



AWS 대규모 마이그레이션을 위한 전략 및 모범 사례

AWS 규범적 지침



AWS 규범적 지침: AWS 대규모 마이그레이션을 위한 전략 및 모범 사례

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
대규모 마이그레이션을 위한 지침	1
범위, 전략, 일정	3
범위 — 무엇을 마이그레이션하고 있습니까?	3
전략 — 마이그레이션을 해야 하는 이유는	3
타임라인 — 언제 마이그레이션을 완료해야 합니까?	4
모범 사례	6
사람	6
경영진 지원	6
팀 협업 및 소유권	7
학습	8
기술	9
자동화, 추적 및 툴링 통합	10
사전 요구 사항 및 마이그레이션 후 검증	12
프로세스	13
대규모 마이그레이션 준비	14
대규모 마이그레이션 실행	18
추가 고려 사항	21
결론	24
리소스	25
AWS대규모 마이그레이션	25
관련AWS 규범 지침 리소스	25
추가 참고	25
동영상	25
기여자	26
문서 기록	27
용어집	28
#	28
A	29
B	31
C	33
D	36
E	40
F	42

G	43
H	44
I	46
L	48
M	49
O	53
P	55
Q	58
R	58
S	61
T	64
U	66
V	66
W	67
Z	68
.....	ix

AWS 대규모 마이그레이션을 위한 전략 및 모범 사례

Amazon Web Services ([기고자](#))

2022년 5월([문서 기록](#))

많은 AWS 고객이 비즈니스에 미치는 영향을 AWS 클라우드 최소화하면서 최대한 빨리 많은 수의 서버와 애플리케이션을 마이그레이션하기를 원합니다. 데이터 센터 임대료가 갱신 또는 종료되기 직전이거나 조직이 기술 혁신의 첫 단계를 밟고 있기 때문에 조직에서 대규모 마이그레이션 프로젝트를 시작할 수 있습니다. 그러나 대규모는 범위 내 서버 수로만 결정되는 것은 아닙니다. 또한 사람, 프로세스, 기술, 우선 순위와 같은 복잡성을 고려하여 마이그레이션으로 인한 조직 변화 수준도 고려합니다.

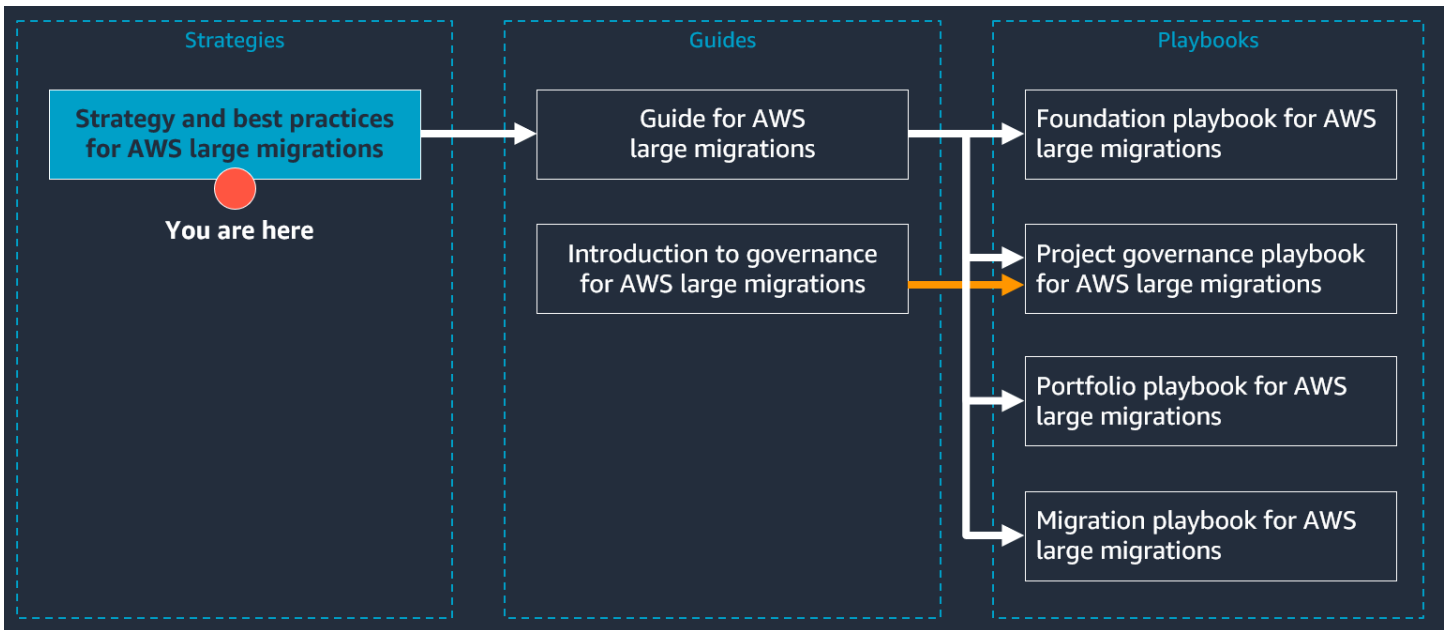
이 가이드는 대규모로 이동할 수 있는 능력에 초점을 맞춥니다. AWS 거의 또는 전혀 변경하지 않고 기존 애플리케이션을 마이그레이션할 수 있습니다. 클라우드를 시작 지점으로 사용하여 이러한 애플리케이션을 클라우드 네이티브 또는 서버리스 기술로 전환하고 애플리케이션을 현대화하여 추가적인 비즈니스 이점을 얻을 수 있습니다.

이 가이드에서는 대규모 마이그레이션의 모범 사례를 설명하고 금융 서비스, 의료 등 다양한 부문에 걸친 고객의 사용 사례를 제공합니다. 또한 고객이 마이그레이션하면서 얻은 교훈의 실제 사례를 제공합니다. AWS이 가이드의 목적은 대규모 마이그레이션의 초기 단계에 있는 고객을 지원하는 것입니다. 하지만 이 가이드의 모범 사례와 전략은 마이그레이션 여정의 모든 단계에서 유용할 수 있습니다. 이미 100레벨의 지식을 갖추고 AWS 서비스 있고 마이그레이션을 [위한 AWS 권장 프로세스](#)를 알고 있다고 가정합니다.

대규모 마이그레이션을 위한 지침

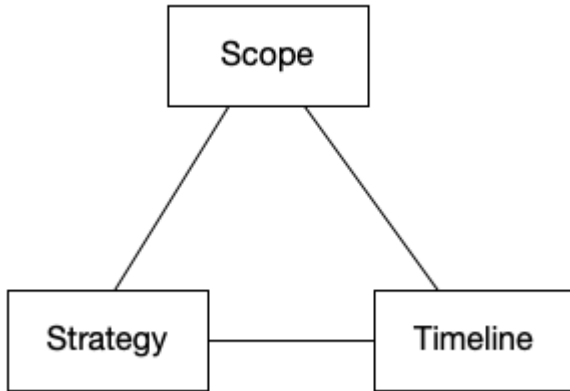
300대 이상의 서버를 마이그레이션하는 것은 대규모 마이그레이션으로 간주됩니다. 대규모 마이그레이션 프로젝트에서 발생하는 인력, 프로세스 및 기술 문제는 일반적으로 대부분의 기업에서 생소합니다. 이 문서는 국가로의 대규모 마이그레이션에 대한 AWS 규범적 지침 시리즈의 일부입니다. AWS 클라우드이 시리즈는 처음부터 올바른 전략과 모범 사례를 적용하여 클라우드로의 여정을 간소화할 수 있도록 설계되었습니다.

다음 그림은 이 시리즈의 다른 문서를 보여줍니다. 먼저 전략을 검토한 다음 가이드를 검토한 다음 플레이북으로 진행하십시오. 전체 시리즈에 액세스하려면 [대규모 마이그레이션을](#) 참조하십시오. AWS 클라우드



범위, 전략 및 일정

모든 프로그램의 구성 요소와 대규모 마이그레이션에서의 관련성을 구성하는 세 가지 핵심 요소는 범위, 전략 및 일정입니다.



마이그레이션 여정의 발판을 마련하려면 마이그레이션 프로그램을 시작할 때부터 이러한 요소를 조율하고 이해해야 합니다. 이러한 요소 중 하나를 변경하면 다른 요소에도 영향을 미칩니다. 변경이 아무리 기본적인거나 합리적으로 보일지라도 모든 변경 사항에는 재조정을 고려해야 합니다.

범위 — 무엇을 마이그레이션하고 있습니까?

마이그레이션이 반쯤 진행된 경우에도 프로그램의 전체 범위가 정의되지 않는 것이 일반적입니다. 이는 이후 단계까지 다양한 요인이 풀리지 않을 수 있기 때문입니다. 예를 들어, 마이그레이션 도중에 CMDB (구성 관리 데이터베이스)에 기록되지 않은 새도우 IT를 발견할 수 있습니다. 또는 해당 애플리케이션을 실행하는 데 필요한 지원 네트워크 및 보안 서비스 (예: AWS 파트너에 대한 VPN 연결 및 인증서 서명을 위한 인증 기관)를 고려하지 않고 서버 뷰에 초점을 맞췄을 수도 있습니다. 범위를 정의하는 데 시간을 투자하여 목표 비즈니스 성과부터 거꾸로 작업하는 것이 좋습니다. 결국 검색 도구를 사용하여 자산을 발견하게 될 수도 있습니다. 이 모범 사례는 이 가이드의 뒷부분에서 설명하겠습니다.

대규모 마이그레이션에는 알 수 없는 사항이 있기 때문에 범위가 변경될 것입니다. 이러한 미지수들은 관련성을 거의 또는 전혀 이해하지 못한 채 환경 고고학의 일부가 된 시스템이나, 여러분이 만든 계획에 지연과 변화를 야기하는 생산 사고의 형태일 수 있습니다. 핵심은 유연하게 대처하고 프로그램을 계속 진행하기 위한 비상 계획을 마련하는 것입니다.

전략 — 마이그레이션을 해야 하는 이유는

다음 중 하나 이상의 이유로 로 마이그레이션하기로 AWS 계획했을 수 있습니다.

- 애플리케이션 팀은 새로운 CI/CD 파이프라인을 구현하거나, 최신 애플리케이션 스택을 배포하거나, 지원이 중단된 레거시 플랫폼을 현대화하기를 원합니다.
- 인프라 팀은 임대 기간이 만료되고 공급자가 전원을 끄기 전에 노후화된 데이터 센터를 빨리 벗어나야 합니다.
- 이사회는 비즈니스 future 빠른 변화를 가져올 수 있도록 전략적 방향으로 클라우드로 전환해야 한다고 결정했습니다.

이유가 무엇이든, 비즈니스 및 IT 조직은 이러한 모든 이유와 그 이상을 염두에 둘 것입니다. 드라이버가 무엇인지 이해하고, 이를 전달하고, 우선순위를 정하는 것이 중요합니다. 각각의 추가 요인은 이미 규모가 큰 마이그레이션에 시간, 비용, 범위 및 위험을 추가할 수 있습니다. 전략이 일정과 범위에 미치는 영향을 완전히 파악하는 것이 중요합니다.

마이그레이션 전략을 정의한 후 성공의 주요 열쇠 중 하나는 다양한 이해 관계자와 팀 간의 요구 사항을 조정하는 것입니다. 마이그레이션을 수행하려면 인프라, 보안, 애플리케이션, 운영 등 조직 전반의 여러 팀이 필요합니다. 이러한 팀에는 개별 우선 순위와 이미 시작되었을 수 있는 기타 프로젝트가 있습니다. 이러한 팀들이 서로 다른 일정과 우선순위로 작업하고 있다면 마이그레이션 계획을 합의하고 구현하기가 더 어려워집니다. 마이그레이션 팀과 주요 이해 관계자는 관련된 모든 팀이 단일 목표를 향해 작업하고 단일 마이그레이션 일정에 따라 우선 순위를 조정하도록 해야 합니다.

다양한 팀에서 원하는 비즈니스 성과를 어떻게 조정할 수 있는지 살펴보는 것이 좋습니다. 예를 들어, 미사용 스토리지로 AWS 마이그레이션하고 AWS Key Management Service (AWS KMS) 를 사용하여 저장 데이터를 암호화하면 마이그레이션 및 보안 목표를 모두 충족할 수 있습니다.

기업들은 애플리케이션을 현대화하여 인프라 업그레이드로 이어질 수 있는 반면, 인프라 팀은 절약하고 인프라 변경을 최소화하고자 하는 경우가 많습니다. 대규모 마이그레이션에 대한 사고방식은 가능한 한 기본적인어야 합니다. 관련된 팀은 모든 것을 한 번에 하려고 하지 않아야 합니다.

이를 달성하려면 프로젝트 초기에 적절한 기대치를 설정해야 합니다. 핵심 메시지는 “먼저 마이그레이션한 다음 현대화”여야 합니다. 이 접근 방식을 통해 조직은 기술적 부채를 줄이고 궁극적으로 규모에 맞게 운영할 수 있을 뿐만 아니라 에서 제공할 AWS 클라우드 수 있는 확장성과 민첩성을 활용하여 다양한 현대화 접근 방식을 도입할 수 있습니다. 장기적으로 생각하면 인프라 팀이 인프라 배포 및 관리를 간소화하는 데 도움이 됩니다. 따라서 기업은 기능 출시 주기를 단축할 수 있습니다.

타임라인 — 언제 마이그레이션을 완료해야 합니까?

비즈니스 사례에 따라 할당된 시간 내에 달성할 수 있는 것보다 더 많은 작업을 수행하지 않도록 해야 합니다. 마이그레이션을 위한 드라이버가 정해진 완료 날짜를 기준으로 하는 경우 해당 일정 요구 사항을

을 충족하는 전략을 선택해야 합니다. 대부분의 대규모 마이그레이션은 이러한 시간 기반 제약을 기반으로 하므로 마이그레이션 전략에는 확장이나 오버런의 여지가 거의 없는 정의되고 고정된 일정과 결과가 있어야 합니다.

시간에 민감한 이러한 유형의 마이그레이션에서는 “먼저 마이그레이션한 다음 현대화” 접근 방식을 사용하는 것이 좋습니다. 이를 통해 기대치를 설정하고 팀이 개별 프로젝트 계획과 예산이 전체 마이그레이션 목표와 일치하는지 확인할 수 있습니다. 프로젝트에서 가능한 한 빨리 의견 불일치를 찾아내고, 빠르게 실패하고, 운영 위원회 수준에서 의견 불일치를 해결하고, 적절한 이해관계자를 참여시켜 조정이 이루어지도록 하는 것이 중요합니다.

