AWS에서 워크로드의 재해 복구: 클라우드에서의 복구

AWS 백서

AWS에서 워크로드의 재해 복구: 클라우드에서의 복구: AWS 백서

Table of Contents

AWS에서 워크로드의 재해 복구: 클라우드에서의 복구

게시 날짜: 2021년 2월 12일(문서 개정 (p. 27))

요약

재해 복구는 재해에 대비하고 재해를 복구하는 프로세스입니다. 워크로드 또는 시스템이 기본 배포 위치에서 비즈니스 목표를 달성하지 못하게 하는 이벤트는 재해로 간주됩니다. 이 백서에서는 AWS에 배포된 모든 워크로드에 대한 재해 복구를 계획하고 테스트하기 위한 모범 사례를 간략하게 설명하고, 위험을 완화하고 해당 워크로드에 대한 복구 시간 목표(RTO) 및 복구 시점 목표(RPO)를 충족하는 다양한 방식을 제공합니다.

소개

워크로드는 의도한 기능을 정확하고 일관되게 수행해야 합니다. 이를 위해서는 복원성을 고려한 설계가 필요합니다. 복원성에는 인프라나 서비스의 시스템 장애를 복구하고, 컴퓨팅 리소스를 동적으로 확보하여 수요에 대응하거나, 구성 오류나 일시적 네트워크 문제와 같은 장애를 완화하는 워크로드의 기능이 포함됩니다.

재해 복구(DR)는 복원성 전략의 중요한 부분이며 재해가 발생했을 때 워크로드가 어떻게 대응하는지에 관한 것입니다. 재해 (p. 6)는 비즈니스에 심각한 부정적인 영향을 미치는 이벤트입니다. 이러한 대응은 데이터 손실을 방지하고(복구 시점 목표(RPO) (p. 8)) 워크로드를 사용할 수 없는 가동 중지 시간을 줄이기(복구 시간 목표(RTO) (p. 8)) 위한 워크로드 전략을 지정하는 조직의 비즈니스 목표를 기반으로 해야 합니다. 따라서 특정한 일회성 재해 이벤트에 대한 복구 목표(RPO 및 RTO (p. 8))를 충족하기 위해 클라우드의 워크로드 설계에서 복원성을 구현해야 합니다. 이 방식은 조직이 비즈니스 연속성 계획(BCP) (p. 8)의 일부로 비즈니스 연속성을 유지하는 데 도움이 됩니다.

이 백서에서는 비즈니스의 재해 복구 목표를 충족하는 AWS 기반 아키텍처를 계획, 설계 및 구현하는 방법을 중점적으로 다룹니다. 이 백서의 정보는 CTO, 아키텍트, 개발자, 운영 팀 팀원 등 기술 업무 담당자를 위해 개발되었습니다.

재해 복구 및 가용성

재해 복구는 복원성 전략의 또 다른 중요한 구성 요소인 가용성과 비교할 수 있습니다. 재해 복구는 일회성 이벤트의 목표를 측정하는 반면 가용성 목표는 일정 기간 동안의 평균값을 측정합니다.

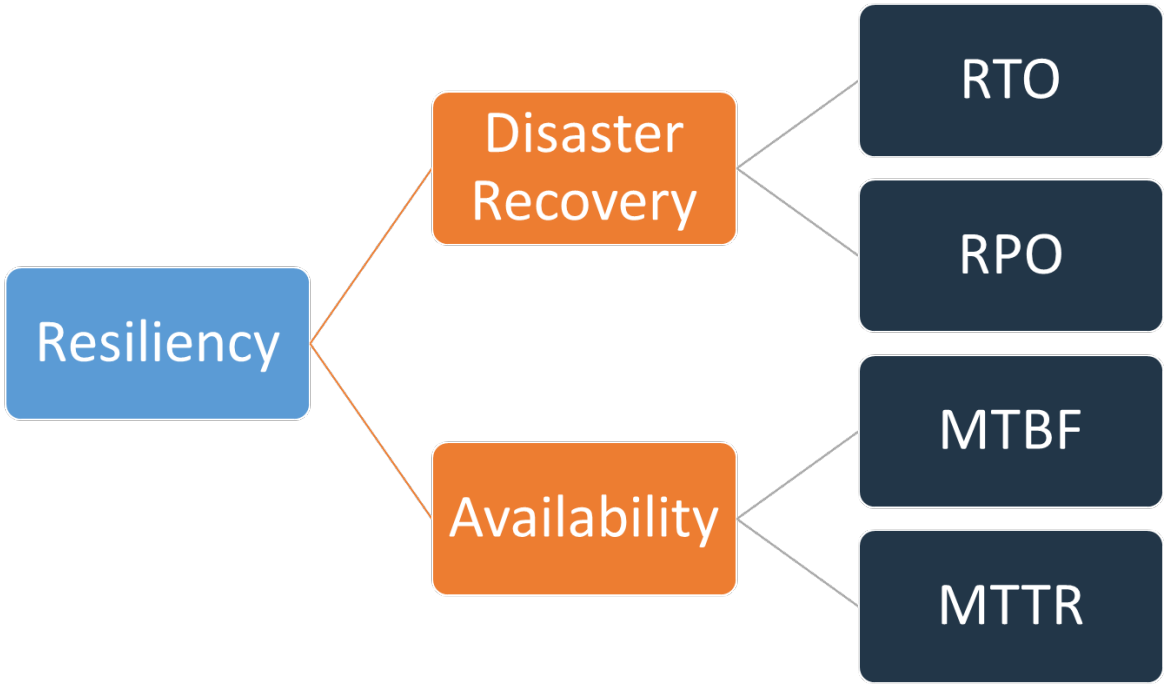

그림 1 - 복원성 목표

가용성은 MTBF와 평균 복구 시간(MTTR)을 사용하여 계산됩니다.

$$Availability = \frac{Available\ for\ Use\ Time}{Total\ Time} = \frac{MTBF}{MTBF + MTTR}$$

이 방식을 흔히 '나인'이라고 하며, 99.9%의 가용성 목표를 '쓰리 나인'이라고 합니다.

