 작업을 참조하십시오](#).

사전 예방 제어

규정을 준수하지 않는 리소스의 배포를 방지하도록 설계된 [보안 제어입니다](#). 이러한 컨트롤은 리소스를 프로비저닝하기 전에 리소스를 스캔합니다. 리소스가 컨트롤과 호환되지 않으면 프로비저닝되지 않습니다. 자세한 내용은 AWS Control Tower 설명서의 [컨트롤 참조 안내서](#)를 참조하고 보안 제어 구현의 [사전 제어를](#) 참조하십시오. AWS

제품 라이프사이클 관리 (PLM)

설계, 개발, 출시부터 성장 및 성숙도, 폐기 및 제거에 이르는 전체 라이프사이클에 걸쳐 제품에 대한 데이터 및 프로세스를 관리하는 것입니다.

프로덕션 환경

[환경](#)을 참조하십시오.

프로그래머블 로직 컨트롤러 (PLC)

제조 분야에서 기계를 모니터링하고 제조 프로세스를 자동화하는 매우 안정적이고 적응력이 뛰어난 컴퓨터입니다.

가명화

데이터세트의 개인 식별자를 자리 표시자 값으로 바꾸는 프로세스입니다. 가명화는 개인 정보를 보호하는 데 도움이 될 수 있습니다. 가명화된 데이터는 여전히 개인 데이터로 간주됩니다.

게시/구독 (게시/구독)

마이크로서비스 간의 비동기 통신을 통해 확장성과 응답성을 개선할 수 있는 패턴입니다. 예를 들어 마이크로서비스 기반 [MES에서](#) 마이크로서비스는 다른 마이크로서비스가 구독할 수 있는 채널에 이벤트 메시지를 게시할 수 있습니다. 시스템은 게시 서비스를 변경하지 않고도 새 마이크로서비스를 추가할 수 있습니다.

Q

쿼리 계획

SQL 관계형 데이터베이스 시스템의 데이터에 액세스하는 데 사용되는 일련의 단계 (예: 지침).

쿼리 계획 회귀

데이터베이스 서비스 최적화 프로그램이 데이터베이스 환경을 변경하기 전보다 덜 최적의 계획을 선택하는 경우입니다. 통계, 제한 사항, 환경 설정, 쿼리 파라미터 바인딩 및 데이터베이스 엔진 업데이트의 변경으로 인해 발생할 수 있습니다.

R

RACI 매트릭스

RACI ([책임, 책임, 상담, 정보 제공](#)) 를 참조하십시오.

랜섬웨어

결제 완료될 때까지 컴퓨터 시스템이나 데이터에 대한 액세스를 차단하도록 설계된 악성 소프트웨어입니다.

RASCI 매트릭스

[책임, 책임, 상담, 정보 제공 \(RACI\)](#) 을 참조하십시오.

RCAC

[행 및 열 액세스 제어를](#) 참조하십시오.

읽기 전용 복제본

읽기 전용 용도로 사용되는 데이터베이스의 사본입니다. 쿼리를 읽기 전용 복제본으로 라우팅하여 기본 데이터베이스의 로드를 줄일 수 있습니다.

재설계

[7 R](#)을 참조하십시오.

Recovery Point Objective(RPO)

마지막 데이터 복구 시점 이후 허용되는 최대 시간입니다. 이에 따라 마지막 복구 시점과 서비스 중단 사이에 허용되는 데이터 손실로 간주되는 범위가 결정됩니다.

Recovery Time Objective(RTO)

서비스 중단과 서비스 복원 사이의 허용 가능한 지연 시간입니다.

리팩터링

[7 R](#)을 참조하십시오.

리전

지리적 AWS 영역별 리소스 모음. AWS 리전 각각은 격리되어 있고 서로 독립적이므로 내결함성, 안정성 및 복원력을 제공합니다. 자세한 내용은 [사용할 수 있는 AWS 리전 계정 지정을](#) 참조하십시오.

회귀

숫자 값을 예측하는 ML 기법입니다. 예를 들어, '이 집은 얼마에 팔릴까?'라는 문제를 풀기 위해 ML 모델은 선형 회귀 모델을 사용하여 주택에 대해 알려진 사실(예: 면적)을 기반으로 주택의 매매 가격을 예측할 수 있습니다.

리호스팅

[7 R을](#) 참조하십시오.

release

배포 프로세스에서 변경 사항을 프로덕션 환경으로 승격시키는 행위입니다.