반대로 마이그레이션의 주요 목표가 애플리케이션 현대화의 이점을 얻는 것이라면 프로그램 초기에 이를 반드시 언급해야 합니다. 많은 프로그램은 고정된 마감일을 기반으로 초기 목표를 세우고 시작하며, 미해결 문제 및 문제를 해결하고자 하는 이해 관계자의 요구 사항에 맞춰 계획을 세우지 않습니다. 소스 시스템에서 이러한 문제가 수년간 존재해왔지만 이제는 마이그레이션을 방해하는 인위적인 장애물이 되는 경우도 있습니다.

마이그레이션 중 현대화 활동은 비즈니스 애플리케이션의 기능에 영향을 미칠 수 있습니다. 운영 체제 버전 변경과 같은 작은 업그레이드로 인식되는 경우에도 프로그램 일정에 큰 영향을 미칠 수 있습니다. 이것들은 사소한 것으로 간주되어서는 안 됩니다.

대규모 마이그레이션 모범 사례

대규모 마이그레이션은 조직의 기능을 좌우하는 요인에 따라 어려울 수 있습니다. 이 섹션에서는 초기 단계에서 해결하고 프로젝트 전반에 걸쳐 추적할 경우 대규모 마이그레이션을 단순화할 수 있는 몇 가지 주요 요소를 다룹니다.

대규모 마이그레이션에 대한 다음 모범 사례는 다른 고객으로부터 캡처한 데이터를 기반으로 합니다. 베스트 프랙티스는 세 가지 범주로 구분됩니다.

- 사람
- 기술
- 프로세스

인력 관점

이 섹션에서는 사람 관점의 다음과 같은 주요 영역에 초점을 맞춥니다.

- 경영진 지원 — 의사 결정 권한이 있는 단일 스레드 리더 발굴
- 팀 협업 및 소유권 — 다양한 팀 간의 협업
- 교육 — 다양한 툴링에 대해 팀을 사전에 교육합니다.

경영진 지원

이 섹션:

- [단일 스레드를 사용하는 리더 찾기](#)
- [고위 경영진을 조정하세요.](#)

단일 스레드를 사용하는 리더 찾기

대규모 마이그레이션을 시작할 때는 프로젝트에 100% 전념하고 책임을 지는 단일 스레드 기술 리더를 찾는 것이 중요합니다. 이 리더는 의사 결정을 내리고, 사일로를 없애고, 일관된 우선 순위를 유지함으로써 업무 흐름을 간소화할 수 있는 권한을 부여받습니다.

대규모 마이그레이션 글로벌 고객은 프로그램을 시작할 때 매주 한 대의 서버를 사용하던 것을 두 번째 달 초에 매주 80대 이상으로 확장할 수 있었습니다. 마이그레이션할 서버를 빠르게 확장하려면 단일 스

레드 리더로서의 CIO의 전폭적인 지원이 매우 중요했습니다. CIO는 마이그레이션 팀과 매주 마이그레이션 컷오버 회의에 참석하여 문제를 실시간으로 에스컬레이션하고 해결함으로써 마이그레이션 속도를 가속화했습니다.

고위 경영진을 조정하세요.

마이그레이션의 성공 기준과 관련하여 여러 팀 간에 의견을 조율하는 것이 중요합니다. 소규모 전담 팀으로도 마이그레이션 계획 및 구현을 수행할 수 있지만 전략을 정의하고 주변 활동을 수행할 때는 문제가 발생합니다. 이러한 잠재적 장애물로 인해 다음을 포함하여 IT 조직의 다양한 영역에서 조치를 취하거나 에스컬레이션해야 할 수 있습니다.

- 업무
- 애플리케이션
- 네트워킹
- 보안
- 인프라
- 타사 공급업체

애플리케이션 소유자의 직접적인 조치, 경영진, 조정, 단일 스레드 리더로의 명확한 에스컬레이션이 중요해집니다.

팀 협업 및 소유권

이 섹션:

- [부서 간 클라우드 지원 팀 만들기](#)
- [핵심 마이그레이션 팀 외부의 팀 및 개인에 대한 요구 사항을 미리 정의하십시오.](#)
- [워크로드를 마이그레이션할 때 라이선스 문제가 없는지 확인하십시오.](#)

부서 간 클라우드 지원 팀 만들기

대규모 마이그레이션 프로젝트의 중요한 첫 단계는 조직이 클라우드에서 작업할 수 있도록 지원하는 것입니다. 이를 위해서는 [클라우드 지원 엔진](#) (CEE) 을 구축하는 것이 좋습니다. CEE는 마이그레이션을 위한 조직의 운영 준비 상태를 중점적으로 검토하는 권한과 책임감을 갖춘 팀입니다. AWS CEE는 인프라, 애플리케이션, 운영 및 보안 담당자를 포함하는 다기능 팀이어야 합니다. 팀의 책임은 다음과 같습니다.

- 정책 개발
- 조직의 클라우드 운영 모델을 수립하는 데 필요한 도구, 프로세스 및 아키텍처의 정의 및 구현
- 이해 관계자가 대표하는 모든 영역에서 이해 관계자의 조정을 지속적으로 촉진합니다.

한 의료 고객은 CEE로 시작하지 않았습니다. 하지만 초기 파일럿 마이그레이션을 통해 격차가 확인되었습니다. 엄격한 기한을 두고 최종 마이그레이션 마감일을 앞두고 팀은 마이그레이션 전쟁실을 마련했습니다. 마이그레이션 전쟁터에서는 인프라, 보안, 애플리케이션 및 비즈니스 분야의 이해 관계자가 문제 해결을 지원할 수 있습니다.

핵심 마이그레이션 팀 외부의 팀 및 개인에 대한 요구 사항을 미리 정의하십시오.

핵심 프로그램 외부의 팀과 개인을 식별하고 마이그레이션 계획 단계에서 이들이 관여하는지 정의하세요. 이후 단계에서 마이그레이션의 추진력을 높이려면 애플리케이션 팀의 참여에 특히 주의를 기울이세요. 애플리케이션에 대한 지식, 문제 진단 능력, 컷오버 승인 요구 사항 등이 필요합니다.

마이그레이션은 핵심 팀이 주도하지만 애플리케이션 팀은 전환 중에 마이그레이션 계획을 검증하고 테스트하는 데 관여할 가능성이 높습니다. 고객은 클라우드 마이그레이션을 애플리케이션 마이그레이션이 아닌 인프라 프로젝트로 접근하는 경우가 많습니다. 이로 인해 마이그레이션 중에 문제가 발생할 수 있습니다.

마이그레이션 전략을 선택할 때는 애플리케이션 팀의 필수 참여를 고려하는 것이 좋습니다. 예를 들어, 재호스팅 전략은 애플리케이션 환경이 더 많이 변경되는 리플랫폼 또는 리팩터링 전략에 비해 애플리케이션 팀의 개입이 덜 필요합니다. 애플리케이션 소유자 가용성이 제한적이라면 리팩터링, 재배포 또는 재구매 전략 대신 재호스팅 또는 리플랫폼 사용을 고려해 보세요.

워크로드를 마이그레이션할 때 라이선스 문제가 없는지 확인하십시오.

기업 기성품을 클라우드로 마이그레이션하면 라이선스가 변경될 수 있습니다. 라이선스 계약은 온프레미스 자산에 초점을 맞출 수 있습니다. 예를 들어 라이선스는 CPU를 기반으로 하거나 특정 MAC 주소에 연결될 수 있습니다. 또는 라이선스 계약에 퍼블릭 클라우드 환경에서 호스팅할 수 있는 권한이 포함되지 않을 수도 있습니다. 그러나 공급업체와 라이선스를 재협상하는 경우 리드 타임이 길어지고 마이그레이션이 어려워질 수 있습니다.

마이그레이션 범위를 정의하는 즉시 소싱 또는 공급업체 관리 팀과 협업하는 것이 좋습니다. 라이선스는 대상 아키텍처 및 마이그레이션 패턴에도 영향을 미칠 수 있습니다.

학습

이 섹션:

- [새로운 도구와 프로세스에 대해 팀을 교육하세요.](#)

새로운 도구와 프로세스에 대해 팀을 교육하세요.

마이그레이션 전략을 정의한 후에는 마이그레이션과 대상 운영 모델에 필요한 교육이 무엇인지 파악하는 데 시간을 투자하세요. 마이그레이션 중에는 조직에 새로 추가된 것과 AWS Database Migration Service 같은 도구를 사용하게 될 것입니다. 팀을 사전에 교육하면 마이그레이션 단계에서 발생하는 지연을 줄일 수 있습니다.

도구를 직접 사용해 볼 수 있는 기회를 제공하는 적극적인 지식 이전 방법을 모색하는 것이 좋습니다. 예를 들어, AWS 전문 서비스에서는 대규모 마이그레이션을 담당하는 시스템 통합업체 (SI) AWS 파트너 3명을 대상으로 여러 클라우드 마이그레이션 팩토리 교육 세션을 제공했습니다. 이를 통해 팀은 마이그레이션 단계로 넘어갈 때 기본적인 지식을 갖추 수 있었습니다. 또한 각 SI 파트너 팀 내에서 1차 에스컬레이션 역할을 할 수 있는 주제 전문가 (SME) 를 식별하는 데도 도움이 되었습니다. AWS

기술 관점

기술은 대규모 마이그레이션을 가속화하기 위한 훌륭한 기반을 제공합니다. 예를 들어, 클라우드 마이그레이션 팩토리 솔루션은 마이그레이션을 end-to-end 자동화하는 방법에 중점을 둡니다. 이 섹션에서는 범위, 전략 및 일정에 맞춰 기술을 사용하여 필요한 규모와 속도를 달성하는 몇 가지 모범 사례를 살펴봅니다.

가장 중요한 원칙은 가능한 경우 자동화 영역을 살펴보는 것입니다. 대상 서버가 수천 대인 경우 수동으로 작업을 수행하는 데 많은 비용과 시간이 소요될 수 있습니다.

마이그레이션을 수행할 때는 일반적으로 다음과 같은 몇 가지 도구가 사용됩니다.

- Discovery
- 마이그레이션 구현
- 구성 관리 데이터베이스 (CMDB)
- 인벤토리 스프레드시트
- 프로젝트 관리

이러한 도구는 평가에서 동원, 구현에 이르는 다양한 마이그레이션 단계에서 사용됩니다. 비즈니스 목표와 일정에 따라 이러한 도구를 선택할 수 있습니다.

마이그레이션 단계를 계획한 후 다음 단계는 마이그레이션 팀이 필요한 도구를 사용할 수 있는 기술을 갖추도록 하는 것입니다. 팀에 기술이나 경험이 부족한 경우 대상 교육을 계획하여 기술을 강화하세요.

가능하면 팀들이 안전한 환경에서 마이그레이션 도구를 사용해 볼 수 있는 이벤트를 만드세요. 예를 들어, 팀에서 이 도구를 사용하여 마이그레이션할 수 있는 샌드팟 또는 랩 서버가 있나요? 아니면 초기 개발 워크로드를 학습 목적으로 사용하는 것이 허용되나요?

자동화, 추적 및 툴링 통합

이 섹션:

- [마이그레이션 검색을 자동화하여 필요한 시간을 단축합니다.](#)
- [반복적인 작업을 자동화하세요.](#)
- [추적 및 보고를 자동화하여 의사 결정 속도를 높입니다.](#)
- [마이그레이션을 촉진할 수 있는 도구를 살펴보세요.](#)

마이그레이션 검색을 자동화하여 필요한 시간을 단축합니다.

대부분의 대규모 마이그레이션 프로그램은 마이그레이션의 범위 (마이그레이션해야 할 대상) 를 이해하고 전략 (마이그레이션 방법) 을 개발하는 것으로 시작됩니다. 발견은 이것의 중요한 측면입니다. 필요한 메타데이터 포인트를 캡처하여 마이그레이션 전략 의사 결정 트리를 구성합니다. 워크로드를 속도에 맞게 마이그레이션하려면 필요한 마이그레이션 메타데이터를 식별하여 마이그레이션 팩토리와 같은 구현 프로세스로 가져와야 합니다. 마이그레이션 메타데이터를 추출, 변환, 로드 (ETL) 하는 완전 자동화된 메커니즘은 검색 프로세스에 소요되는 시간과 노력을 크게 줄여줍니다.

한 고객은 마이그레이션 팩토리를 위한 완전 자동화된 데이터 수집 프로세스를 개발했습니다. 모든 마이그레이션 메타데이터가 포함된 마이그레이션 웨이브 플랜은 SharePoint Microsoft의 스프레드시트에서 호스팅되고 유지 관리되었습니다. 원본이 변경되면 수동 개입 없이 데이터를 마이그레이션 팩토리로 로드하는 AWS Lambda 기능이 시작되었습니다. 이 자동화된 데이터 수집 프로세스를 통해 고객은 수동 작업을 줄이고 인적 오류를 최소화하며 작업 속도를 가속화할 수 있었습니다. 1,000대 이상의 서버를 로 마이그레이션할 수 있었습니다AWS.

반복적인 작업을 자동화하세요.

마이그레이션 구현 단계에서는 많은 소규모 프로세스를 자주 반복해야 합니다. 예를 들어 AWS Application Migration Service (MGN) 을 사용하는 경우 마이그레이션 범위 내에 있는 각 서버에 에이전트를 설치해야 합니다.

성공적인 대규모 마이그레이션을 제공하는 데 필요한 효율성과 속도를 달성하는 가장 효과적인 방법은 특정 비즈니스 및 기술 요구 사항에 맞는 마이그레이션 팩토리를 구축하는 것입니다. 마이그레이션 팩토리는 표준화된 데이터 세트를 사용하여 마이그레이션을 가속화하는 통합 및 오케스트레이션 프레임워크입니다.

임워크를 제공합니다. 모든 작업을 식별한 후에는 규범적 런북과 함께 자동화할 수 있는 모든 수동 작업을 자동화하는 데 시간을 할애하십시오.

[클라우드 마이그레이션 팩토리](#) 솔루션이 그 예입니다. 클라우드 마이그레이션 팩토리는 조직 고유의 측면을 자동화할 수 있는 마이그레이션 자동화 기반을 제공하도록 설계되었습니다. 예를 들어, 이제 온프레미스 서버를 폐기할 수 있다는 점을 강조하기 위해 CMDB의 플래그를 업데이트해야 할 수 있습니다. 이 시나리오에서는 마이그레이션 웨이브가 끝날 때 이 작업을 수행하는 자동화를 만들 수 있습니다. Cloud Migration Factory에는 모든 웨이브, 애플리케이션 및 서버 메타데이터가 포함된 중앙 집중식 메타데이터 저장소가 있습니다. 자동화 스크립트는 클라우드 마이그레이션 팩토리에 연결하여 해당 웨이브의 서버 목록을 가져오고 그에 따라 작업을 수행할 수 있습니다. 클라우드 마이그레이션 팩토리가 지원됩니다 [AWS Application Migration Service](#).