워크로드의 경우 시간 기반 방식을 사용하는 대신 성공 및 실패한 요청을 계산하는 것이 더 쉬울 수 있습니다. 이 경우 다음 계산을 사용할 수 있습니다.

$$Availability = \frac{Successful\ Responses}{Valid\ Requests}$$

재해 복구는 재해 이벤트에 초점을 맞추는 반면, 가용성은 구성 요소 장애, 네트워크 문제 및 부하 급증과 같은 소규모의 보다 일반적인 중단에 중점을 둡니다. 재해 복구의 목적은 비즈니스 연속성이지만 가용성은 의도한 비즈니스 기능을 수행하는 데 워크로드를 사용할 수 있는 시간을 최대화하는 것과 관련이 있습니다. 둘 다 복원성 전략에 포함되어야 합니다.

복원성을 위한 공동 책임 모델

복원성은 AWS와 고객 간의 공동 책임입니다. 복원성의 일부로서 재해 복구 및 가용성이 이 공유 모델에서 어떻게 작동하는지 이해하는 것이 중요합니다.

AWS의 책임 '클라우드의 복원성'

AWS는 AWS 클라우드에서 제공하는 모든 서비스가 실행되는 인프라의 복원성을 제공할 책임이 있습니다. 이 인프라는 AWS 클라우드 서비스를 실행하는 하드웨어, 소프트웨어, 네트워킹 및 시설로 구성됩니다. AWS는 이러한 AWS 클라우드 서비스를 제공하기 위해 상업적으로 합당한 노력을 기울여 서비스 가용성이 AWS 서비스 수준 계약(SLA)을 충족하거나 초과할 수 있도록 합니다.

AWS 글로벌 클라우드 인프라는 고객이 복원성이 뛰어난 워크로드 아키텍처를 구축할 수 있도록 설계되었습니다. 각 AWS 리전은 완전히 격리되어 있으며 물리적으로 격리된 인프라 파티션인 가용 영역 여러 개로 구성됩니다. 가용 영역은 워크로드 복원성에 영향을 줄 수 있는 결함을 격리하여 리전의 다른 영역에 영향을 주지 않도록 합니다. AWS 리전의 모든 가용 영역은 높은 대역폭, 대기 시간이 짧은 네트워킹, 완전한 중복성을 갖춘 전용 메트로 광 네트워크와 상호 연결되어 있어 가용 영역 간에 높은 처리량과 대기 시간이 짧은 네트워킹을 제공합니다. 영역 간의 모든 트래픽은 암호화됩니다. 네트워크 성능은 영역 간 동기 복제 기능을 충분히 수행할 수 있습니다. 가용 영역은 고가용성을 위해 애플리케이션 파티셔닝 프로세스를 단순화합니다.

고객 책임 '클라우드에서의 복원성'

고객의 책임은 선택한 AWS 클라우드 서비스에 따라 결정됩니다. 이는 복원성 책임의 일부로 수행해야 하는 구성 작업의 양을 결정합니다. 예를 들어 Amazon Elastic Compute Cloud(Amazon EC2)와 같은 서비스의 경우 필요한 모든 복원성 구성 및 관리 작업을 고객이 수행해야 합니다. Amazon EC2 인스턴스를 배포하는 고객은 여러 위치에 EC2 인스턴스를 배포하고(예: AWS 가용 영역), AWS Auto Scaling과 같은 서비스 및 인스턴스에 설치된 애플리케이션에 대한 복원성이 뛰어난 워크로드 아키텍처 모범 사례를 사용하여 자가 복구를 구현할 책임이 있습니다. Amazon S3 및 Amazon DynamoDB와 같은 관리형 서비스의 경우, AWS는 인프라 계층, 운영 체제, 플랫폼을 운영하고 고객은 데이터를 저장하고 검색하기 위해 엔드포인트에 액세스합니다. 사용자는 백업, 버전 관리 및 복제 전략을 포함하여 데이터의 복원성을 관리할 책임이 있습니다.

AWS 리전의 여러 가용 영역에 워크로드를 배포하는 것은 가용 영역 하나로 문제를 격리하여 워크로드를 보호하도록 설계된 고가용성 전략의 일부이며, 다른 가용 영역의 중복성을 사용하여 요청을 계속 처리합니다. 다중 AZ 아키텍처는 정전, 낙뢰, 토네이도, 지진 등과 같은 문제로부터 워크로드를 더 잘 격리하고 보호하도록 설계된 DR 전략의 일부이기도 합니다. DR 전략에서 여러 AWS 리전을 사용할 수도 있습니다. 예를 들어 활성/비활성 구성에서 활성 리전이 더 이상 요청을 처리할 수 없는 경우 워크로드에 대한 서비스가 활성 리전에서 DR 리전으로 장애 조치됩니다.

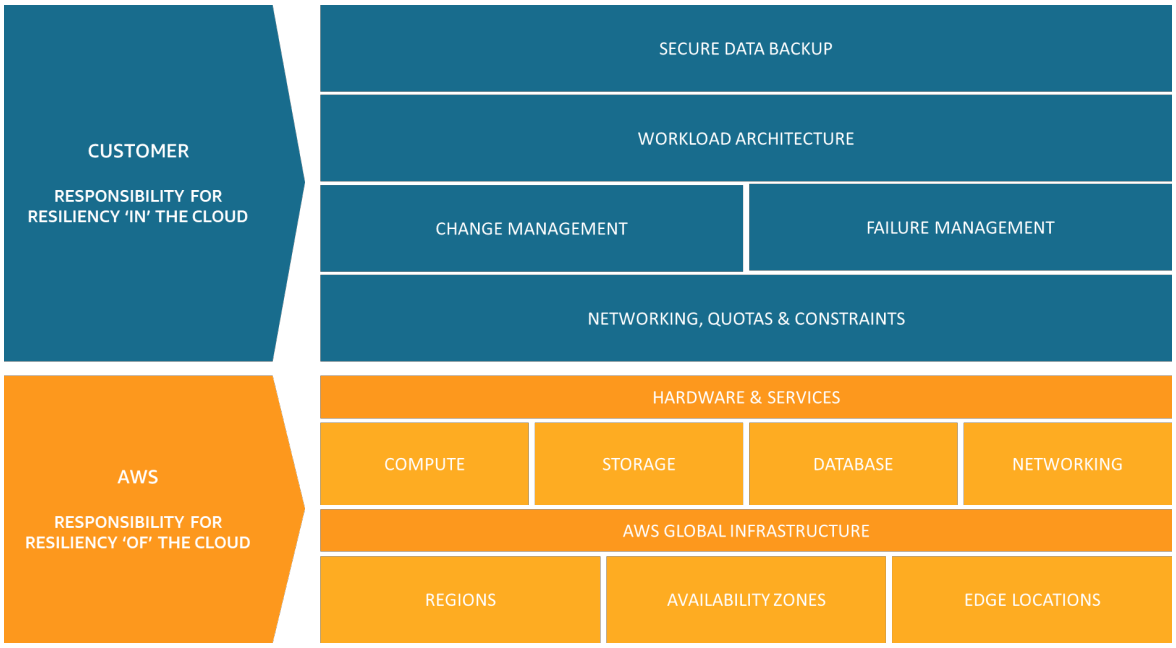

그림 2 - 복원성은 AWS와 고객의 공동 책임

재해란 무엇입니까?