고쳐 놓다

[7 R을](#) 참조하십시오.

리플랫폼

[7 R을](#) 참조하십시오.

환매

[7 R을](#) 참조하십시오.

복원력

장애를 견디거나 장애를 복구할 수 있는 애플리케이션의 능력 [고가용성](#) 및 [재해 복구](#)는 복원력을 계획할 때 일반적으로 고려해야 할 사항입니다. AWS 클라우드 자세한 내용은 [AWS 클라우드 복원력을](#) 참조하십시오.

리소스 기반 정책

Amazon S3 버킷, 엔드포인트, 암호화 키 등의 리소스에 연결된 정책입니다. 이 유형의 정책은 액세스가 허용된 보안 주체, 지원되는 작업 및 충족해야 하는 기타 조건을 지정합니다.

RACI(Responsible, Accountable, Consulted, Informed) 매트릭스

마이그레이션 활동 및 클라우드 운영에 참여하는 모든 당사자의 역할과 책임을 정의하는 매트릭스입니다. 매트릭스 이름은 매트릭스에 정의된 책임 유형에서 파생됩니다. 실무 담당자 (R), 의사 결

정권자 (A), 업무 수행 조연자 (C), 결과 통보 대상자 (I). 지원자는 (S) 선택사항입니다. 지원자를 포함하면 매트릭스를 RASCI 매트릭스라고 하고, 지원자를 제외하면 RACI 매트릭스라고 합니다.

대응 제어

보안 기준에서 벗어나거나 부정적인 이벤트를 해결하도록 설계된 보안 제어입니다. 자세한 내용은 [Implementing security controls on AWS의 Responsive controls](#)를 참조하십시오.

retain

[7 R](#)을 참조하십시오.

은퇴

[7 R](#)을 참조하십시오.

회전

공격자가 자격 증명에 액세스하는 것을 더 어렵게 만들기 위해 [암호](#)를 주기적으로 업데이트하는 프로세스입니다.

행 및 열 액세스 제어(RCAC)

액세스 규칙이 정의된 기본적인 유연한 SQL 표현식을 사용합니다. RCAC는 행 권한과 열 마스크로 구성됩니다.

RPO

[복구 지점 목표를](#) 참조하십시오.

RTO

[복구 시간 목표를](#) 참조하십시오.

런복

특정 작업을 수행하는 데 필요한 일련의 수동 또는 자동 절차입니다. 일반적으로 오류율이 높은 반복 작업이나 절차를 간소화하기 위해 런복을 만듭니다.

S

SAML 2.0

많은 ID 제공업체 (IdPs) 가 사용하는 개방형 표준입니다. 이 기능을 사용하면 페더레이션 싱글 사인온 (SSO) 이 가능하므로 조직의 모든 사용자를 위해 IAM에서 사용자를 생성하지 않고도 사용자가 AWS API 작업에 AWS Management Console 로그인하거나 API 작업을 호출할 수 있습니다.

SAML 2.0 기반 페더레이션에 대한 자세한 내용은 IAM 설명서의 [SAML 2.0 기반 페더레이션 정보](#)를 참조하십시오.

SCADA

[감독 제어 및 데이터 수집](#)을 참조하십시오.

SCP

[서비스 제어 정책](#)을 참조하십시오.

secret

에는 AWS Secrets Manager 암호화된 형태로 저장하는 비밀번호나 사용자 자격 증명과 같은 기밀 또는 제한된 정보. 비밀 값과 해당 메타데이터로 구성됩니다. 비밀 값은 바이너리, 단일 문자열 또는 여러 문자열일 수 있습니다. 자세한 내용은 [Secrets Manager 시크릿에는 무엇이 들어 있나요?](#)를 참조하십시오. Secrets Manager 설명서에서 확인할 수 있습니다.

보안 제어

위험 행위자가 보안 취약성을 악용하는 능력을 방지, 탐지 또는 감소시키는 기술적 또는 관리적 가드레일입니다. [보안 제어에는 예방적, 탐정적, 대응적, 사전 예방적 제어의 네 가지 기본 유형이 있습니다.](#)

보안 강화

공격 표면을 줄여 공격에 대한 저항력을 높이는 프로세스입니다. 더 이상 필요하지 않은 리소스 제거, 최소 권한 부여의 보안 모범 사례 구현, 구성 파일의 불필요한 기능 비활성화 등의 작업이 여기에 포함될 수 있습니다.