추적 및 보고를 자동화하여 의사 결정 속도를 높입니다.

프로그램의 핵심 성과 지표 (KPI) 를 포함한 실시간 데이터를 추적하고 보고하려면 자동화된 마이그레이션 보고 대시보드를 구축하는 것이 좋습니다. 마이그레이션 프로젝트에는 다음을 포함하여 조직 전체의 이해 관계자가 참여합니다.

- 애플리케이션 팀
- 테스터
- 해체 팀
- 아키텍트
- 인프라 팀
- 리더십

이러한 이해관계자들은 역할을 수행하기 위해 실시간 데이터가 필요합니다. 예를 들어 네트워크 팀은 온프레미스 리소스와 AWS 리소스 간의 공유 연결에 미치는 영향을 이해하려면 다가오는 마이그레이션 물결을 알아야 합니다. 경영진은 마이그레이션이 얼마나 완료되었는지 알고 싶어합니다. 신뢰할 수 있고 자동화된 실시간 데이터 피드를 통해 잘못된 의사소통을 방지하고 결정을 내릴 수 있는 기반을 마련할 수 있습니다.

한 대형 의료 고객은 마감일이 다가옴에 따라 데이터 센터 출구를 위해 노력하고 있었습니다. 규모와 복잡성을 고려했을 때 처음에는 이해 관계자 간에 마이그레이션 상태를 추적하고 전달하는 데 상당한 시간이 소요되었습니다. 마이그레이션 팀은 나중에 QuickSight Amazon을 사용하여 데이터를 시각화하는 자동화된 대시보드를 구축하여 추적 및 통신을 크게 단순화하는 동시에 마이그레이션 속도를 높였습니다.

마이그레이션을 촉진할 수 있는 도구를 살펴보세요.

마이그레이션에 적합한 도구를 선택하는 것은 쉽지 않습니다. 특히 조직 내에서 이전에 대규모 마이그레이션을 관리한 적이 없는 경우에는 더욱 그렇습니다.

시간을 들여 마이그레이션을 지원하는 데 적합한 도구를 선택하는 것이 좋습니다. 이 탐색에는 라이선스 비용이 포함될 수 있지만, 보다 광범위한 이니셔티브를 고려할 때 비용상의 이점을 제공할 수 있습니다. 또는 조직에 내장된 도구를 사용하여 비슷한 결과를 얻을 수도 있습니다. 예를 들어 풍부한 검색 정보를 제공할 수 있는 애플리케이션 성능 모니터링 도구가 이미 자산 전체에 배포되어 있을 수 있습니다.

한 기술 고객은 처음에는 익숙하지 않아 마이그레이션 중에 자동 검색 도구를 실행하는 것을 꺼렸습니다. 그 결과 SI AWS 파트너는 서버 이름, 운영 체제 버전, 종속성 등을 포함한 자산을 수작업으로 검색하기 위해 애플리케이션당 510시간의 회의를 진행해야 했습니다. 검색 도구를 사용했다면 검색 노력을 1,000시간 이상 줄일 수 있었을 것으로 추산되었습니다.

사전 요구 사항 및 마이그레이션 후 검증

이 섹션:

- [마이그레이션 전 단계에서 Landing Zone을 구축하세요](#)
- [사전 필수 활동을 간략하게 설명하십시오.](#)
- [지속적인 개선을 위해 마이그레이션 후 점검을 실시하십시오.](#)

마이그레이션 전 단계에서 Landing Zone을 구축하세요

마이그레이션 물결 중에 AWS 대상 가상 사설 클라우드 (VPC) 및 서브넷을 구축하는 대신 대상 환경 또는 랜딩 존 (landing zone) 을 미리 구축하는 것이 좋습니다. 잘 설계된 랜딩 존을 구축하는 것은 마이그레이션을 위한 전제 조건입니다. Landing Zone에는 모니터링, 거버넌스, 운영 및 보안 통제가 포함되어야 합니다.

마이그레이션에 앞서 랜딩 존을 구축하고 검증하면 새로운 환경에서 워크로드를 실행할 때 발생하는 불확실성을 최소화할 수 있습니다. Landing Zone이 마련되면 이해 관계자는 계정 또는 VPC 수준에서 관리되는 측면에 대해 걱정하지 않고 워크로드를 마이그레이션하는 데 집중할 수 있습니다.

사전 필수 활동을 간략하게 설명하십시오.

Landing Zone과 함께 마이그레이션 전에 다른 기술적 사전 요구 사항, 특히 리드 타임이 긴 프로세스를 조정하는 것이 중요합니다. 예를 들어, 데이터를 온프레미스로 복제할 수 있도록 필요한 방화벽을 변경

하십시오. AWS 기술 사전 요구 사항을 조기에 알리면 필요한 리소스를 준비하고 할당하는 데 도움이 됩니다. 사전 요구 사항이 충족되지 않아 마이그레이션이 중단되는 경우가 많습니다. 이는 진행 중인 마이그레이션 물결에 영향을 미칠 뿐만 아니라 문제가 해결되는 동안 향후 모든 마이그레이션 날짜가 연기될 수 있습니다.

한 금융 서비스 회사는 여러 데이터 센터를 비우는 것을 목표로 대량 마이그레이션을 수행하려고 AWS 했습니다. 하지만 온프레미스 간에 사용할 수 있는 대역폭은 의도한 속도에 비해 충분하지 AWS 않았 습니다. 안타깝게도 대역폭을 늘리려면 새 연결이 필요했고 리드 타임은 3개월이었습니다. 즉, 처음 3 개월 동안은 마이그레이션 속도가 제한되었습니다.

지속적인 개선을 위해 마이그레이션 후 점검을 실시하십시오.

마지막으로, 운영 통합, 비용 최적화, 거버넌스 및 규정 준수 검사와 같은 마이그레이션 후 검증을 구현 해야 합니다. 마이그레이션 후 검증에는 이전에 마이그레이션된 워크로드를 평가하여 미래의 흐름에 적용해야 하는 기술적 교훈을 찾아내는 것이 포함됩니다.

또한 이는 비용 관리 작업을 구현할 수 있는 좋은 기회입니다. 예를 들어, 마이그레이션 중에 성능 테스트의 필요성을 줄이기 위해 AWS 인스턴스의 크기를 온프레미스 자산에 맞게 조정하기로 결정할 수 있습니다. 이제 테스트는 더 이상 데이터 센터 폐쇄의 중요한 경로가 아니므로 Amazon을 CloudWatch 사용하여 인스턴스 사용률을 평가하고 더 작은 크기의 인스턴스가 적합한지 여부를 결정할 수 있습니다.

이 단계의 중요성을 설명하기 위해 대규모 기술 고객이 대규모 마이그레이션을 수행했지만 처음에는 마이그레이션 후 검증을 포함하지 않았습니다. 100개가 넘는 서버를 마이그레이션한 후 AWS Systems Manager 에이전트 (SSM Agent) 가 올바르게 구성되지 않았음을 확인했습니다. 이전에 마이그레이션한 모든 서버를 수정해야 했지만 마이그레이션이 중단되었습니다. 또한 고객은 인스턴스가 초기 추정치의 5배에 달한다는 사실을 파악하여 각 마이그레이션 작업이 끝날 때마다 비용 체크포인트를 구현했습니다.

프로세스 관점

프로세스는 일관성을 제공하지만 각 프로젝트가 고유하기 때문에 진화하고 변경되기 쉽습니다. 프로세스를 반복해서 실행하면 실패하고, 배우고, 채택하고, 반복할 때 큰 이익을 가져올 수 있는 격차와 개선의 여지를 파악할 수 있습니다. 이러한 변화는 프로젝트와 비즈니스가 미래에 활용할 수 있는 새로운 아이디어나 혁신으로 이어질 수 있으며, 이는 품질과 팀 신뢰를 가져다주는 성장의 촉매제가 됩니다.

마이그레이션 프로세스는 이전에는 연결되지 않았을 수 있는 기술과 경계를 넘나들기 때문에 복잡할 수 있습니다. 이 관점은 대규모 마이그레이션의 특정 요구 사항에 대한 프로세스와 지침을 제공합니다.

대규모 마이그레이션 준비

다음 섹션에서는 성공에 중요한 이해 관계자의 동의와 명확한 방향에 따라 마이그레이션 여정을 시작하는 데 필요한 핵심 원칙을 간략하게 설명합니다.

이 섹션:

- [비즈니스 동인을 정의하고 일정, 범위, 전략을 전달하세요.](#)
- [방해 요소를 제거하는 데 도움이 되는 명확한 에스컬레이션 경로를 정의하십시오.](#)
- [불필요한 변경을 최소화하세요](#)
- [end-to-end 프로세스를 조기에 문서화하세요.](#)
- [표준 마이그레이션 패턴 및 아티팩트를 문서화하십시오.](#)
- [마이그레이션 메타데이터 및 상태에 대한 신뢰할 수 있는 단일 소스를 구축하십시오.](#)

비즈니스 동인을 정의하고 일정, 범위, 전략을 전달하세요.

로 대규모 마이그레이션을 진행하다 보면 서버를 마이그레이션하는 방법이 다양하다는 사실을 금방 알게 될 것입니다. AWS 예를 들어 다음을 수행할 수 있습니다.

- 를 사용하여 워크로드를 재호스팅합니다. [AWS Application Migration Service](#)
- 애플리케이션을 컨테이너화하고 Amazon [Elastic Container Service \(Amazon ECS\)](#) 또는 [Amazon EKS \(Elastic Kubernetes Service\)](#) 관리 컨테이너 플랫폼에서 호스팅하십시오.
- 워크로드를 완전한 서버리스 애플리케이션으로 재설계하십시오.

올바른 마이그레이션 경로를 결정하려면 비즈니스 동인을 거꾸로 검토하는 것이 중요합니다. 궁극적인 목표가 비즈니스 민첩성을 높이는 것이라면 더 높은 수준의 혁신이 필요한 두 번째 패턴을 선호할 수 있습니다. 연말까지 데이터 센터를 비우는 것이 목표라면 재호스팅이 제공하는 속도 때문에 워크로드를 재호스팅하는 방법을 선택할 수 있습니다.

대규모 마이그레이션에는 일반적으로 다음을 비롯한 다양한 이해 관계자가 참여합니다.

- 애플리케이션 소유자
- 네트워크 팀
- 데이터베이스 관리자
- 총괄 스폰서

마이그레이션의 비즈니스 동인을 파악하고 마이그레이션 프로그램 구성원이 액세스할 수 있는 프로젝트 현장과 같은 문서에 해당 목록을 포함하는 것이 중요합니다. 또한 목표 비즈니스 성과와 밀접하게 일치하는 핵심 성과 지표 (KPI) 를 작성하십시오.

예를 들어, 한 고객은 데이터 센터를 비우는 목표 비즈니스 성과를 달성하기 위해 12개월 이내에 2,000대의 서버를 마이그레이션하려고 했습니다. 하지만 이 회사의 보안 팀은 이 목표를 향해 나아가지 못했습니다. 그 결과 데이터 센터 폐쇄일을 놓치고 애플리케이션을 더 현대화할 것인지, 아니면 초기에 데이터 센터를 적시에 폐쇄한 다음 애플리케이션을 현대화할 수 있도록 호스트를 변경할 것인지를 두고 몇 달 간의 기술적 논쟁이 벌어졌습니다. AWS

방해 요소를 제거하는 데 도움이 되는 명확한 에스컬레이션 경로를 정의하십시오.

대규모 클라우드 마이그레이션 프로그램에는 일반적으로 다양한 이해 관계자가 참여합니다. 결국 수십 년 동안 온프레미스에서 호스팅되어 온 애플리케이션이 변경될 가능성이 있습니다. 각 이해 관계자의 우선 순위가 서로 상충되는 것은 흔한 일입니다.

모든 우선 순위가 가치를 창출할 수 있지만 프로그램의 예산 규모가 제한되고 목표 결과가 정해져 있을 가능성이 높습니다. 다양한 이해관계자를 관리하고 목표 비즈니스 성과에 집중하는 것은 어려울 수 있습니다. 마이그레이션 범위에 포함되는 수백 또는 수천 개의 애플리케이션을 곱하면 이 문제는 더욱 복잡해집니다. 게다가 이해관계자들은 우선 순위가 다른 여러 리더십 팀에 보고할 가능성이 높습니다. 이를 염두에 두고 목표 비즈니스 성과를 명확하게 문서화하는 동시에 방해 요소를 제거하는 데 도움이 되는 명확한 에스컬레이션 매트릭스를 정의하는 것이 중요합니다. 이를 통해 상당한 시간을 절약하고 여러 팀을 공동의 목표를 향해 조율하는 데 도움이 될 수 있습니다.

이를 잘 보여주는 한 가지 예가 12개월 내에 주 데이터 센터를 비우는 것을 목표로 삼은 금융 서비스 회사를 들 수 있습니다. 명확한 의무나 에스컬레이션 경로가 없었기 때문에 이해 관계자들은 시간과 예산의 제약에 관계없이 원하는 마이그레이션 경로를 만들게 되었습니다. CIO로 에스컬레이션한 후 명확한 권한이 설정되었고 필요한 결정을 요청하기 위한 메커니즘이 제공되었습니다.

불필요한 변경을 최소화하세요

변화는 좋지만 변화가 많을수록 위험도 커집니다. 대규모 마이그레이션에 대한 비즈니스 사례가 승인되면 특정 날짜까지 데이터 센터를 비우는 것과 같이 이 이니셔티브를 주도하는 목표 비즈니스 성과가 있을 가능성이 높습니다. 기술 전문가들이 AWS 서비스를 최대한 활용하기 위해 모든 것을 다시 작성하려는 것은 흔한 일이지만, 이것이 비즈니스 목표가 아닐 수도 있습니다.

한 고객은 2년 동안 회사의 전체 웹 스케일 인프라를 다음으로 마이그레이션하는 데 집중했습니다. AWS 그들은 애플리케이션 팀이 애플리케이션을 다시 작성하는 데 몇 개월을 소비하지 않도록 하기 위한 메커니즘으로 2주 규칙을 만들었습니다. 2주 규칙을 사용함으로써 고객은 수년 동안 수백 개의 애플

리케이션을 이동해야 하는 상황에서도 일관된 주기로 장기 마이그레이션을 유지할 수 있었습니다. 자세한 내용은 블로그 게시물 ['2주 규칙: 10일 내에 클라우드용으로 애플리케이션을 리팩토링하는 것'](#)을 참조하십시오.

비즈니스 성과와 맞지 않는 변화는 최소화하는 것이 좋습니다. 대신 향후 프로젝트에서 이러한 추가 변경 사항을 관리할 수 있는 메커니즘을 구축하세요.

end-to-end 프로세스를 조기에 문서화하세요.