재해 복구를 계획할 때 다음 세 가지 주요 재해 범주에 대한 계획을 평가하세요.

- 지진이나 홍수와 같은 자연 재해
- 정전 또는 네트워크 연결과 같은 기술적 장애
- 부주의한 잘못된 구성 또는 권한이 없는/외부 당사자의 액세스 또는 수정과 같은 사람의 행위

이러한 각 잠재적 재난의 지리적 영향은 그 범위가 현지, 지역, 국가 전체, 대륙 또는 전 세계가 될 수 있습니다. 재해 복구 전략을 고려할 때 재해의 특성과 지리적 영향 모두 중요합니다. 예를 들어 다중 AZ 전략을 사용하면 둘 이상의 가용 영역에 영향을 주지 않으므로 데이터 센터 중단을 유발하는 로컬 플러딩 문제를 완화할 수 있습니다. 그러나 프로덕션 데이터에 대한 공격이 발생한 경우에는 다른 AWS 리전의 백업 데이터로 장애 조치하는 재해 복구 전략을 호출해야 합니다.

고가용성은 재해 복구가 아님

가용성과 재해 복구는 모두 오류 모니터링, 여러 위치에 배포, 자동 장애 조치 등 동일한 모범 사례에 의존합니다. 그러나 가용성은 워크로드의 구성 요소에 초점을 맞추는 반면 재해 복구는 전체 워크로드의 개별 복사본에 중점을 둡니다. 재해 복구의 목표는 가용성과는 다르며, 재해로 간주되는 대규모 이벤트 이후의 복구 시간을 측정합니다. 가용성에 영향을 미치는 이벤트가 발생할 경우 고가용성 아키텍처를 통해 고객의 요구를 충족할 수 있으므로 먼저 워크로드가 가용성 목표를 충족하는지 확인해야 합니다. 재해 복구 전략에는 가용성에 대한 방식과는 다른 방식이 필요하며, 필요한 경우 전체 워크로드에 대해 장애 조치를 수행할 수 있도록 개별 시스템을 여러 위치에 배포하는 데 중점을 둡니다.

재해 복구 계획에서 워크로드의 가용성은 어떤 방식을 취하느냐에 영향을 주므로 반드시 고려해야 합니다. 한 가용 영역의 단일 Amazon EC2 인스턴스에서 실행되는 워크로드는 고가용성이 없습니다. 로컬 플러딩 문제가 해당 가용 영역에 영향을 미치는 경우 이 시나리오에서는 DR 목표를 달성하기 위해 다른 AZ로의 장애 조치가 필요합니다. 이 시나리오를 다중 사이트 활성/활성로 배포된 고가용성 워크로드와 비교해봅니다. 이 워크로드는 여러 활성 리전에 배포되고 모든 리전은 프로덕션 트래픽을 처리합니다. 이 경우 예상치 못한 대규모 재해가 전체 리전에 영향을 주더라도 DR 전략은 모든 트래픽을 나머지 리전으로 라우팅하여 수행됩니다.

데이터에 접근하는 방식에서도 가용성과 재해 복구는 다릅니다. 고가용성(예: 다중 사이트, 활성/활성 워크로드)을 달성하기 위해 다른 사이트로 지속적으로 복제하는 스토리지 솔루션이 있다고 가정해 보겠습니다. 운영 스토리지 디바이스에서 파일이 삭제되거나 손상된 경우 이러한 파괴적인 변경 사항을 보조 스토리지 디바이스에 복제할 수 있습니다. 이 시나리오에서는 고가용성에도 불구하고 데이터 삭제 또는 손상 시 장애 조치 기능이 손상됩니다. 대신 DR 전략의 일부로 특정 시점으로 백업이 필요합니다.

비즈니스 연속성 계획(BCP)

재해 복구 계획은 독립 실행형 문서가 아니라 조직의 비즈니스 연속성 계획(BCP)에 포함되어야 합니다. 재해가 워크로드 이외의 비즈니스 요소에 미치는 영향으로 인해 워크로드의 비즈니스 목표를 달성할 수 없는 경우 워크로드를 복원하기 위한 적극적인 재해 복구 목표를 유지할 필요가 없습니다. 예를 들어 지진으로 인해 전자 상거래 애플리케이션에서 구매한 제품을 운송하지 못할 수 있습니다. 이 경우 효과적인 DR로 워크로드가 계속 작동하더라도 BCP에 운송 요구 사항을 포함해야 합니다. DR 전략은 비즈니스 요구 사항, 우선 순위 및 컨텍스트를 기반으로 해야 합니다.

비즈니스 영향 분석 및 위험 평가

비즈니스 영향 분석은 워크로드에 미치는 비즈니스 중단의 영향을 정량화해야 합니다. 워크로드를 사용할 수 없는 내부 및 외부 고객에게 미치는 영향과 비즈니스에 미치는 영향을 파악해야 합니다. 분석은 워크로드를 얼마나 빨리 가용 상태로 전환해야 하는지, 얼마나 많은 데이터 손실을 허용할 수 있는지를 결정하는 데 도움이 됩니다. 그러나 복구 목표를 개별적으로 설정해서는 안 됩니다. 중단 가능성과 복구 비용은 워크로드에 재해 복구를 제공하는 비즈니스 가치를 알리는 데 도움이 되는 핵심 요소입니다.