보안 정보 및 이벤트 관리(SIEM) 시스템

보안 정보 관리(SIM)와 보안 이벤트 관리(SEM) 시스템을 결합하는 도구 및 서비스입니다. SIEM 시스템은 서버, 네트워크, 디바이스 및 기타 소스에서 데이터를 수집, 모니터링 및 분석하여 위협과 보안 침해를 탐지하고 알림을 생성합니다.

보안 대응 자동화

보안 이벤트에 자동으로 대응하거나 보안 이벤트를 해결하도록 설계된 사전 정의되고 프로그래밍된 조치입니다. 이러한 자동화는 보안 모범 사례를 구현하는 데 도움이 되는 [탐지](#) 또는 [대응형](#) 보안 제어 역할을 합니다. AWS 자동 응답 조치의 예로는 VPC 보안 그룹 수정, Amazon EC2 인스턴스 패치, 자격 증명 교체 등이 있습니다.

서버 측 암호화

수신자에 의한 목적지의 데이터 암호화 AWS 서비스

서비스 제어 정책(SCP)

AWS Organizations에 속한 조직의 모든 계정에 대한 권한을 중앙 집중식으로 제어하는 정책입니다. SCP는 관리자가 사용자 또는 역할에 위임할 수 있는 작업에 대해 제한을 설정하거나 가드레일을 정의합니다. SCP를 허용 목록 또는 거부 목록으로 사용하여 허용하거나 금지할 서비스 또는 작업을 지정할 수 있습니다. 자세한 내용은 AWS Organizations 설명서의 [서비스 제어 정책을](#) 참조하십시오.

서비스 엔드포인트

의 진입점 URL입니다 AWS 서비스. 엔드포인트를 사용하여 대상 서비스에 프로그래밍 방식으로 연결할 수 있습니다. 자세한 내용은 AWS 일반 참조의 [AWS 서비스 엔드포인트](#)를 참조하십시오.

서비스 수준에 관한 계약(SLA)

IT 팀이 고객에게 제공하기로 약속한 내용(예: 서비스 가동 시간 및 성능)을 명시한 계약입니다.

서비스 수준 표시기 (SLI)

오류율, 가용성 또는 처리량과 같은 서비스의 성능 측면을 측정하는 것입니다.

서비스 수준 목표 (SLO)

[서비스 수준 지표로 측정되는 서비스 상태를 나타내는 대상 지표입니다.](#)

공동 책임 모델

클라우드 보안 및 규정 준수에 AWS 대한 책임을 공유하는 것을 설명하는 모델입니다. AWS 클라우드의 보안을 책임지는 반면, 사용자는 클라우드에서의 보안을 담당합니다. 자세한 내용은 [공동 책임 모델](#)을 참조하십시오.

시앰

[보안 정보 및 이벤트 관리 시스템을](#) 참조하십시오.

단일 장애 지점 (SPOF)

응용 프로그램의 중요한 단일 구성 요소에서 발생한 오류로 인해 시스템이 중단될 수 있습니다.

SLA

SLA ([서비스 수준 계약](#)) 를 참조하십시오.

SLI

[서비스 수준 표시기](#) 참조.

SLO

[서비스 수준 목표를](#) 참조하십시오.

split-and-seed 모델

현대화 프로젝트를 확장하고 가속화하기 위한 패턴입니다. 새로운 기능과 제품 릴리스가 정의되면 핵심 팀이 분할되어 새로운 제품 팀이 만들어집니다. 이를 통해 조직의 역량과 서비스 규모를 조정하고, 개발자 생산성을 개선하고, 신속한 혁신을 지원할 수 있습니다. 자세한 내용은 [의 애플리케이션 현대화를 위한 단계별 접근 방식을 참조하십시오. AWS 클라우드](#)

SPOF

[단일 장애 지점 보기.](#)

스타 스키마

하나의 큰 팩트 테이블을 사용하여 트랜잭션 또는 측정 데이터를 저장하고 하나 이상의 작은 차원 테이블을 사용하여 데이터 속성을 저장하는 데이터베이스 구성 구조입니다. 이 구조는 [데이터 웨어하우스에서](#) 사용하거나 비즈니스 인텔리전스 용도로 설계되었습니다.