대규모 마이그레이션 프로그램의 초기 단계에서 전체 마이그레이션 프로세스와 소유권 할당을 문서화하세요. 이 문서는 모든 이해 관계자에게 마이그레이션의 진행 방식과 역할 및 책임에 대해 교육하는 데 중요합니다. 이 설명서는 또한 문제가 발생할 수 있는 위치를 이해하고 마이그레이션을 진행하면서 프로세스를 업데이트하고 반복하는 데 도움이 됩니다.

마이그레이션 프로젝트를 개발하는 동안 기존 프로세스를 이해하고 통합 지점과 종속성을 명확하게 문서화해야 합니다. 변경 요청, 서비스 요청, 공급업체 지원, 네트워크 및 방화벽 지원 등 외부 프로세스 소유자와의 협력이 필요한 곳을 포함하세요. 프로세스를 이해한 후에는 RACI (책임, 책임, 자문, 정보 제공) 매트릭스에 소유권을 문서화하여 다양한 활동을 추적하는 것이 좋습니다. 프로세스를 마무리하려면 마이그레이션의 각 단계와 관련된 일정을 파악하여 카운트다운 계획을 수립하세요. 카운트다운 계획은 일반적으로 워크로드 마이그레이션 컷오버 날짜 및 시간을 거꾸로 적용합니다.

이 문서화 방식은 1년도 채 되지 않아 데이터 센터 4곳을 철수하면서 AWS 성공적으로 마이그레이션한 다국적 가전제품 기업에 효과적이었습니다. 여기에는 6개의 서로 다른 조직 팀과 여러 제3자가 참여했기 때문에 관리 오버헤드가 발생하여 back-and-forth 의사 결정이 내려지고 구현이 지연되었습니다. AWS프로페셔널 서비스 팀은 고객 및 제3자와 함께 마이그레이션 활동의 주요 프로세스를 파악하고 각 소유자와 함께 이를 문서화했습니다. 그 결과로 나온 RACI 매트릭스는 모든 관련 팀이 공유하고 합의했습니다. 고객은 RACI 매트릭스와 에스컬레이션 매트릭스를 사용하여 지연을 초래하는 장애물과 문제를 완화했습니다. 그러자 그들은 예정보다 빨리 데이터 센터를 떠날 수 있었습니다.

RACI와 에스컬레이션 매트릭스를 사용한 또 다른 예로, 한 보험 회사는 4개월 이내에 데이터 센터를 종료할 수 있었습니다. 고객은 공동 책임 모델을 이해하고 구현했으며 상세한 RACI 매트릭스를 바탕으로 마이그레이션 전반에 걸친 각 프로세스와 활동의 진행 상황을 추적했습니다. 그 결과 고객은 구현 첫 12주 동안 350대 이상의 서버를 마이그레이션할 수 있었습니다.

표준 마이그레이션 패턴 및 아티팩트를 문서화하십시오.

이를 구현을 위한 쿠키 커터를 만드는 것과 같다고 생각하세요. 재사용 가능한 참조, 문서, 런북, 패턴이 확장의 핵심입니다. 이들은 향후 마이그레이션 프로젝트에서 재사용하고 피할 수 있는 경험, 학습, 합성, 문제 및 솔루션을 일지에 기록하여 마이그레이션을 크게 가속화합니다. 패턴과 아티팩트는 프로세스를 개선하고 향후 프로젝트를 안내하는 데 도움이 되는 투자이기도 합니다.

예를 들어, 한 고객은 서로 다른 세 SI 파트너가 1년 동안 애플리케이션을 마이그레이션하는 마이그레이션을 수행했습니다. AWS 초기 단계에서 각 AWS 파트너는 자체 표준, 런북 및 아티팩트를 사용하고 있었습니다. 동일한 정보가 다른 방식으로 제공될 수 있기 때문에 이로 인해 고객 팀은 많은 부담을 안게 되었습니다. 초기의 이러한 어려움을 겪은 후, 고객은 권장 변경 사항을 제출하는 프로세스를 통해 마이그레이션에 사용할 모든 문서와 아티팩트의 중앙 소유권을 설정했습니다. 이러한 자산에는 다음이 포함됩니다.

- 표준 마이그레이션 프로세스 및 체크리스트
- 네트워크 다이어그램 스타일 및 형식 표준
- 비즈니스 중요도에 기반한 애플리케이션 아키텍처 및 보안 표준

또한 이러한 문서 및 표준에 대한 변경 사항은 매주 한 번씩 모든 팀에 발송되었으며, 각 파트너는 변경 사항의 접수 및 준수 여부를 확인해야 했습니다. 이를 통해 마이그레이션 프로젝트의 의사소통과 일관성이 크게 향상되었으며, 다른 사업부에서 별도의 대규모 마이그레이션 작업이 시작되었을 때 해당 팀은 기존 프로세스와 문서를 채택하여 성공을 크게 가속화할 수 있었습니다.

마이그레이션 메타데이터 및 상태에 대한 신뢰할 수 있는 단일 소스를 구축하십시오.

대규모 마이그레이션을 계획할 때는 다양한 팀을 조율하고 데이터에 기반한 결정을 내릴 수 있도록 신뢰할 수 있는 출처를 마련하는 것이 중요합니다. 이 과정을 시작하면 구성 관리 데이터베이스 (CMDB), 응용 프로그램 성능 모니터링 도구, 인벤토리 목록 등과 같이 사용할 수 있는 수많은 데이터 원본을 찾을 수 있습니다.

또는 데이터 원본이 거의 없어서 필요한 데이터를 캡처하기 위한 메커니즘을 만들어야 할 수도 있습니다. 예를 들어, 검색 도구를 사용하여 기술 정보를 발견하고 IT 리더를 대상으로 설문조사를 실시하여 비즈니스 정보를 얻어야 할 수 있습니다.

다양한 데이터 소스를 마이그레이션에 사용할 수 있는 단일 데이터 세트로 집계하는 것이 중요합니다. 그런 다음 구현 중에 마이그레이션을 추적하는 데 신뢰할 수 있는 단일 소스를 사용할 수 있습니다. 예를 들어 어떤 서버가 마이그레이션되었는지 추적할 수 있습니다.

제공된 데이터 세트를 사용하여 마이그레이션을 계획하는 AWS 데 집중하기 위해 모든 워크로드를 마이그레이션하려는 금융 서비스 고객. 이 데이터셋에는 비즈니스 중요도 및 종속성 정보와 같은 주요 격차가 있었기 때문에 프로그램은 검색 작업을 시작했습니다.

또 다른 예로, 같은 업계의 한 회사가 서버 인프라 인벤토리에 out-of-date 대한 이해를 바탕으로 마이그레이션 웨이브 구현에 뛰어 들었습니다. 데이터가 잘못되었기 때문에 마이그레이션 건수가 빠르게 감소하기 시작했습니다. 이 경우 애플리케이션 소유자를 이해할 수 없었고, 이는 테스터를 제때 찾을 수

없었다는 것을 의미했습니다. 또한 애플리케이션 팀이 완료한 서비스 해체 작업과 데이터가 일치하지 않아 서버가 업무용으로 사용되지 않고 운영되고 있었습니다.

대규모 마이그레이션 실행

비즈니스 성과를 수립하고 이해 관계자에게 전략을 전달한 후에는 대규모 마이그레이션의 범위를 지속 가능한 마이그레이션 이벤트 또는 물결에 맞게 확대하는 방법을 계획할 수 있습니다. 다음 예는 물결 계획을 세우는 데 필요한 주요 지침을 제공합니다.

이 섹션:

- [마이그레이션 물결을 미리 계획하여 안정적인 흐름을 확보하세요.](#)
- [웨이브 구현과 웨이브 계획을 별도의 프로세스와 팀으로 유지하십시오.](#)
- [작게 시작해서 큰 성과를 거둘 수 있습니다.](#)
- [컷오버 윈도우 수를 최소화하세요.](#)
- [빨리 실패하고, 경험을 적용하고, 반복하세요](#)
- [회고전도 잊지 마세요.](#)

마이그레이션 물결을 미리 계획하여 안정적인 흐름을 확보하세요.

마이그레이션 계획은 프로그램에서 가장 중요한 단계 중 하나입니다. “계획을 세우지 못하면 실패할 계획을 세웁니다.” 라는 말과 함께 쓰여집니다. 마이그레이션 물결을 미리 계획하면 팀이 마이그레이션 상황에 보다 능동적으로 대처하여 프로젝트를 신속하게 진행할 수 있습니다. 이를 통해 프로젝트를 더 쉽게 확장할 수 있으며, 프로젝트 요구가 증가하고 복잡해짐에 따라 의사 결정 및 예측이 개선됩니다. 미리 계획을 세우면 팀이 변화에 적응하는 능력도 향상됩니다.

예를 들어, 한 대규모 금융 서비스 고객이 데이터 센터 출구 프로그램을 개발 중이었습니다. 처음에 고객은 마이그레이션 단계를 순차적으로 계획하여 다음 단계를 계획하기 전에 한 단계를 완료했습니다. 이 접근 방식 덕분에 준비 시간이 단축되었습니다. 이해 관계자들은 애플리케이션이 마이그레이션된다는 소식을 들었을 AWS 때 마이그레이션을 시작하기 전에 수행해야 할 몇 가지 단계가 남아 있었습니다. 이로 인해 프로그램이 크게 지연되었습니다. 고객은 이 사실을 깨닫고 몇 달 전에 마이그레이션 물결을 미리 계획하는 총체적이고 미래 지향적인 마이그레이션 계획 스트림을 구현했습니다. 이를 통해 애플리케이션 팀은 충분한 공지를 통해 AWS 파트너에게 알리고, 라이선싱 분석 등과 같은 마이그레이션 전 작업을 수행할 수 있었습니다. 그러면 해당 작업을 프로그램의 주요 경로에서 제거할 수 있습니다.

웨이브 구현과 웨이브 계획을 별도의 프로세스와 팀으로 유지하십시오.

웨이브 플래닝 팀과 웨이브 구현 팀이 분리되어 있으면 두 프로세스가 병렬로 작동할 수 있습니다. 의사소통과 조정을 통해 예상 속도를 달성할 준비가 된 서버나 애플리케이션이 충분하지 않아 마이그레이션 속도가 느려지는 것을 방지할 수 있습니다. 예를 들어 마이그레이션 팀은 매주 30대의 서버를 마이그레이션해야 하는데 현재 상황에서는 10대의 서버만 준비할 수 있습니다. 이 문제는 주로 다음과 같은 원인으로 인해 발생합니다.

- 마이그레이션 구현 팀은 웨이브 플래닝에 관여하지 않았으며 웨이브 플래닝 단계에서 수집된 데이터는 아직 완전하지 않습니다. 마이그레이션 구현 팀은 웨이브를 시작하기 전에 더 많은 서버 데이터를 수집해야 합니다.
- 마이그레이션 구현은 웨이브 계획 직후부터 시작되며, 중간에 버퍼 없이 바로 시작될 예정입니다.

웨이브를 미리 계획하고 웨이브 구현 준비와 시작 사이에 버퍼를 만드는 것이 중요합니다. 웨이브 계획 팀과 마이그레이션 팀이 협력하여 올바른 데이터를 수집하고 재작업을 방지하는 것도 중요합니다.

작게 시작해서 큰 성과를 거둘 수 있습니다.

소규모로 시작하여 후속 물결이 발생할 때마다 마이그레이션 속도를 높이도록 계획하세요. 초기 웨이브는 서버 10대 미만의 소규모 단일 애플리케이션이어야 합니다. 후속 단계에서 애플리케이션과 서버를 추가하여 전체 마이그레이션 속도를 높이십시오. 덜 복잡하거나 위험한 애플리케이션의 우선 순위를 정하고 일정에 따라 속도를 높이면 팀이 공동 작업에 적응하고 프로세스를 익힐 시간을 확보할 수 있습니다. 또한 팀은 각 웨이브에 대한 프로세스 개선 사항을 파악하고 구현할 수 있으며, 이를 통해 이후 웨이브의 속도를 크게 개선할 수 있습니다.

한 고객은 1년에 1,300대 이상의 서버를 마이그레이션하고 있었습니다. 파일럿 마이그레이션을 시작하고 몇 번의 소규모 단계를 거쳐 마이그레이션 팀은 이후 마이그레이션을 개선할 수 있는 여러 가지 방법을 찾아낼 수 있었습니다. 예를 들어, 그들은 이전에 새로운 데이터 센터 네트워크 세그먼트를 식별했습니다. 그들은 프로세스 초기에 방화벽 팀과 협력하여 마이그레이션 도구와의 통신을 허용하는 방화벽 규칙을 마련했습니다. 이를 통해 미래 파도가 불필요하게 지연되는 것을 방지할 수 있었습니다. 또한 팀은 각 웨이브에서 더 많은 검색 및 컷오버 프로세스를 자동화하는 데 도움이 되는 스크립트를 개발할 수 있었습니다. 소규모로 시작한 덕분에 팀은 초기 프로세스 개선에 집중하고 자신감을 크게 높일 수 있었습니다.

컷오버 윈도우 수를 최소화하세요.

대량 마이그레이션에는 규모 확대를 위한 체계적인 접근 방식이 필요합니다. 일부 지역에서 너무 유연하게 대처하면 대규모 마이그레이션을 할 수 없습니다. 주간 컷오버 기간의 수를 제한함으로써 컷오버 활동에 소요되는 시간이 더 큰 가치를 지닙니다.

예를 들어 컷오버 기간이 너무 유연하면 각각 서버 5대로 구성된 20개의 컷오버가 발생할 수 있습니다. 대신 각각 50대의 서버를 포함하는 컷오버를 두 번 만들 수 있습니다. 각 컷오버에 드는 시간과 노력이 비슷하기 때문에 컷오버의 수가 적고 규모가 커지면 스케줄링의 운영 부담이 줄어들고 불필요한 지연이 줄어듭니다.

한 대형 기술 회사는 계약 만료 전에 임대 데이터 센터 몇 군데에서 다른 곳으로 마이그레이션하려고 했습니다. 만료를 놓치면 비용이 많이 드는 단기 갱신 기간이 발생할 수 있습니다. 마이그레이션 초기에는 애플리케이션 팀이 마지막 순간까지 마이그레이션 일정을 결정할 수 있었습니다. 여기에는 컷오버를 며칠 앞두고 어떤 이유로든 마이그레이션을 거부하는 것도 포함됩니다. 이로 인해 프로젝트 초기 단계에서 많은 지연이 발생했습니다. 고객이 마지막 순간에 다른 애플리케이션 팀과 협의해 내용을 입력해야 하는 경우가 많았습니다. 결국 고객은 계획 원칙을 강화했지만 초기의 이러한 실수로 인해 마이그레이션 팀은 계속해서 스트레스를 받게 되었습니다. 전체 일정이 지연되면서 일부 애플리케이션이 제시간에 데이터 센터를 떠나지 못하는 상황이 발생했습니다.