비즈니스에 미치는 영향은 시기에 따라 달라질 수 있으며 재해 복구 계획 시 이 점을 고려하는 것이 좋습니다. 예를 들어, 급여 시스템의 중단은 모든 사람이 급여를 받기 직전에는 비즈니스에 매우 큰 영향을 미칠 수 있지만 모든 사람이 급여를 받은 직후에는 영향이 적을 수 있습니다.

워크로드의 기술적 구현에 대한 개요와 함께 재해의 유형 및 지리적 영향에 대한 위험 평가를 수행하면 각 유형의 재해 시 중단이 발생할 가능성을 확인할 수 있습니다.

매우 중요한 워크로드의 경우 비즈니스에 미치는 영향을 최소화하기 위해 지속적인 백업을 통해 여러 리전에 걸친 고가용성을 검토할 수 있습니다. 덜 중요한 워크로드의 경우 재해 복구를 전혀 수행하지 않는 것이 유효한 전략일 수 있습니다. 또한 일부 재해 시나리오의 경우 재해 발생 가능성이 낮기 때문에 정보에 입각한 결정에 따라 재해 복구 전략을 마련하지 않는 것이 타당합니다. AWS 리전 내의 가용 영역은 이미 두 영역 사이에 의미 있는 거리를 두고 설계되어 대부분의 일반적인 재해가 한 영역에만 영향을 미치고 다른 영역에는 영향을 주지 않도록 주의 깊게 위치가 계획되어 있습니다. 따라서 AWS 리전 내의 다중 AZ 아키텍처가 위험 완화 요구 사항을 이미 충족할 수도 있습니다.

재해 복구 전략이 비즈니스에 미치는 영향과 위험을 고려하여 적합한 수준의 비즈니스 가치를 제공하는지 확인하기 위해 재해 복구 옵션의 비용을 평가해야 합니다.

이 모든 정보에 입각하여 다양한 재해 시나리오가 가져오는 위협, 위험, 영향 및 비용과 관련 복구 옵션을 문서화할 수 있습니다. 이 정보를 기반으로 각 워크로드에 대한 복구 목표를 결정해야 합니다.

복구 목표(RTO 및 RPO)

재해 복구(DR) 전략을 수립할 때 조직은 일반적으로 복구 시간 목표(RTO)와 복구 시점 목표(RPO)를 계획합니다.

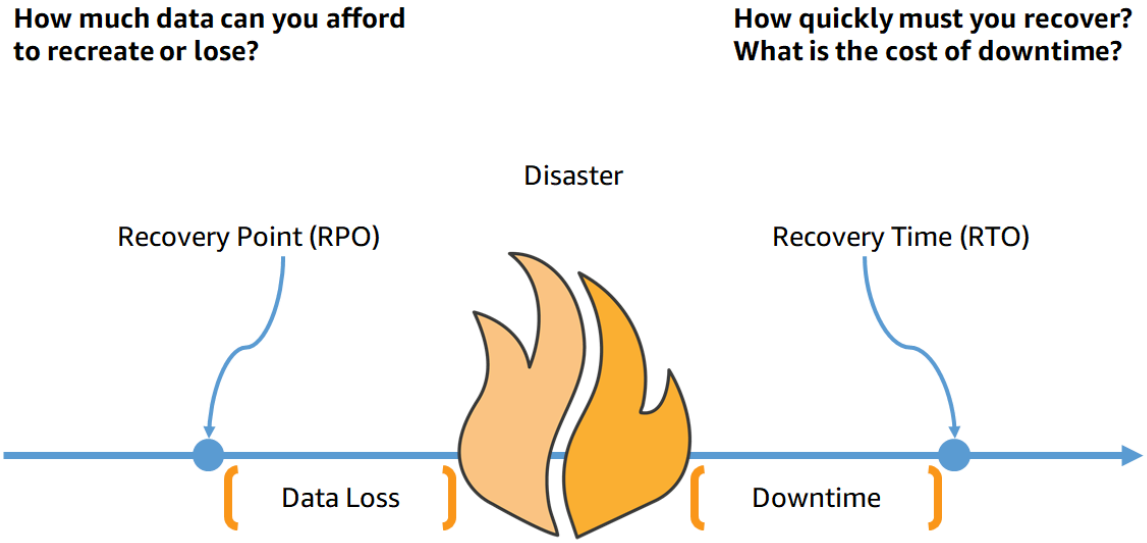

그림 3 - 복구 목표

복구 시간 목표(RTO)는 서비스 중단 시점과 서비스 복원 시점 간에 허용되는 최대 지연 시간으로, 서비스 불가능 상태가 허용되는 기간을 고려하여 결정되며, 조직에서 정의합니다.

이 백서에서는 백업 및 복원, 파일럿 라이트, 웜 스탠바이, 다중 사이트 활성/활성(클라우드의 재해 복구 옵션 (p. 13) 참조)의 4가지 DR 전략에 대해 설명합니다. 다음 다이어그램에서 기업은 최대 허용 RTO와 서비스 복원 전략에 사용할 수 있는 금액의 한도를 결정했습니다. 비즈니스 목표를 고려할 때 파일럿 라이트 또는 웜 스탠바이 DR 전략은 RTO와 비용 기준을 모두 충족합니다.