Strangler Fig 패턴

레거시 시스템을 폐기할 수 있을 때까지 시스템 기능을 점진적으로 다시 작성하고 교체하여 모놀리식 시스템을 현대화하기 위한 접근 방식. 이 패턴은 무화과 덩굴이 나무로 자라 결국 숙주를 압도하고 대체하는 것과 비슷합니다. [Martin Fowler](#)가 모놀리식 시스템을 다시 작성할 때 위험을 관리하는 방법으로 이 패턴을 도입했습니다. 이 패턴을 적용하는 방법의 예는 [컨테이너 및 Amazon API Gateway를 사용하여 기존의 Microsoft ASP.NET\(ASMX\) 웹 서비스를 점진적으로 현대화하는 방법을 참조하십시오.](#)

서브넷

VPC의 IP 주소 범위입니다. 서브넷은 단일 가용 영역에 상주해야 합니다.

감독 통제 및 데이터 수집 (SCADA)

제조 시 하드웨어와 소프트웨어를 사용하여 물리적 자산과 생산 작업을 모니터링하는 시스템입니다.

대칭 암호화

동일한 키를 사용하여 데이터를 암호화하고 복호화하는 암호화 알고리즘입니다.

합성 테스트

잠재적 문제를 감지하거나 성능을 모니터링하기 위해 사용자 상호 작용을 시뮬레이션하는 방식으로 시스템을 테스트합니다. [Amazon CloudWatch Synthetics](#)를 사용하여 이러한 테스트를 생성할 수 있습니다.

T

tags

리소스 구성을 위한 메타데이터 역할을 하는 키-값 쌍. AWS 태그를 사용하면 리소스를 손쉽게 관리, 식별, 정리, 검색 및 필터링할 수 있습니다. 자세한 내용은 [AWS 리소스에 태그 지정](#)을 참조하십시오.

대상 변수

지도 ML에서 예측하려는 값으로, 결과 변수라고도 합니다. 예를 들어, 제조 설정에서 대상 변수는 제품 결함일 수 있습니다.

작업 목록

런북을 통해 진행 상황을 추적하는 데 사용되는 도구입니다. 작업 목록에는 런북의 개요와 완료해야 할 일반 작업 목록이 포함되어 있습니다. 각 일반 작업에 대한 예상 소요 시간, 소유자 및 진행 상황이 작업 목록에 포함됩니다.

테스트 환경

[환경을 참조하십시오.](#)

훈련

ML 모델이 학습할 수 있는 데이터를 제공하는 것입니다. 훈련 데이터에는 정답이 포함되어야 합니다. 학습 알고리즘은 훈련 데이터에서 대상(예측하려는 답)에 입력 데이터 속성을 매핑하는 패턴을 찾고, 이러한 패턴을 캡처하는 ML 모델을 출력합니다. 그런 다음 ML 모델을 사용하여 대상을 모르는 새 데이터에 대한 예측을 할 수 있습니다.

전송 게이트웨이

VPC와 온프레미스 네트워크를 상호 연결하는 데 사용할 수 있는 네트워크 전송 허브입니다. 자세한 내용은 AWS Transit Gateway 설명서의 [트랜짓 게이트웨이란 무엇입니까?](#)를 참조하십시오.

트렁크 기반 워크플로

개발자가 기능 브랜치에서 로컬로 기능을 구축하고 테스트한 다음 해당 변경 사항을 기본 브랜치에 병합하는 접근 방식입니다. 이후 기본 브랜치는 개발, 프로덕션 이전 및 프로덕션 환경에 순차적으로 구축됩니다.

신뢰할 수 있는 액세스

조직 내 AWS Organizations 및 해당 계정에서 사용자를 대신하여 작업을 수행하도록 지정한 서비스에 권한 부여 신뢰할 수 있는 서비스는 필요할 때 각 계정에 서비스 연결 역할을 생성하여 관

리 작업을 수행합니다. 자세한 내용은 AWS Organizations 설명서의 [다른 AWS 서비스와 AWS Organizations 함께 사용](#)을 참조하십시오.

튜닝

ML 모델의 정확도를 높이기 위해 훈련 프로세스의 측면을 여러 변경하는 것입니다. 예를 들어, 레이블링 세트를 생성하고 레이블을 추가한 다음 다양한 설정에서 이러한 단계를 여러 번 반복하여 모델을 최적화하는 방식으로 ML 모델을 훈련할 수 있습니다.

피자 두 판 팀

피자 두 판만 들고 배블리 먹을 수 있는 소규모 DevOps 팀. 피자 두 판 팀 규모는 소프트웨어 개발에 있어 가능한 최상의 공동 작업 기회를 보장합니다.