빨리 실패하고, 경험을 적용하고, 반복하세요

모든 마이그레이션에는 처음에는 함정이 따릅니다. 초기에 실패하면 팀이 배우고, 병목 현상을 이해하고, 배운 교훈을 대규모 물결에 적용하는 데 도움이 됩니다. 마이그레이션의 처음 두 번의 웨이브는 다음과 같은 이유로 느릴 것으로 예상됩니다.

- 팀원들은 서로 적응하고 프로세스에 적응하고 있습니다.
- 대규모 마이그레이션에는 일반적으로 다양한 도구와 인력이 필요합니다.
- end-to-end 프로세스를 통합하고, 테스트하고, 실패하고, 학습하고, 지속적으로 개선하는 데에는 시간이 걸립니다.

문제는 흔히 발생하며 처음 몇 번의 파동에서 발생할 것으로 예상됩니다. 일부 팀은 새로운 것을 시도하고 실패하는 것을 싫어할 수 있으므로 이 점을 이해하고 조직 전체에 전달하는 것이 중요합니다. 실패는 팀을 낙담시키고 향후 마이그레이션을 방해할 수 있습니다. 초기 문제가 업무의 일부라는 점을 모든 사람이 이해하고 모든 사람이 시도하고 실패하도록 장려하는 것이 성공적인 마이그레이션의 핵심입니다.

한 회사는 24~36개월 내에 10,000대 이상의 서버를 마이그레이션할 계획을 세웠습니다. 이 목표를 달성하려면 한 달에 거의 300대의 서버를 마이그레이션해야 했습니다. 하지만 그렇다고 해서 첫날부터 300대의 서버를 마이그레이션했다는 의미는 아닙니다. 처음 몇 차례는 학습 단계였습니다. 이를 통해 팀은 작업이 어떻게 작동하고 누가 어떤 작업을 수행할 권한이 있는지 이해할 수 있었습니다. 또한 CMDB 및 와의 통합과 같이 프로세스를 개선할 수 있는 통합도 확인했습니다. CyberArk 그들은 학습 과정을 통해 실패하고, 개선하고, 다시 실패하면서 프로세스와 자동화를 개선했습니다. 6개월 후 그들은 매주 120대 이상의 서버를 마이그레이션할 수 있었습니다.

회고전도 잊지 마세요.

이는 애자일 프로세스의 중요한 부분입니다. 팀이 소통하고, 조정하고, 배우고, 동의하고, 앞으로 나아가는 곳입니다. 가장 기본적인 수준의 회고는 과거를 돌아보고 무슨 일이 있었는지 논의하고 잘된 점과 개선해야 할 점을 결정하는 것입니다. 그런 다음 이러한 논의를 바탕으로 개선 사항을 구축할 수 있습니다. 회고는 배운 교훈을 바탕으로 형식이나 프로세스를 어느 정도 정리합니다. 대규모 마이그레이션이 성공할 수 있는 규모와 속도를 달성하려면 프로세스, 도구 및 팀이 지속적으로 발전하고 개선되어야 하기 때문에 회고가 중요합니다. 회고는 이 점에서 중요한 역할을 할 수 있습니다.

기존의 교훈을 통해 학습한 세션은 프로그램이 끝날 때까지 진행되지 않으므로 다음 마이그레이션 물결이 시작될 때 이러한 교훈을 검토하지 못하는 경우가 많습니다. 대규모 마이그레이션의 경우 배운 교훈을 다음 단계에도 적용해야 하며, 이는 웨이브 계획 프로세스의 핵심 부분이 되어야 합니다.

한 고객의 경우 컷오버를 통해 얻은 교훈을 논의하고 문서화하기 위해 매주 회고를 실시했습니다. 이 세션에서는 프로세스 관점이나 자동화의 관점에서 간소화의 여지가 있는 영역을 찾아냈습니다. 그 결과 특정 활동, 소유자 및 자동화 스크립트가 포함된 카운트다운 일정을 구현하여 컷오버 중에 타사 도구 검증 및 Amazon CloudWatch 에이전트 설치 등 수동 작업을 최소화할 수 있었습니다.

또 다른 대형 기술 회사에서는 이전 마이그레이션 물결에서 발생한 문제를 식별하기 위해 팀과 함께 정기적인 회고를 실시했습니다. 그 결과 프로세스, 스크립트 및 자동화가 개선되어 프로그램 진행 과정에서 평균 마이그레이션 시간이 40% 단축되었습니다.

추가 고려 사항

대규모 마이그레이션 프로그램에는 많은 영역이 포함되어야 합니다. 다음 섹션에서는 고려해야 할 다른 항목에 대한 생각을 제공합니다.

이 섹션:

- [진행하면서 정리하세요.](#)
- [추가 변환을 위해 여러 단계를 구현하십시오.](#)

진행하면서 정리하세요.

예상보다 10배 많은 비용이 들고 마이그레이션에 사용된 리소스를 종료하고 정리할 때까지 프로젝트가 완료되지 않으면 마이그레이션은 성공한 것으로 간주되지 않습니다. 이 정리는 마이그레이션 후 활동의 일부여야 합니다. 이를 통해 비용을 가중시키는 사용되지 않은 리소스와 서비스가 환경에 남아 있지 않도록 할 수 있습니다. 마이그레이션 후 정리는 환경을 노출시키는 위협과 취약성을 방지하기 위한 좋은 보안 방법이기도 합니다.

로 전환하여 얻을 수 있는 두 가지 주요 결과는 비용 절감과 AWS 클라우드 보안입니다. 사용하지 않는 리소스를 그대로 두면 클라우드로 전환하려는 비즈니스 목적을 무산시킬 수 있습니다. 정리되지 않는 가장 일반적인 리소스는 다음과 같습니다.

- 테스트 데이터
- 테스트 데이터베이스
- 방화벽 규칙, 보안 그룹, 네트워크 ACL (액세스 제어 목록) IP 주소를 포함한 테스트 계정
- 테스트용으로 프로비저닝된 포트
- Amazon Elastic Block Store(Amazon EBS) 볼륨
- 스냅샷
- 복제 (예: 온프레미스에서 온프레미스로의 데이터 복제 중지) AWS
- 공간을 차지하는 파일 (예: 마이그레이션에 사용되는 임시 데이터베이스 백업)
- 마이그레이션 도구를 호스팅하는 인스턴스

잘못된 정리 관행의 한 예로, SI AWS 파트너는 마이그레이션에 성공한 후 복제 에이전트를 제거하지 않았습니다. AWS 감사 결과 이미 마이그레이션된 복제 서버와 EBS 볼륨의 경우 매월 20,000달러 (USD) 의 비용이 드는 것으로 나타났습니다. 이 문제를 완화하기 위해 AWS 전문 서비스에서는 오래된 서버가 여전히 복제되고 있는 경우 SI AWS 파트너에게 알리는 자동 감사 프로세스를 만들었습니다. 그러면 SI AWS 파트너가 사용하지 않고 오래된 인스턴스에 대해 조치를 취할 수 있습니다.

향후 마이그레이션을 위해 원활한 플랫폼 채택을 보장하기 위해 마이그레이션 후 하이퍼케어 기간을 48시간으로 정의하는 프로세스가 채택되었습니다. 그런 다음 고객의 인프라 팀이 온프레미스 서버의 서비스 중단 요청을 제출했습니다. 서비스 해제 요청이 승인되면 해당 웨이브의 서버가 애플리케이션 마이그레이션 서비스 콘솔에서 제거될 것이라는 안내를 받았습니다.

추가 변환을 위해 여러 단계를 구현하십시오.

대규모 마이그레이션을 수행할 때는 데이터 센터 폐쇄 또는 인프라 혁신과 같은 핵심 목표에 계속 집중하는 것이 중요합니다. 소규모 마이그레이션에서는 범위 변동이 영향을 최소화할 수 있습니다. 하지만 며칠간의 추가 작업에 수천 대의 서버를 더하면 프로그램에 상당한 시간이 소요될 수 있습니다. 또한 추가 변경 사항으로 인해 지원 팀을 위한 문서, 프로세스 및 교육에 대한 업데이트가 필요할 수도 있습니다.

잠재적인 범위 변동을 극복하기 위해 마이그레이션에 다단계 접근 방식을 구현할 수 있습니다. 예를 들어 데이터 센터를 비우는 것이 목표라면 1단계에는 워크로드를 최대한 빨리 재호스팅하는 것만 포함될 수 있습니다. AWS 워크로드를 재호스팅한 후 2단계에서는 목표 비즈니스 성과에 영향을 주지 않으면서 혁신 활동을 구현할 수 있습니다.

예를 들어, 한 고객은 12개월 후에 데이터 센터를 그만두려고 계획했습니다. 그러나 마이그레이션에는 새로운 애플리케이션 성능 모니터링 도구 출시 및 운영 체제 업그레이드와 같은 다른 혁신 활동이 포함되었습니다. 1,000대 이상의 서버가 마이그레이션 범위에 포함되었으므로 이러한 활동으로 인해 마이그레이션이 상당히 지연되었습니다. 또한 이 접근 방식을 사용하려면 새 도구 사용에 대한 교육이 필요했습니다. 고객은 나중에 처음에는 재호스팅에 중점을 두고 다단계 접근 방식을 구현하기로 결정했습니다. 이를 통해 마이그레이션 속도가 빨라지고 데이터 센터 폐쇄 날짜를 지키지 못할 위험이 줄어들었습니다.

결론

대규모 마이그레이션은 소규모 마이그레이션과 비교할 때 다른 문제를 야기합니다. 이는 주로 스케일로 인한 복잡성 때문입니다. 예를 들어 단일 서버에 에이전트를 설치하는 것은 매우 간단하며 약 5분이 소요됩니다. 하지만 마이그레이션 범위에 5,000대의 서버가 있는 경우 약 416시간이 소요되며 다음과 같은 문제가 발생할 수 있습니다.

- 서로 다른 프로세스를 필요로 하는 운영 체제가 여러 개 있을 수 있습니다.
- 이전 인수 합병으로 인해 관리해야 할 Microsoft Active Directory 도메인이 별도로 있을 수 있습니다.
- 각 웨이브에 대한 에이전트 설치를 조율한 다음 진행 상황을 추적하고 보고하려면 효과적인 프로세스와 도구가 필요합니다.

이 전략은 다양한 고객을 지원하는 AWS 전문 서비스 경험을 바탕으로 대규모 마이그레이션 모범 사례를 설명합니다. 여기에는 사람, 프로세스 및 기술 관점이 포함됩니다. 시작하고 싶거나 마이그레이션을 진행 중인 경우 AWS Professional Services의 컨설턴트가 기꺼이 도와드릴 것입니다. AWS 담당자에게 문의하여 대화를 시작하세요.

다음 단계에서는 대규모 마이그레이션을 계획하고 완료하는 데 도움이 되도록 설계된 AWS 규범적 지침 시리즈를 검토하는 것이 좋습니다. 전체 시리즈를 보려면 [로의 대규모 마이그레이션을 참조하십시오](#) AWS 클라우드.

리소스

AWS대규모 마이그레이션

대규모 마이그레이션에 대한 전체AWS 규범적 지침 시리즈를 보려면 [로의 대규모 마이그레이션을 참조하십시오](#) AWS 클라우드.

관련AWS 규범 지침 리소스

- [클라우드 마이그레이션 팩토리를 통한 대규모 서버 마이그레이션 자동화](#)
- [로 마이그레이션하는 동안 폐기될 애플리케이션을 평가하는 모범 사례](#) AWS 클라우드
- [안전하고 확장 가능한 다중 계정AWS 환경 설정](#)
- [마이그레이션 준비 상태 평가](#)
- [조직을 동원하여 대규모 마이그레이션을 가속화하세요](#)

추가 참고

- [AWS클라우드 마이그레이션 팩토리 솔루션](#)
- [무료 클라우드 마이그레이션 서비스 onAWS](#)
- [AWS Database Migration Service](#)
- [다음을 사용하여 마이그레이션하세요](#) AWS

동영상

- AWS(AWS리:인벤트 2020) [로의 대규모 마이그레이션 실행](#)
- [CloudEndure 마이그레이션 팩토리 best practices](#) (AWSre:Invent 2020)

기여자

이 전략은 전문 서비스 내 글로벌 대규모 마이그레이션 타이거 팀이 작성했습니다. AWS 이 팀은 고객을 AWS 대신하여 수천 대의 서버를 성공적으로 마이그레이션했습니다. AWS 다음은 이 문서의 기여자입니다.

- 크리스 베이커, 수석 제품 엔지니어
- 드웨인 보델론, 선임 클라우드 애플리케이션 아키텍트
- 로돌포 주니어 세라다, 선임 애플리케이션 아키텍트
- 프라틱 추나왈라, 수석 클라우드 아키텍트
- 빌 데이비드, 수석 고객 솔루션 관리자
- 데브 카, 선임 컨설턴트
- 윌리 루, 수석 컨설턴트
- 존 매디슨, 수석 클라우드 아키텍트
- 아비섹 나익, 선임 솔루션 아키텍트
- 데미안 레너, 선임 마이그레이션 전문가
- 아밋 루드라주, 선임 클라우드 아키텍트

문서 기록

다음 표에는 이 전략의 중요한 변경 사항이 설명되어 있습니다. future 업데이트에 대한 알림을 받으려면 [RSS 피드를](#) 구독하면 됩니다.

변경 사항	설명	날짜
CloudEndure 마이그레이션 서비스 제거	CloudEndure 마이그레이션 서비스에 대한 참조를 제거했습니다. AWS Application Migration Service로 lift-and-shift 마이그레이션할 때 권장되는 기본 마이그레이션AWS 클라우드 서비스입니다.	2022년 5월 11일
업데이트된AWS 솔루션 이름	참조AWS 솔루션의 이름을 마이그레이션 팩토리에서 클라우드 CloudEndure 마이그레이션 팩토리로 업데이트했습니다.	2022년 5월 2일
업데이트된 리소스	대규모 마이그레이션 시리즈의 최신 문서로 소개 및 리소스 섹션을 업데이트했습니다.	2022년 3월 8일
최초 간행물	—	2021년 9월 16일

AWS 규범적 지침 용어집

다음은 AWS 규범적 지침에서 제공하는 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 마이그레이션합니다.
- 리플랫폼(리프트 앤드 리세이프) - 애플리케이션을 클라우드로 이동하고 일정 수준의 최적화를 도입하여 클라우드 기능을 활용합니다. 예:에서 온프레미스 Oracle 데이터베이스를 Oracle용 Amazon Relational Database Service(Amazon RDS)로 마이그레이션합니다 AWS 클라우드.
- 재구매(드롭 앤드 슝) - 일반적으로 기존 라이선스에서 SaaS 모델로 전환하여 다른 제품으로 전환합니다. 예: 고객 관계 관리(CRM) 시스템을 Salesforce.com 마이그레이션합니다.
- 리호스팅(리프트 앤드 시프트) - 애플리케이션을 변경하지 않고 클라우드로 이동하여 클라우드 기능을 활용합니다. 예:의 EC2 인스턴스에서 온프레미스 Oracle 데이터베이스를 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: 마이그레이션 Microsoft Hyper-V 에 대한 애플리케이션 AWS입니다.
- 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](#) 참조하세요.