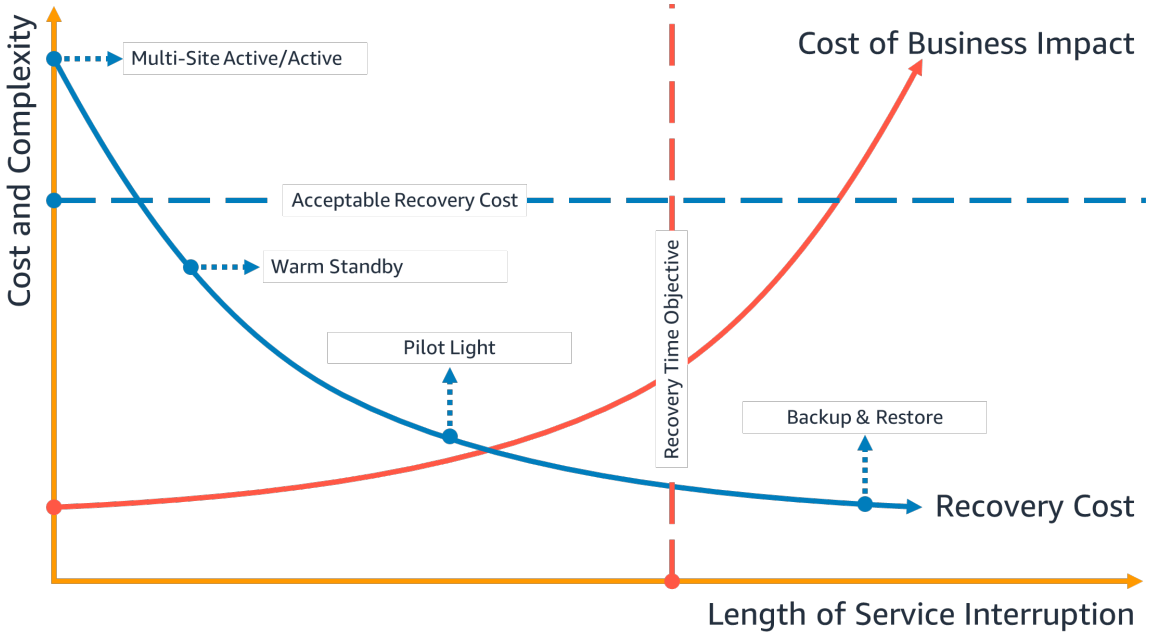

그림 4 - 복구 시간 목표

복구 시점 목표(RPO)는 마지막 데이터 복구 시점 이후 허용되는 최대 시간으로, 마지막 복구 시점과 서비스 중단 시점 사이에 허용되는 데이터 손실량을 고려하여 결정되며, 조직에서 정의합니다.

다음 다이어그램에서 기업은 최대 허용 RPO와 데이터 복구 전략에 사용할 수 있는 비용의 한도를 결정했습니다. 네 가지 DR 전략 중 파일럿 라이트 또는 웜 스탠바이 DR 전략이 RPO 및 비용에 대한 두 가지 기준을 모두 충족합니다.

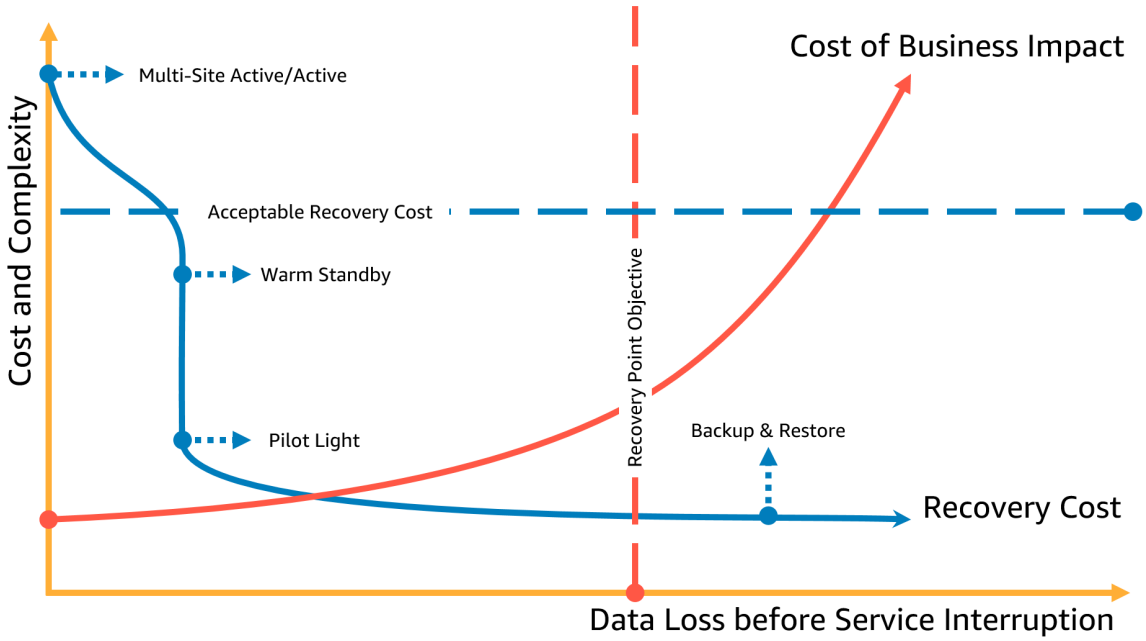

그림 5 - 복구 지점 목표

Note

복구 비용이 장애 또는 손실 비용보다 높은 경우 규정 요구 사항과 같은 부차적 요인이 없는 한 복구 옵션을 적용해서는 안 됩니다.

클라우드에서는 재해 복구 방식이 다름

재해 복구 전략은 기술 혁신과 함께 진화합니다. 온프레미스 재해 복구 계획에는 테이프를 물리적으로 전송하거나 데이터를 다른 사이트로 복제하는 작업이 포함될 수 있습니다. 조직은 AWS를 기반으로 DR 목표를 달성하기 위해 이전 재해 복구 전략의 비즈니스 영향, 위험 및 비용을 재평가해야 합니다. AWS 클라우드의 재해 복구는 기존 환경에 비해 다음과 같은 이점이 있습니다.

- 복잡성을 줄이면서 재해로부터 신속하게 복구
- 간단하고 반복 가능한 테스트를 통해 보다 쉽고 잦은 테스트 가능
- 관리 오버헤드 감소로 운영 부담 경감
- 자동화를 통해 오류 발생 가능성을 줄이고 복구 시간 단축

AWS를 사용하면 물리적 백업 데이터 센터의 고정 자본 비용을 용량 최적화된 클라우드 환경의 가변 운영 비용으로 바꿀 수 있습니다.

많은 조직에서 온프레미스 재해 복구는 워크로드 또는 데이터 센터의 워크로드에 대한 중단 위험과, 백업 또는 복제된 데이터를 보조 데이터 센터로 복구하는 것을 기반으로 했습니다. AWS에 워크로드를 배포하는 조직은 잘 설계된 워크로드를 구현하고 AWS 글로벌 클라우드 인프라 설계를 활용하여 이러한 중단의 영향을 완화할 수 있습니다. 클라우드에서 안정적이고 안전하며 효율적이고 경제적인 워크로드를 설계 및 운영하기 위한 아키텍처 모범 사례에 대한 자세한 내용은 AWS Well-Architected Framework - 안정성 원칙 백서를 참조하세요.