U

불확실성

예측 ML 모델의 신뢰성을 저해할 수 있는 부정확하거나 불완전하거나 알려지지 않은 정보를 나타내는 개념입니다. 불확실성에는 두 가지 유형이 있습니다. 인식론적 불확실성은 제한적이고 불완전한 데이터에 의해 발생하는 반면, 우연한 불확실성은 데이터에 내재된 노이즈와 무작위성에 의해 발생합니다. 자세한 내용은 [Quantifying uncertainty in deep learning systems](#) 가이드를 참조하십시오.

차별화되지 않은 작업

애플리케이션을 만들고 운영하는 데 필요하지만 최종 사용자에게 직접적인 가치를 제공하거나 경쟁 우위를 제공하지 못하는 작업을 헤비 리프팅이라고도 합니다. 차별화되지 않은 작업의 예로는 조달, 유지보수, 용량 계획 등이 있습니다.

상위 환경

[환경을](#) 보세요.

V

정리

스토리지를 회수하고 성능을 향상시키기 위해 증분 업데이트 후 정리 작업을 수반하는 데이터베이스 유지 관리 작업입니다.

버전 제어

리포지토리의 소스 코드 변경과 같은 변경 사항을 추적하는 프로세스 및 도구입니다.

VPC 피어링

프라이빗 IP 주소를 사용하여 트래픽을 라우팅할 수 있게 하는 두 VPC 간의 연결입니다. 자세한 내용은 Amazon VPC 설명서의 [VPC 피어링이란?](#)을 참조하십시오.

취약성

시스템 보안을 손상시키는 소프트웨어 또는 하드웨어 결함입니다.

W

웹 캐시

자주 액세스하는 최신 관련 데이터를 포함하는 버퍼 캐시입니다. 버퍼 캐시에서 데이터베이스 인스턴스를 읽을 수 있기 때문에 주 메모리나 디스크에서 읽는 것보다 빠릅니다.

웹 데이터

자주 액세스하지 않는 데이터입니다. 이런 종류의 데이터를 쿼리할 때는 일반적으로 적절히 느린 쿼리가 허용됩니다.

윈도우 함수

현재 레코드와 어떤 식으로든 관련된 행 그룹에 대해 계산을 수행하는 SQL 함수입니다. 윈도우 함수는 이동 평균을 계산하거나 현재 행의 상대적 위치를 기반으로 행 값에 액세스하는 등의 작업을 처리하는 데 유용합니다.

워크로드

고객 대면 애플리케이션이나 백엔드 프로세스 같이 비즈니스 가치를 창출하는 리소스 및 코드 모음입니다.

워크스트림

마이그레이션 프로젝트에서 특정 작업 세트를 담당하는 직무 그룹입니다. 각 워크스트림은 독립적이지만 프로젝트의 다른 워크스트림을 지원합니다. 예를 들어, 포트폴리오 워크스트림은 애플리케이션 우선순위 지정, 웨이브 계획, 마이그레이션 메타데이터 수집을 담당합니다. 포트폴리오 워크스트림은 이러한 자산을 마이그레이션 워크스트림에 전달하고, 마이그레이션 워크스트림은 서버와 애플리케이션을 마이그레이션합니다.

원

한 번 쓰고, 많이 읽으세요.

WQF

AWS 워크로드 검증 프레임워크를 참조하십시오.

한 번 작성하고 여러 번 읽기 (WORM)

데이터를 한 번 쓰고 데이터가 삭제되거나 수정되지 않도록 하는 스토리지 모델입니다. 인증된 사용자는 필요한 만큼 데이터를 여러 번 읽을 수 있지만 변경할 수는 없습니다. 이 데이터 스토리지 인프라는 변경할 수 없는 것으로 간주됩니다.

Z

제로데이 익스플로잇

제로데이 취약점을 악용하는 공격 (일반적으로 멀웨어)입니다.

제로데이 취약성

프로덕션 시스템의 명백한 결함 또는 취약성입니다. 위협 행위자는 이러한 유형의 취약성을 사용하여 시스템을 공격할 수 있습니다. 개발자는 공격의 결과로 취약성을 인지하는 경우가 많습니다.

좀비 애플리케이션

평균 CPU 및 메모리 사용량이 5% 미만인 애플리케이션입니다. 마이그레이션 프로젝트에서는 이러한 애플리케이션을 사용 중지하는 것이 일반적입니다.

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