추상화된 서비스

[관리형 서비스를](#) 참조하세요.

ACID

[원자성, 일관성, 격리, 내구성](#)을 참조하세요.

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. 유연성이 뛰어나지만 [능동 수동 마이그레이션](#)보다 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 작동하고 그룹의 단일 반환 값을 계산하는 SQL 함수입니다. 집계 함수의 예로는 SUM 및 MAX가 있습니다.

AI

[인공 지능](#)을 참조하세요.

AIOps

[인공 지능 작업을](#) 참조하세요.

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

맬웨어로부터 시스템을 보호하기 위해 승인된 애플리케이션만 사용할 수 있도록 허용하는 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 작업(AIOps)

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

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, 격리, 내구성(ACID)

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

속성 기반 액세스 제어(ABAC)

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

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 절연 AWS 리전 되고 동일한 리전의 다른 가용 영역에 저렴하고 지연 시간이 짧은 네트워크 연결을 제공하는 내 고유 위치입니다.

AWS 클라우드 채택 프레임워크(AWS CAF)

조직이 클라우드로 성공적으로 전환 AWS 하기 위한 효율적이고 효과적인 계획을 개발하는 데 도움이 되는 지침 및 모범 사례 프레임워크입니다. AWS CAF는 지침을 비즈니스, 사람, 거버넌스, 플랫폼, 보안 및 운영이라는 6가지 중점 영역으로 구성합니다. 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 성공적인 클라우드 채택을 위해 조직을 준비하는 데 도움이 되는 인력 개발, 훈련 및 커뮤니케이션에 대한 지침을 제공합니다. 자세한 내용은 [AWS CAF 웹 사이트](#) 및 [AWS CAF 백서](#)를 참조하세요.

AWS 워크로드 검증 프레임워크(AWS WQF)

데이터베이스 마이그레이션 워크로드를 평가하고, 마이그레이션 전략을 권장하고, 작업 추정치를 제공하는 도구입니다. AWS WQF는 AWS Schema Conversion Tool (AWS SCT)에 포함되어 있습니다. 데이터베이스 스키마 및 코드 객체, 애플리케이션 코드, 종속성 및 성능 특성을 분석하고 평가 보고서를 제공합니다.

B

잘못된 봇

개인 또는 조직을 방해하거나 해를 입히기 위한 [봇](#)입니다.

BCP

[비즈니스 연속성 계획](#)을 참조하세요.

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. [Endianness](#)도 참조하세요.

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

별개의 동일한 두 환경을 생성하는 배포 전략입니다. 현재 애플리케이션 버전은 한 환경(파란색)에서 실행하고 새 애플리케이션 버전은 다른 환경(녹색)에서 실행합니다. 이 전략을 사용하면 영향을 최소화하면서 빠르게 롤백할 수 있습니다.

bot

인터넷을 통해 자동화된 작업을 실행하고 인적 활동 또는 상호 작용을 시뮬레이션하는 소프트웨어 애플리케이션입니다. 인터넷에서 정보를 인덱싱하는 웹 크롤러와 같은 일부 봇은 유용하거나 유용합니다. 잘못된 봇이라고 하는 다른 봇은 개인 또는 조직에 방해가 되거나 피해를 입히기 위한 것입니다.

봇넷

[맬웨어](#)에 감염되어 [있고 봇](#) 세더 또는 봇 운영자라고 하는 단일 당사자의 제어 하에 있는 봇 네트워크입니다. Botnet은 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](#)(GitHub) .

브레이크 글라스 액세스

예외적인 상황에서 승인된 프로세스를 통해 사용자가 일반적으로 액세스할 권한이 없는데 액세스할 수 있는 빠른 방법입니다. 자세한 내용은 Well-Architected 지침의 [브레이크 글라스 절차 구현](#) 표시기를 AWS 참조하세요.

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

[AWS 클라우드 채택 프레임워크](#)를 참조하세요.

canary 배포

최종 사용자에게 버전의 느린 증분 릴리스입니다. 확신이 드는 경우 새 버전을 배포하고 현재 버전을 완전히 교체합니다.

CCoE

[Cloud Center of Excellence](#)를 참조하세요.

CDC

[데이터 캡처 변경을](#) 참조하세요.

데이터 캡처 변경(CDC)

데이터베이스 테이블과 같은 데이터 소스의 변경 내용을 추적하고 변경 사항에 대한 메타데이터를 기록하는 프로세스입니다. CDC를 사용하여 대상 시스템의 변경 사항을 감사하거나 복제하여 동기화를 유지할 수 있습니다.

카오스 엔지니어링

시스템 복원력을 테스트하기 위해 의도적으로 장애 또는 중단 이벤트를 도입합니다. [AWS Fault Injection Service \(AWS FIS\)](#)를 사용하여 AWS 워크로드에 스트레스를 가하고 응답을 평가하는 실험을 수행할 수 있습니다.

CI/CD

[지속적 통합 및 지속적 전달](#)을 참조하세요.

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

대상에서 데이터를 AWS 서비스 수신하기 전에 로컬에서 데이터를 암호화합니다.

Cloud Center of Excellence(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 엔터프라이즈 전략 블로그의 [CCoE 게시물](#)을 참조하세요.

클라우드 컴퓨팅

원격 데이터 스토리지와 IoT 디바이스 관리에 일반적으로 사용되는 클라우드 기술 클라우드 컴퓨팅은 일반적으로 [엣지 컴퓨팅](#) 기술에 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 로 마이그레이션할 때 일반적으로 거치는 4단계: AWS 클라우드

- 프로젝트 - 개념 증명 및 학습 목적으로 몇 가지 클라우드 관련 프로젝트 실행
- 파운데이션 - 클라우드 채택을 확장하기 위한 기본 투자(예: 랜딩 존 생성, CCoE 정의, 운영 모델 설정)
- 마이그레이션 - 개별 애플리케이션 마이그레이션
- Re-invention - 제품 및 서비스 최적화와 클라우드 혁신

이러한 단계는 Stephen Orban이 블로그 게시물 [The Journey Toward Cloud-First 및 Enterprise Strategy 블로그의 채택 단계에](#) 정의했습니다. AWS 클라우드 AWS 마이그레이션 전략과 어떤 관련이 있는지에 대한 자세한 내용은 [마이그레이션 준비 가이드](#)를 참조하세요.

CMDB

[구성 관리 데이터베이스](#)를 참조하세요.

코드 리포지토리

소스 코드와 설명서, 샘플, 스크립트 등의 기타 자산이 버전 관리 프로세스를 통해 저장되고 업데이트되는 위치입니다. 일반적인 클라우드 리포지토리에는 다음이 포함됩니다. GitHub or Bitbucket Cloud. 코드의 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](#) 필드입니다. 예를 들어 온프레미스 카메라 네트워크에 CV를 추가하는 디바이스를 AWS Panorama 제공하고 Amazon SageMaker 는 CV에 대한 이미지 처리 알고리즘을 제공합니다.

구성 드리프트

워크로드의 경우 구성이 예상 상태에서 변경됩니다. 이로 인해 워크로드가 규정 미준수가 될 수 있으며 일반적으로 점진적이고 의도하지 않습니다.

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

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

지속적 통합 및 지속적 전달(CI/CD)

소프트웨어 릴리스 프로세스의 소스, 빌드, 테스트, 스테이징 및 프로덕션 단계를 자동화하는 프로세스입니다. CI/CD is commonly described as a pipeline. CI/CD를 사용하면 프로세스를 자동화하고, 생산성을 개선하고, 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](#) 참조하세요.

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 Framework의 보안 원칙 구성 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 거버넌스를 통해 분산되고 분산된 데이터 소유권을 제공하는 아키텍처 프레임워크입니다.

데이터 최소화

꼭 필요한 데이터만 수집하고 처리하는 원칙입니다. 에서 데이터를 최소화하면 프라이버시 위험, 비용 및 분석 탄소 발자국을 줄일 AWS 클라우드 수 있습니다.

데이터 경계

신뢰할 수 있는 자격 증명만 예상 네트워크에서 신뢰할 수 있는 리소스에 액세스할 수 있도록 하는 AWS 환경의 예방 가드레일 세트입니다. 자세한 내용은 [데이터 경계 구축을 참조하세요 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 시스템입니다. 데이터 웨어하우스에는 일반적으로 많은 양의 기록 데이터가 포함되며 일반적으로 쿼리 및 분석에 사용됩니다.

데이터베이스 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 참조하세요.

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 여러 제어를 AWS 추가하여 리소스를 보호할 수 있습니다. 예를 들어 a defense-in-depth 접근 방식은 다중 인증, 네트워크 세분화 및 암호화를 결합할 수 있습니다.

위임된 관리자

에서 AWS Organizations 호환되는 서비스는 AWS 멤버 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 Organizations 와 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 참조하세요.

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 스트림 매핑(DVSM)

소프트웨어 개발 라이프사이클에서 속도와 품질에 부정적인 영향을 미치는 제약 조건을 식별하고 우선 순위를 지정하는 데 사용되는 프로세스입니다. DVSM는 원래 린 제조 관행을 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.

차원 테이블

[스타 스키마](#)에서는 팩트 테이블의 정량적 데이터에 대한 데이터 속성을 포함하는 더 작은 테이블입니다. 차원 테이블 속성은 일반적으로 텍스트 필드 또는 텍스트처럼 동작하는 개별 숫자입니다. 이러한 속성은 쿼리 제약, 필터링 및 결과 세트 레이블 지정에 일반적으로 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)

[재해](#)로 인한 가동 중지 시간과 데이터 손실을 최소화하기 위해 사용하는 전략 및 프로세스입니다. 자세한 내용은 [AWS Well-Architected Framework의 워크로드 재해 복구](#) [AWS: 클라우드에서의 복구](#)를 참조하세요.

DML

[데이터베이스 조작 언어](#)를 참조하세요.

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 패턴으로 도메인 기반 설계를 사용하는 방법에 대한 자세한 내용은 [컨테이너 및 Amazon Word Gateway를 사용하여 레거시 Microsoft ASP.NET\(ASMX\) 웹 서비스 중분 현대화API](#) 참조하세요.

DR

[재해 복구](#)를 참조하세요.

드리프트 감지

기존 구성과의 편차 추적. 예를 들어 AWS CloudFormation 를 사용하여 [시스템 리소스의 드리프트를 감지](#)하거나 사용하여 AWS Control Tower 거버넌스 요구 사항 준수에 영향을 미칠 수 있는 [랜딩 존의 변경 사항을 감지](#)할 수 있습니다.

DVSM

[개발 값 스트림 매핑](#)을 참조하세요.

E

EDA

[탐색적 데이터 분석](#)을 참조하세요.

EDI

[전자 데이터 교환](#)을 참조하세요.

엣지 컴퓨팅

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

전자 데이터 교환(EDI)

조직 간의 비즈니스 문서 자동 교환입니다. 자세한 내용은 [Electronic Data Interchange란 무엇입니까?](#)를 참조하세요.

암호화

사람이 읽을 수 있는 일반 텍스트 데이터를 암호 텍스트로 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 엔드포인트](#)를 참조하세요.

엔드포인트 서비스

Virtual Private Cloud(VPC)에서 호스팅하여 다른 사용자와 공유할 수 있는 서비스입니다. 를 사용하여 엔드포인트 서비스를 생성하고 다른 보안 주체 AWS 계정 또는 AWS Identity and Access Management (IAM) 보안 주체에 권한을 AWS PrivateLink 부여할 수 있습니다. 이러한 계정 또는 보안 주체는 인터페이스 VPC 엔드포인트를 생성하여 엔드포인트 서비스에 비공개로 연결할 수 있습니다. 자세한 내용은 Amazon Virtual Private Cloud(Amazon VPC) 설명서의 [엔드포인트 서비스 생성을 참조하세요](#).

엔터프라이즈 리소스 계획(ERP)

엔터프라이즈의 주요 비즈니스 프로세스(예: 회계, [MES](#) 및 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 참조하세요.

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 자격 증명 및 액세스 관리, 탐지 제어, 인프라 보안, 데이터 보호 및 인시던트 대응이 포함됩니다. AWS 마이그레이션 전략의 에픽에 대한 자세한 내용은 [프로그램 구현 가이드](#)를 참조하십시오.

ERP

[엔터프라이즈 리소스 계획을 참조하세요](#).

탐색적 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

[별표 스키마](#)의 중앙 테이블입니다. 비즈니스 운영에 대한 정량적 데이터를 저장합니다. 일반적으로 팩트 테이블에는 측정값이 포함된 열과 차원 테이블에 대한 외래 키가 포함된 열의 두 가지 유형이 있습니다.

빠른 실패

자주 증분 테스트를 사용하여 개발 수명 주기를 줄이는 철학입니다. 애자일 접근 방식의 중요한 부분입니다.

장애 격리 경계

에서 장애의 영향을 제한하고 워크로드의 복원력을 개선하는 데 도움이 되는 가용 영역, AWS 리전 제어 영역 또는 데이터 영역과 같은 AWS 클라우드경계입니다. 자세한 내용은 [AWS 장애 격리 경계를 참조하세요](#).

기능 브랜치

[브랜치를 참조하세요](#).

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 Shapley Additive Explanations(SHAP) 및 통합 그라데이션과 같은 다양한 기법을 통해 계산할 수 있는 수치 점수로 표현됩니다. 자세한 내용은 [를 사용한 기계 학습 모델 해석 가능성을 참조하세요AWS](#).

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 활용

할 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.

몇 번의 샷 프롬프트

유사한 작업을 수행하도록 요청하기 전에 작업과 원하는 출력을 보여주는 몇 가지 예를 [LLM](#)에 제공합니다. 이 기법은 컨텍스트 내 학습을 적용하여 모델이 프롬프트에 포함된 예제(샷)에서 학습합니다. 샷 프롬프트는 특정 형식 지정, 추론 또는 도메인 지식이 필요한 작업에 효과적일 수 있습니다. [제로샷 프롬프트도 참조하세요.](#)

FGAC

[세분화된 액세스 제어를 참조하세요.](#)

세분화된 액세스 제어(FGAC)

여러 조건을 사용하여 액세스 요청을 허용하거나 거부합니다.