워크로드가 AWS에 있는 경우 데이터 센터 연결(액세스 가능 여부는 제외), 전력, 에어컨, 화재 진압 및 하드웨어에 대해 걱정할 필요가 없습니다. 이 모든 것이 자동으로 관리되며 장애로부터 격리된 여러 가용 영역(각각 하나 이상의 개별 데이터 센터로 구성됨)에 액세스할 수 있습니다.

단일 AWS 리전

물리적 데이터 센터 하나의 중단 또는 손실에 따른 재해 이벤트의 경우, 단일 AWS 리전 내의 여러 가용 영역에 고가용성 워크로드를 구현하면 자연 재해 및 기술 재해의 위험을 완화하고 데이터 손실을 야기할 수 있는 오류 또는 무단 활동과 같은 인적 위협의 위험을 줄일 수 있습니다. 각 AWS 리전은 여러 가용 영역으로 구성되며, 각 가용 영역은 다른 영역의 장애로부터 격리됩니다. 각 가용 영역은 물리적 데이터 센터 여러 개로 구성됩니다. 동일한 리전의 여러 영역에 걸쳐 워크로드를 분할하면 파급력이 큰 문제를 더 잘 격리하고 고가용성을 달성할 수 있습니다. 가용 영역은 물리적 중복성을 위해 설계되었으며, 복원성을 제공하여 정전, 인터넷 가동 중지, 홍수 및 기타 자연 재해가 발생하더라도 중단되지 않는 성능을 제공합니다. AWS에서 이러한 작업을 수행하는 방법을 알아보려면 AWS 글로벌 클라우드 인프라를 참조하세요.

단일 AWS 리전의 여러 가용 영역에 배포하면 단일(또는 여러) 데이터 센터의 장애로부터 워크로드를 더 잘 보호할 수 있습니다. 단일 리전 배포에 대한 보장성을 강화하기 위해 데이터 및 구성(인프라 정의 포함)을 다른 리전에 백업할 수 있습니다. 이 전략은 데이터 백업 및 복원만 포함하도록 재해 복구 계획의 범위를 줄입니다. 다른 AWS 리전으로 백업하여 다중 리전 복원성을 활용하는 것은 다음 단원에서 설명하는 기타 다중 리전 옵션에 비해 간단하고 저렴합니다. 예를 들어 Amazon Simple Storage Service(Amazon S3)에 백업하면 데이터를 즉시 검색할 수 있습니다. 그러나 일부 데이터에 대한 DR 전략에서 검색 시간 요구 사항이 더 완화되는 경우(몇 분에서 몇 시간으로) Amazon S3 Glacier 또는 Amazon S3 Glacier Deep Archive를 사용하면 백업 및 복구 전략 비용을 크게 줄일 수 있습니다.

일부 워크로드에는 데이터 상주 규제 요구 사항이 있을 수 있습니다. 이 요구 사항이 현재 AWS 리전이 하나만 있는 지역의 워크로드에 적용되는 경우 위에서 설명한 대로 고가용성을 위한 다중 AZ 워크로드를 설계하

는 것 외에도 해당 리전 내의 AZ를 개별 위치로 사용할 수 있으므로 해당 리전 내 워크로드에 적용되는 데이터 상주 요구 사항을 해결하는 데 도움이 될 수 있습니다. 다음 단원에서 설명하는 DR 전략은 여러 AWS 리전을 사용하지만 리전 대신 가용 영역을 사용하여 구현할 수도 있습니다.

다중 AWS 리전

서로 멀리 떨어져 있는 여러 데이터 센터에 손실이 발생할 수 있는 위험이 포함된 재해 이벤트의 경우, AWS 내 전체 리전에 영향을 미치는 자연 재해 및 기술 재해에 대비할 수 있는 재해 복구 옵션을 고려해야 합니다. 다음 단원에서 설명하는 모든 옵션은 이러한 재해로부터 보호하기 위해 다중 리전 아키텍처로 구현할 수 있습니다.

클라우드의 재해 복구 옵션

AWS 내에서 사용할 수 있는 재해 복구 전략은 백업을 만드는 비용이 저렴하고 복잡성이 낮은 것부터 여러 활성 리전을 사용하는 보다 복잡한 전략에 이르기까지 크게 4가지 방식으로 분류할 수 있습니다. 필요할 때 바로 실행할 수 있도록 재해 복구 전략을 정기적으로 테스트하는 것이 중요합니다.

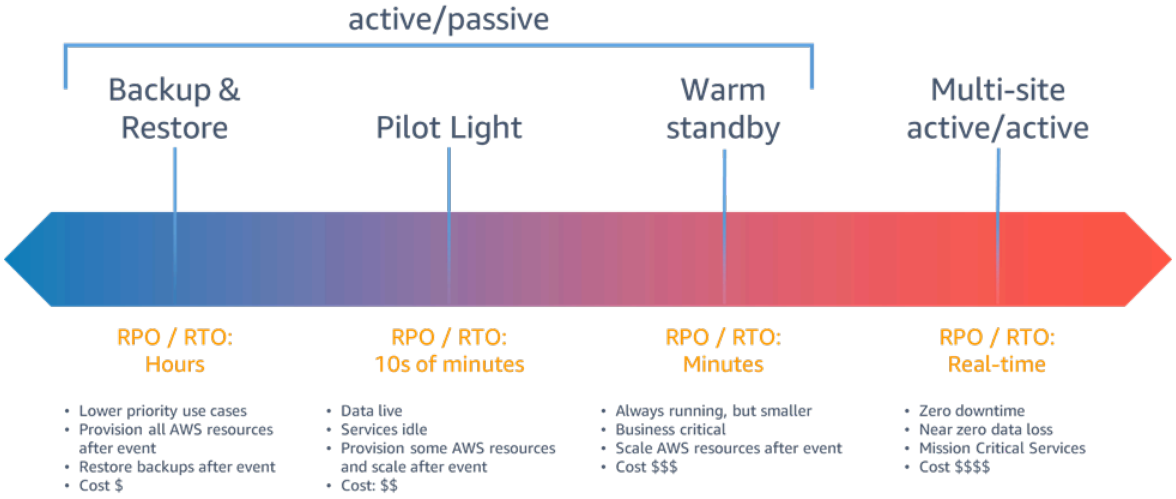

그림 6 - 재해 복구 전략

잘 설계되고 가용성이 높은 워크로드를 위한 단일 물리적 데이터 센터의 중단 또는 손실을 야기하는 재해 이벤트의 경우 재해 복구에 대한 백업 및 복원 방식만 필요할 수 있습니다. 재해에 대한 정의가 물리적 데이터 센터의 중단 또는 손실을 넘어 한 리전의 중단 또는 손실로 이어지거나 이를 요구하는 규정 요구 사항의 적용을 받는 경우에는 파일럿 라이트, 웜 스탠바이 또는 다중 사이트 활성/활성 방식을 고려해야 합니다.

백업 및 복원

백업 및 복원은 데이터 손실 또는 손상을 완화하는 데 적합한 방식입니다. 이 방식은 데이터를 다른 AWS 리전으로 복제하여 리전 재해에 대비하거나 단일 가용 영역에 배포된 워크로드의 중복성 부족을 완화하는 데에도 사용할 수 있습니다. 데이터 외에도 복구 리전에 인프라, 구성 및 애플리케이션 코드를 재배포해야 합니다. 인프라를 오류 없이 신속하게 재배포할 수 있으려면 항상 AWS CloudFormation 또는 AWS Cloud Development Kit (AWS CDK)와 같은 서비스를 사용하는 코드형 인프라(IaC)를 사용하여 배포해야 합니다. IaC를 사용하지 않으면 복구 리전에서 워크로드를 복원하는 것이 복잡할 수 있으며, 이로 인해 복구 시간이 늘어나고 RTO를 초과할 수 있습니다. 사용자 데이터 외에도 Amazon EC2 인스턴스를 생성하는 데 사용하는Amazon Machine Image(AMI)를 비롯한 코드 및 구성도 백업해야 합니다. AWS CodePipeline을 사용하여 애플리케이션 코드 및 구성의 재배포를 자동화할 수 있습니다.

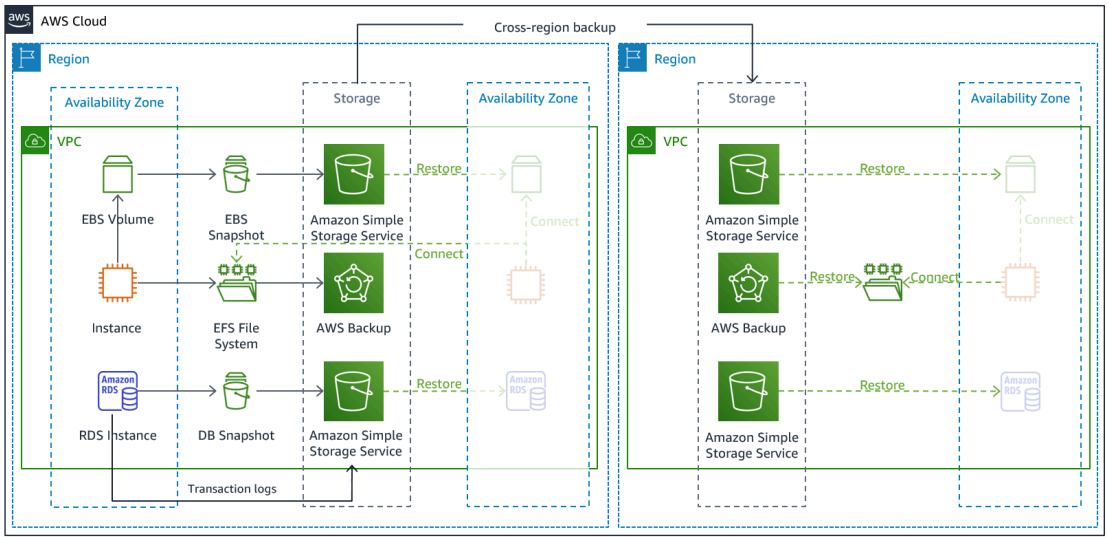

그림 7 - 백업 및 복원 아키텍처

AWS 서비스

워크로드 데이터에는 주기적으로 실행되거나 지속적인 백업 전략이 필요합니다. 백업을 얼마나 자주 실행하느냐에 따라 달성 가능한 복구 시점이 결정됩니다(RPO에 맞게 조정되어야 함). 또한 백업의 경우 백업을 수행한 시점으로 복원할 수 있는 방법을 제공해야 합니다. 특정 시점으로 복구를 사용한 백업은 다음 서비스 및 리소스를 통해 사용할 수 있습니다.

- Amazon Elastic Block Store(Amazon EBS) 스냅샷
- Amazon DynamoDB 백업
- Amazon RDS 스냅샷
- Amazon Aurora DB 스냅샷
- Amazon EFS 백업(AWS Backup을 사용하는 경우)
- Amazon Redshift 스냅샷
- Amazon Neptune 스냅샷

Amazon Simple Storage Service(Amazon S3)의 경우 Amazon S3 교차 리전 복제(CRR)를 사용하여 DR 리전의 S3 버킷에 객체를 비동기적으로 복사하는 동시에 저장된 객체에 대한 버전 관리를 제공하여 복원 지점을 선택할 수 있습니다. 연속 데이터 복제는 데이터 백업 시간이 가장 짧다는 장점이 있지만 데이터 손상이나 악의적인 공격(예: 데이터 무단 삭제) 및 특정 시점으로 백업과 같은 재해 이벤트로부터 데이터를 보호하지 못할 수 있습니다. 연속 복제는 파일럿 라이트를 위한 AWS 서비스 (p. 17) 단원에서 다룹니다.

AWS Backup은 다음 서비스 및 리소스에 대한 AWS 백업 기능을 구성, 예약 및 모니터링할 수 있는 중앙 위치를 제공합니다.

- Amazon Elastic Block Store(Amazon EBS) 볼륨
- Amazon EC2 인스턴스
- Amazon Relational Database Service(Amazon RDS) 데이터베이스(Amazon Aurora 데이터베이스 포함)
- Amazon DynamoDB 테이블
- Amazon Elastic File System(Amazon EFS) 파일 시스템
- AWS Storage Gateway 볼륨

- Amazon FSx for Windows File Server 및 Amazon FSx for Lustre

AWS Backup은 리전 간 백업 복사(예: 재해 복구 리전으로 복사)를 지원합니다.