플래시컷 마이그레이션

단계적 접근 방식을 사용하는 대신 [변경 데이터 캡처](#)를 통해 지속적인 데이터 복제를 사용하여 가능한 최단 시간 내에 데이터를 마이그레이션하는 데이터베이스 마이그레이션 방법입니다. 목표는 가동 중지 시간을 최소화하는 것입니다.

FM

[파운데이션 모델을 참조하세요.](#)

파운데이션 모델(FM)

일반화된 데이터와 레이블이 지정되지 않은 데이터의 대규모 데이터 세트에 대해 훈련된 대규모 딥러닝 신경망입니다. FMs는 언어 이해, 텍스트 및 이미지 생성, 자연어 대화와 같은 다양한 일반 작업을 수행할 수 있습니다. 자세한 내용은 [파운데이션 모델이란 무엇입니까?](#)를 참조하세요.

G

생성형 AI

대량의 데이터에 대해 훈련되었으며 간단한 텍스트 프롬프트를 사용하여 이미지, 비디오, 텍스트 및 오디오와 같은 새 콘텐츠 및 아티팩트를 생성할 수 있는 [AI](#) 모델의 하위 집합입니다. 자세한 내용은 [생성형 AI란 무엇입니까?](#)를 참조하세요.

지리적 차단

[지리적 제한을 참조하세요.](#)

지리적 제한(지리적 차단)

Amazon CloudFront에서는 특정 국가의 사용자가 콘텐츠 배포에 액세스하지 못하도록 하는 옵션입니다. 허용 목록 또는 차단 목록을 사용하여 승인된 국가와 차단된 국가를 지정할 수 있습니다. 자세한 내용은 CloudFront 설명서 [의 콘텐츠의 지리적 배포 제한을 참조하세요](#).

Gitflow 워크플로

하위 환경과 상위 환경이 소스 코드 리포지토리의 서로 다른 브랜치를 사용하는 방식입니다. Gitflow 워크플로는 레거시로 간주되며 [트렁크 기반 워크플로](#)는 현대적이고 선호하는 접근 방식입니다.

골든 이미지

해당 시스템 또는 소프트웨어의 새 인스턴스를 배포하기 위한 템플릿으로 사용되는 시스템 또는 소프트웨어의 스냅샷입니다. 예를 들어 제조에서 황금색 이미지를 사용하여 여러 디바이스에 소프트웨어를 프로비저닝할 수 있으며 디바이스 제조 작업의 속도, 확장성 및 생산성을 개선하는 데 도움이 됩니다.

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(OUs) 전반의 리소스, 정책 및 규정 준수를 관리하는 데 도움이 되는 상위 수준 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, 이는 AWS Config, Amazon GuardDuty AWS Security Hub, , Amazon AWS Trusted Advisor Amazon Inspector 및 사용자 지정 AWS Lambda 검사를 사용하여 구현됩니다.

H

HA

[고가용성을 참조하세요](#).

이기종 데이터베이스 마이그레이션

다른 데이터베이스 엔진을 사용하는 대상 데이터베이스로 소스 데이터베이스 마이그레이션(예: Oracle에서 Amazon Aurora로) 이기종 마이그레이션은 일반적으로 리아키텍트 작업의 일부이며 스

키마를 변환하는 것은 복잡한 작업일 수 있습니다. AWS 는 스키마 변환에 도움이 되는 [AWS SCT를](#) [제공](#)합니다.

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.

홀드아웃 데이터

[기계 학습](#) 모델을 훈련하는 데 사용되는 데이터 세트에서 보류된 레이블이 지정된 기록 데이터의 일부입니다. 홀드아웃 데이터를 사용하여 모델 예측을 홀드아웃 데이터와 비교하여 모델 성능을 평가할 수 있습니다.

동종 데이터베이스 마이그레이션

소스 데이터베이스를 동일한 데이터베이스 엔진을 공유하는 대상 데이터베이스로 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. 긴급성으로 인해 핫픽스는 일반적으로 typical DevOps 릴리스 워크플로 외부에서 이루어집니다.

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](#) 참조하세요.

자격 증명 기반 정책

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

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 인터넷](#)을 참조하세요.

변경할 수 없는 인프라

기존 인프라를 업데이트, 패치 또는 수정하는 대신 프로덕션 워크로드를 위해 새 인프라를 배포하는 모델입니다. 변경 가능한 인프라는 본질적으로 [변경 가능한 인프라](#)보다 더 일관되고 안정적이며 예측 가능합니다. 자세한 내용은 AWS Well-Architected Framework의 [변경할 수 없는 인프라를 사용한 배포](#) 모범 사례를 참조하세요.

인바운드(수신) VPC

AWS 다중 계정 아키텍처에서 애플리케이션 외부에서 네트워크 연결을 수락, 검사 및 라우팅하는 VPC입니다. [AWS 보안 참조 아키텍처](#)는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPCs로 네트워크 계정을 설정하는 것이 좋습니다.

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 것이 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

연결성, 실시간 데이터, 자동화, 분석 및 AI/ML의 발전을 통한 제조 프로세스의 현대화를 언급하기 위해 2016년에 [Klaus Schwab](#)에서 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\(IIoT\) 디지털 혁신 전략 구축을](#) 참조하세요.

검사 VPC

AWS 다중 계정 아키텍처에서는 VPC(동일하거나 다른 AWS 리전), 인터넷 및 온프레미스 네트워크 간의 네트워크 트래픽 검사를 관리하는 중앙 집중식 VPCs입니다. [AWS 보안 참조 아키텍처](#)는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPCs로 네트워크 계정을 설정하는 것이 좋습니다.

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 [AWS를 사용한 기계 학습 모델 해석 가능성을](#) 참조하세요.

IoT

[사물 인터넷을](#) 참조하세요.

IT 정보 라이브러리(ITIL)

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

IT 서비스 관리(ITSM)

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

ITIL

[IT 정보 라이브러리](#)를 참조하세요.

ITSM

[IT 서비스 관리](#)를 참조하세요.

L

레이블 기반 액세스 제어(LBAC)

사용자와 데이터 자체에 각각 보안 레이블 값이 명시적으로 할당되는 필수 액세스 제어(MAC) 구현입니다. 사용자 보안 레이블과 데이터 보안 레이블 간의 교차 부분에 따라 사용자가 볼 수 있는 행과 열이 결정됩니다.

랜딩 존

랜딩 존은 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.

대형 언어 모델(LLM)

방대한 양의 데이터에 대해 사전 훈련된 딥 러닝 [AI](#) 모델입니다. LLM는 질문 답변, 문서 요약, 텍스트를 다른 언어로 번역, 문장 완성과 같은 여러 작업을 수행할 수 있습니다. 자세한 내용은 [LLMs란 무엇입니까?](#)를 참조하세요.

대규모 마이그레이션

300대 이상의 서버 마이그레이션입니다.

LBAC

[레이블 기반 액세스 제어](#)를 참조하세요.

최소 권한

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

리프트 앤드 시프트

[7 Rs](#)를 참조하세요.

리틀 엔디안 시스템

가장 덜 중요한 바이트를 먼저 저장하는 시스템입니다. [Endianness](#)도 참조하세요.

LLM

[큰 언어 모델을](#) 참조하세요.

하위 환경

[환경을](#) 참조하세요.

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

[브랜치를](#) 참조하세요.

맬웨어

컴퓨터 보안 또는 프라이버시를 손상시키도록 설계된 소프트웨어입니다. 맬웨어는 컴퓨터 시스템을 중단하거나 민감한 정보를 유출하거나 무단 액세스를 가져올 수 있습니다. 맬웨어의 예로는 바이러스, 웜, 랜섬웨어, 트로이 목마, 스파이웨어, 키로거 등이 있습니다.

관리형 서비스

AWS 서비스는 인프라 계층, 운영 체제 및 플랫폼을 AWS 작동하며 엔드포인트에 액세스하여 데이터를 저장하고 검색합니다. Amazon Simple Storage Service(Amazon S3) 및 Amazon DynamoDB는 관리형 서비스의 예입니다. 이를 추상화된 서비스라고도 합니다.

제조 실행 시스템(MES)

원재료를 생산 현장의 완제품으로 변환하는 생산 프로세스를 추적, 모니터링, 문서화 및 제어하기 위한 소프트웨어 시스템입니다.

MAP

[마이그레이션 가속화 프로그램을](#) 참조하세요.

메커니즘

도구를 생성하고 도구 채택을 유도한 다음 결과를 검사하여 조정하는 전체 프로세스입니다. 메커니즘은 작동 시 자체를 강화하고 개선하는 주기입니다. 자세한 내용은 AWS Well-Architected Framework의 [메커니즘 구축](#)을 참조하세요.

멤버 계정

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

MES

[제조 실행 시스템을](#) 참조하세요.

메시지 대기열 원격 측정 전송(MQTT)

리소스가 제한된 IoT 디바이스에 대한 [게시/구독](#) 패턴을 기반으로 하는 경량 machine-to-machine(M2M) 통신 프로토콜입니다.

마이크로서비스

잘 정의된 APIs를 통해 통신하고 일반적으로 소규모 독립 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 [AWS 서버리스 서비스를 사용하여 마이크로서비스 통합을 참조하세요](#).

마이크로서비스 아키텍처

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

마이그레이션 가속화 프로그램(MAP)

조직이 클라우드로 전환하기 위한 강력한 운영 기반을 구축하고 초기 마이그레이션 비용을 상쇄하는 데 도움이 되는 컨설팅 지원, 훈련 및 서비스를 제공하는 AWS 프로그램입니다. MAP에는 체계적인 방식으로 레거시 마이그레이션을 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 스프린트에서 작업하는 운영, 비즈니스 분석가 및 소유자, 마이그레이션 엔지니어, 개발자 및 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 및 AWS 계정이 있습니다.

마이그레이션 패턴

사용되는 마이그레이션 전략, 마이그레이션 대상, 마이그레이션 애플리케이션 또는 서비스를 자세히 설명하는 반복 가능한 마이그레이션 작업입니다. 예: AWS Application Migration Service를 사용하여 Amazon EC2로 마이그레이션을 다시 호스팅합니다.

마이그레이션 포트폴리오 평가(MPA)

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

마이그레이션 준비 평가(MRA)

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

마이그레이션 전략

워크로드를 로 마이그레이션하는 데 사용되는 접근 방식입니다 AWS 클라우드. 자세한 내용은 이 용어집의 [7 Rs](#) 항목을 참조하고 [대규모 마이그레이션을 가속화하기 위해 조직 동원을 참조하세요.](#)

ML

[기계 학습](#)을 참조하세요.

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 애플리케이션 현대화 전략을 참조하세요 AWS 클라우드.](#)

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 참조하세요 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 평가](#)를 참조하세요.

MQTT

[메시지 대기열 원격 측정 전송](#)을 참조하세요.

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 워크로드의 기존 인프라를 업데이트하고 수정하는 모델입니다. 일관성, 신뢰성 및 예측 가능성을 높이기 위해 AWS Well-Architected Framework는 [변경할 수 없는 인프라](#)를 모범 사례로 사용할 것을 권장합니다.

O

OAC

[오리진 액세스 제어를](#) 참조하세요.

OAI

[오리진 액세스 자격 증명을](#) 참조하세요.

OCM

[조직 변경 관리를](#) 참조하세요.

오프라인 마이그레이션

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

OI

[작업 통합](#)을 참조하세요.

OLA

[운영 수준 계약을](#) 참조하세요.

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

[Open Process Communications - Unified Architecture](#)를 참조하세요.

Open Process Communications - 통합 아키텍처(OPC-UA)

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

운영 수준 계약(OLA)

서비스 수준 계약(SLA)을 지원하기 위해 기능 IT 그룹이 서로에게 제공할 것을 명확히 하는 계약입니다.

운영 준비 검토(ORR)

인시던트 및 가능한 장애의 범위를 이해, 평가, 예방 또는 줄이는 데 도움이 되는 질문 및 관련 모범 사례 체크리스트입니다. 자세한 내용은 AWS Well-Architected Framework의 [운영 준비 검토\(ORR\)](#)를 참조하세요.

운영 기술(OT)

물리적 환경과 협력하여 산업 운영, 장비 및 인프라를 제어하는 하드웨어 및 소프트웨어 시스템입니다. 제조업에서 OT와 정보 기술(IT) 시스템의 통합은 [Industry 4.0](#) 혁신의 주요 초점입니다.

운영 통합(OI)

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

조직 트레일

에서 생성한 추적은의 조직 AWS 계정 내 모든에 대한 모든 이벤트를 AWS CloudTrail 기록합니다 AWS Organizations. 이 트레일은 조직에 속한 각 AWS 계정에 생성되고 각 계정의 활동을 추적합니다. 자세한 내용은 CloudTrail 설명서의 [조직에 대한 추적 생성](#)을 참조하세요.

조직 변경 관리(OCM)

사람, 문화 및 리더십 관점에서 중대하고 파괴적인 비즈니스 혁신을 관리하기 위한 프레임워크입니다. OCM는 변화 채택을 가속화하고, 전환 문제를 해결하고, 문화적 및 조직적 변화를 주도하여 조직이 새로운 시스템 및 전략을 준비하고 전환할 수 있도록 지원합니다. AWS 마이그레이션 전략에서는 클라우드 채택 프로젝트에 필요한 변경 속도 때문에이 프레임워크를 인력 가속화라고 합니다. 자세한 내용은 [OCM 가이드](#)를 참조하세요.

오리진 액세스 제어(OAC)

In CloudFront, Amazon Simple Storage Service(Amazon S3) 콘텐츠를 보호하기 위한 액세스를 제한하는 향상된 옵션입니다. OAC는 AWS KMS (SSE-KMS)를 사용한 모든 AWS 리전서버 측 암호화, 동적 PUT 및 S3 버킷에 대한 DELETE 요청에서 모든 S3 버킷을 지원합니다.

오리진 액세스 자격 증명(OAI)

In CloudFront, Amazon S3 콘텐츠를 보호하기 위해 액세스를 제한하는 옵션입니다. OAI를 사용하면 CloudFront 는 Amazon S3가 인증할 수 있는 보안 주체를 생성합니다. 인증된 보안 주체는 특정

CloudFront 배포를 통해서만 S3 버킷의 콘텐츠에 액세스할 수 있습니다. 보다 세분화되고 향상된 액세스 제어를 제공하는 [OAC](#)도 참조하세요.

ORR

[운영 준비 상태 검토](#)를 참조하세요.

OT

[운영 기술을](#) 참조하세요.

아웃바운드(송신) VPC

AWS 다중 계정 아키텍처에서 애플리케이션 내에서 시작된 네트워크 연결을 처리하는 VPC입니다. [AWS 보안 참조 아키텍처](#)는 애플리케이션과 더 넓은 인터넷 간의 양방향 인터페이스를 보호하기 위해 인바운드, 아웃바운드 및 검사 VPCs로 네트워크 계정을 설정하는 것이 좋습니다.

P

권한 경계

사용자 또는 역할이 가질 수 있는 최대 권한을 설정하기 위해 IAM 보안 주체에 연결된 IAM 관리 정책입니다. 자세한 내용은 IAM 설명서의 [권한 경계](#)를 참조하세요.

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 및 연락처 정보가 있습니다.

PII

[개인 식별 정보를](#) 참조하세요.

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

[프로그래밍 가능한 로직 컨트롤러](#)를 참조하세요.

PLM

[제품 수명 주기 관리](#)를 참조하세요.

정책

권한을 정의하거나([자격 증명 기반 정책](#) 참조), 액세스 조건을 지정하거나([리소스 기반 정책](#) 참조), 조직의 모든 계정에 대한 최대 권한을 정의할 수 있는 객체입니다 AWS Organizations ([서비스 제어 정책](#) 참조).

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

WHERE 절에서 false 일반적으로 위치한 true 또는를 반환하는 쿼리 조건입니다.

조건부 푸시다운

전송 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 엔터티입니다. 이 엔터티는 일반적으로 AWS 계정, IAM 역할 또는 사용자의 루트 사용자입니다. 자세한 내용은 IAM 설명서의 역할의 보안 주체 용어 및 개념을 참조하세요. https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html#id_roles_terms-and-concepts

프라이버시 중심 설계

전체 개발 프로세스를 통해 프라이버시를 고려하는 시스템 엔지니어링 접근 방식입니다.

프라이빗 호스팅 영역

Amazon Route 53이 하나 이상의 DNS 내에서 도메인 및 하위 도메인에 대한 VPCs 쿼리에 응답하는 방법에 대한 정보를 포함하는 컨테이너입니다. 자세한 내용은 Route 53 설명서의 [프라이빗 호스팅 영역 작업](#)을 참조하십시오.

사전 예방적 제어

규정 미준수 리소스의 배포를 방지하도록 설계된 [보안 제어](#)입니다. 이러한 제어는 리소스가 프로비저닝되기 전에 스캔 리소스를 스캔합니다. 리소스가 컨트롤을 준수하지 않으면 프로비저닝되지 않습니다. 자세한 내용은 AWS Control Tower 설명서의 [제어 참조 가이드](#)를 참조하고의 보안 [제어 구현의 사전](#) 예방적 제어를 참조하세요. AWS

제품 수명 주기 관리(PLM)

설계, 개발 및 출시부터 성장 및 성숙도, 거부 및 제거에 이르기까지 전체 수명 주기 동안 제품의 데이터 및 프로세스 관리.

프로덕션 환경

[환경](#)을 참조하세요.

프로그래밍 가능한 로직 컨트롤러(PLC)

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

프롬프트 체인

한 [LLM](#) 프롬프트의 출력을 다음 프롬프트의 입력으로 사용하여 더 나은 응답을 생성합니다. 이 기법은 복잡한 작업을 하위 작업으로 세분화하거나 예비 응답을 반복적으로 개선하거나 확장하는 데 사용됩니다. 이는 모델 응답의 정확성과 관련성을 개선하고 보다 세분화되고 개인화된 결과를 가능하게 합니다.

가명화

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

publish/subscribe (pub/sub)

마이크로서비스 간의 비동기 통신을 지원하여 확장성과 응답성을 개선하는 패턴입니다. 예를 들어 마이크로서비스 기반 [MES](#)에서 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 지침과 같은 일련의 단계입니다.

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 행렬

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

RAG

[증강된 생성 검색](#)을 참조하세요.

랜섬웨어

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

RASCI 행렬

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

RCAC

[행 및 열 액세스 제어](#)를 참조하세요.

읽기 전용 복제본

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

재설계

[7 Rs](#)를 참조하세요.

복구 시점 목표(RPO)

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

복구 시간 목표(RTO)

서비스 중단과 서비스 복원 사이의 허용 가능한 지연 시간입니다.

리팩터

[7 Rs](#)를 참조하세요.

리전

지리적 영역의 AWS 리소스 모음입니다. 각 AWS 리전은 내결함성, 안정성 및 복원력을 제공하기 위해 서로 격리되고 독립적입니다. 자세한 내용은 [계정에서 사용할 수 있는 항목 지정을 참조 AWS 리전 하세요.](#)

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 Rs](#)를 참조하세요.

release

배포 프로세스에서 변경 사항을 프로덕션 환경으로 승격시키는 행위입니다.

재배치

[7 Rs](#)를 참조하세요.

리플랫폼

[7 Rs](#)를 참조하세요.

재구매

[7 Rs](#)를 참조하세요.

복원력

중단에 저항하거나 복구할 수 있는 애플리케이션의 기능입니다. 에서 복원력을 계획할 때 [고가용성](#) 및 [재해 복구](#)는 일반적인 고려 사항입니다 AWS 클라우드. 자세한 내용은 [AWS 클라우드 복원력을 참조하세요.](#)

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.

책임, 책임, 상담, 정보(RACI) 매트릭스

마이그레이션 활동 및 클라우드 운영에 참여하는 모든 당사자의 역할과 책임을 정의하는 매트릭스입니다. 매트릭스 이름은 매트릭스에 정의된 책임 유형에서 파생됩니다. 실무 담당자 (R), 의사 결정권자 (A), 업무 수행 조연자 (C), 결과 통보 대상자 (I). 지원자는 (S) 선택사항입니다. 지원을 포함하는 경우 매트릭스를 RASCI 매트릭스라고 하고, 이를 제외하는 경우 이를 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 Rs](#)를 참조하세요.

사용 중지

[7 Rs](#)를 참조하세요.

검색 증강 세대(RAG)

[LLM](#)가 응답을 생성하기 전에 훈련 데이터 소스 외부에 있는 신뢰할 수 있는 데이터 소스를 참조하는 [생성형 AI](#) 기술입니다. 예를 들어 Word RAG 모델은 조직의 지식 기반 또는 사용자 지정 데이터에 대한 의미 검색을 수행할 수 있습니다. 자세한 내용은 [RAG란 무엇입니까?](#)를 참조하세요.

교체

공격자가 보안 인증 정보에 액세스하는 것을 더 어렵게 만들기 위해 [보안 암호](#)를 주기적으로 업데이트하는 프로세스입니다.

행 및 열 액세스 제어(RCAC)

액세스 규칙을 정의한 기본적이고 유연한 SQL 표현식 사용. RCAC는 행 권한과 열 마스크로 구성됩니다.

RPO

[복구 시점 목표](#)를 참조하세요.

RTO

[복구 시간 목표를](#) 참조하세요.

런북

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

S

SAML 2.0

많은 자격 증명 공급자(IdPs)가 사용하는 개방형 표준입니다. 이 기능을 사용하면 페더레이션 Single Sign-On(SSO)을 사용할 수 있으므로 사용자는 조직의 모든 사용자를 위해 IAM에서 사용자를 생성할 필요 없이 AWS Management Console 에 로그인하거나 AWS API 작업을 호출할 수 있습니다. SAML 2.0 기반 페더레이션에 대한 자세한 내용은 [SAML 설명서의 Word 2.0 기반 페더레이션 정보를](#) 참조하세요. IAM

SCADA

[감독 제어 및 데이터 수집](#)을 참조하세요.

SCP

[서비스 제어 정책을](#) 참조하세요.

secret

에는 암호 또는 사용자 자격 증명과 같이 암호화된 형식으로 저장하는 AWS Secrets Manager 기밀 또는 제한된 정보가 있습니다. 보안 암호 값과 메타데이터로 구성됩니다. 보안 암호 값은 바이너리, 단일 문자열 또는 여러 문자열일 수 있습니다. 자세한 내용은 [Secrets Manager 설명서의 Secrets Manager 보안 암호의 내용을](#) 참조하세요.

설계별 보안

전체 개발 프로세스를 통해 보안을 고려하는 시스템 엔지니어링 접근 방식입니다.

보안 제어

위험 행위자가 보안 취약성을 악용하는 능력을 방지, 탐지 또는 감소시키는 기술적 또는 관리적 가드레일입니다. 보안 제어에는 [예방](#), [탐지](#), [대응](#) 및 [사전](#) 예방이라는 네 가지 주요 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) 및 보안 이벤트 관리(SEM) 시스템을 결합하는 도구 및 서비스입니다. SIEM 시스템은 서버, 네트워크, 디바이스 및 기타 소스에서 데이터를 수집, 모니터링 및 분석하여 위협 및 보안 위반을 탐지하고 알림을 생성합니다.

보안 응답 자동화

보안 이벤트에 자동으로 응답하거나 이를 해결하도록 설계된 사전 정의되고 프로그래밍된 작업입니다. 이러한 자동화는 보안 모범 사례를 구현하는 데 도움이 되는 [탐지적](#) 또는 [대응적](#) AWS 보안 제어 역할을 합니다. 자동 응답 작업의 예로는 VPC 보안 그룹 수정, Amazon EC2 인스턴스 패치 적용 또는 보안 인증 정보 교체 등이 있습니다.

서버 측 암호화

대상에서 데이터를 AWS 서비스 수신하는에 의한 데이터 암호화.

서비스 제어 정책(SCP)

AWS Organizations에 속한 조직의 모든 계정에 대한 권한을 중앙 집중식으로 제어하는 정책입니다. SCPs는 가드레일을 정의하거나 관리자가 사용자 또는 역할에 위임할 수 있는 작업에 대한 제한을 설정합니다. SCPs를 허용 목록 또는 거부 목록으로 사용하여 허용되거나 금지된 서비스 또는 작업을 지정할 수 있습니다. 자세한 내용은 AWS Organizations 설명서의 [서비스 제어 정책을](#) 참조하세요.

서비스 엔드포인트

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

서비스 수준 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 측면 측정입니다.

서비스 수준 목표(SLO)

서비스 [수준 지표](#)로 측정된 서비스 상태를 나타내는 대상 지표입니다.

공동 책임 모델

클라우드 보안 및 규정 준수에 AWS 대해 공유하는 책임을 설명하는 모델. AWS 는 클라우드의 보안을 책임지고, 는 클라우드의 보안을 책임집니다. 자세한 내용은 [공동 책임 모델](#)을 참조하십시오.

SIEM

[보안 정보 및 이벤트 관리 시스템](#)을 참조하세요.

단일 장애 지점(SPOF)

시스템을 중단시킬 수 있는 애플리케이션의 중요한 단일 구성 요소의 장애입니다.

SLA

[서비스 수준 계약을 참조하세요.](#)

SLI

[서비스 수준 표시기를 참조하세요.](#)

SLO

[서비스 수준 목표를 참조하세요.](#)

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 [에서 애플리케이션 현대화에 대한 단계별 접근 방식을 참조하세요 AWS 클라우드.](#)

SPOF

[단일 장애 지점을 참조하세요.](#)

스타 스키마

하나의 큰 팩트 테이블을 사용하여 트랜잭션 또는 측정된 데이터를 저장하고 하나 이상의 작은 차원 테이블을 사용하여 데이터 속성을 저장하는 데이터베이스 조직 구조입니다. 이 구조는 [데이터 웨어하우스](#) 또는 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 ASP API Gateway를 사용하여 레거시 Microsoft Word.NET\(ASMX\) 웹 서비스 중분 현대화를 참조하세요.](#)

서브넷

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

감독 제어 및 데이터 수집(SCADA)

제조에서 하드웨어와 소프트웨어를 사용하여 물리적 자산과 생산 작업을 모니터링하는 시스템입니다.

대칭 암호화

동일한 키를 사용하여 데이터를 암호화하고 복호화하는 암호화 알고리즘입니다.

합성 테스트

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

시스템 프롬프트

Word의 동작을 지시하기 위해 [LLM](#)에 컨텍스트, 지침 또는 지침을 제공하는 기법입니다. 시스템 프롬프트는 컨텍스트를 설정하고 사용자와의 상호 작용 규칙을 설정하는 데 도움이 됩니다.

T

tags

AWS 리소스를 구성하기 위한 메타데이터 역할을 하는 키-값 페어입니다. 태그를 사용하면 리소스를 손쉽게 관리, 식별, 정리, 검색 및 필터링할 수 있습니다. 자세한 내용은 [AWS 리소스에 태그 지정을 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](#) 참조하세요.

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

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

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

사용자를 대신하여 AWS Organizations 및 해당 계정에서 조직에서 작업을 수행하도록 지정한 서비스에 권한 부여. 신뢰할 수 있는 서비스는 필요할 때 각 계정에 서비스 연결 역할을 생성하여 관리 작업을 수행합니다. 자세한 내용은 설명서의 [다른 AWS 서비스와 AWS Organizations 함께 사용](#)을 참조하세요 AWS Organizations .

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

두 개의 피자로 피드할 수 있는 small DevOps 팀입니다. 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

[환경](#)을 참조하세요.

V

정리

스토리지를 회수하고 성능을 향상시키기 위해 증분 업데이트 후 정리 작업을 수행하는 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 있는 두 VPCs 간의 연결입니다. 자세한 내용은 Amazon [VPC 설명서의 Word 피어링이란 무엇입니까](#) VPC를 참조하세요.

취약성

시스템 보안을 손상시키는 소프트웨어 또는 하드웨어 결함입니다.

W

웜 캐시

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

웜 데이터

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

창 함수

현재 레코드와 어떤 식으로든 관련된 행 그룹에 대해 계산을 수행하는 SQL 함수입니다. 창 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.

WORM

[쓰기를 한 번 보고, 많이 읽습니다.](#)

WQF

[AWS 워크로드 검증 프레임워크를 참조하세요.](#)

한 번 쓰기, 많이 읽기(WORM)

데이터를 한 번에 쓰고 데이터가 삭제되거나 수정되지 않도록 하는 스토리지 모델입니다. 권한 있는 사용자는 필요한 만큼 데이터를 읽을 수 있지만 변경할 수는 없습니다. 이 데이터 스토리지 인프라는 [변경할 수 없는](#) 것으로 간주됩니다.

Z

제로데이 익스플로잇

[제로데이 취약성](#)을 활용하는 공격, 일반적으로 맬웨어입니다.

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.

제로샷 프롬프트

[LLM](#)에 작업 수행에 대한 지침을 제공하지만, 작업 가이드에 도움이 될 수 있는 예제(샷)는 제공하지 않습니다. LLM는 사전 훈련된 지식을 사용하여 작업을 처리해야 합니다. 제로샷 프롬프트의 효과는 작업의 복잡성과 프롬프트의 품질에 따라 달라집니다. 또한 [몇 번의 샷 프롬프트를 참조하세요](#).

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.