Amazon S3 데이터에 대한 추가 재해 복구 전략으로 S3 객체 버전 관리를 활성화는 것이 좋습니다. 객체 버전 관리는 작업 전의 원래 버전을 유지함으로써 S3의 데이터를 삭제 또는 수정 작업의 결과로부터 보호합니다. 객체 버전 관리는 사람의 실수로 인한 재해의 위험을 완화하는 데 유용할 수 있습니다. S3 복제를 사용하여 DR 리전에 데이터를 백업하는 경우, 기본적으로 소스 버킷에서 객체가 삭제되면 Amazon S3은 소스 버킷에만 삭제 마커를 추가합니다. 이 방식은 DR 리전의 데이터를 소스 리전의 악의적인 삭제로부터 보호합니다.

데이터 외에도 워크로드를 재배포하고 복구 시간 목표(RTO)를 달성하는 데 필요한 구성 및 인프라를 백업해야 합니다. AWS CloudFormation은 코드형 인프라(IaC)를 제공하며, 워크로드의 모든 AWS 리소스를 정의할 수 있으므로 여러 AWS 계정 및 AWS 리전에 안정적으로 배포 및 재배포할 수 있습니다. 워크로드에서 사용하는 Amazon EC2 인스턴스를 Amazon Machine Image(AMI)로 백업할 수 있습니다. AMI는 인스턴스의 루트 볼륨과 인스턴스에 연결된 다른 EBS 볼륨의 스냅샷에서 생성됩니다. 이 AMI를 사용하여 EC2 인스턴스의 복원된 버전을 시작할 수 있습니다. AMI 복사는 리전 내에서 또는 리전 간에 수행할 수 있습니다. 또는 AWS Backup을 사용하여 계정 간에 또는 다른 AWS 리전으로 백업을 복사할 수 있습니다. 교차 계정 백업 기능은 내부 위협 또는 계정 침해를 포함한 재해 이벤트로부터 보호하는 데 도움이 됩니다. 또한 AWS Backup은 인스턴스의 개별 EBS 볼륨 외에도 EC2 백업을 위한 추가 기능을 추가로 제공합니다. AWS Backup은 인스턴스 유형, 구성된 Virtual Private Cloud(VPC), 보안 그룹, IAM 역할, 모니터링 구성 및 태그 등의 메타데이터를 저장하고 추적합니다. 하지만 이 추가 메타데이터는 EC2 백업을 동일한 AWS 리전으로 복원할 때만 사용됩니다.

재해 복구 리전에 백업으로 저장된 모든 데이터는 장애 조치 시 복원해야 합니다. AWS Backup은 복원 기능을 제공하지만 현재 예약된 복원 또는 자동 복원은 사용할 수 없습니다. AWS Backup용 API를 호출하는 AWS SDK를 사용하여 DR 리전에 자동 복원을 구현할 수 있습니다. 이를 정기적으로 반복하는 작업으로 설정하거나 백업이 완료될 때마다 복원을 트리거할 수 있습니다. 다음 그림은 Amazon Simple Notification Service(Amazon SNS) 및 AWS Lambda를 사용한 자동 복원의 예를 보여 줍니다.

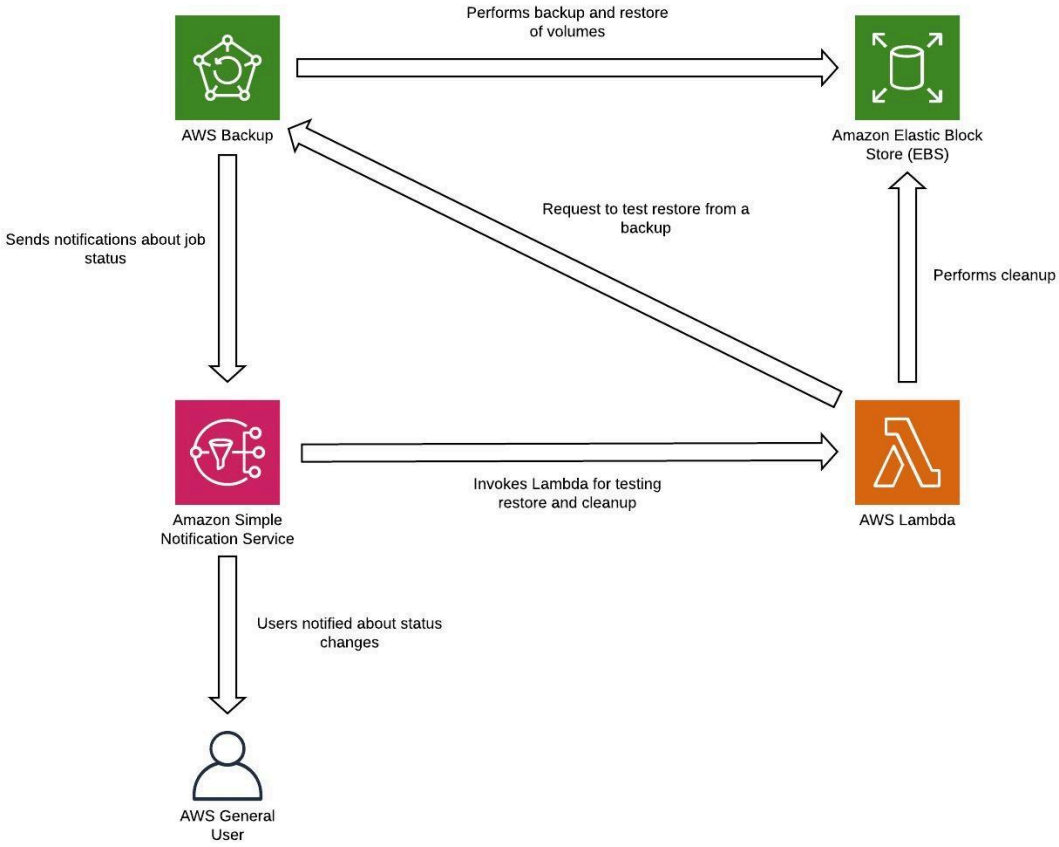

그림 8 - 백업 복원 및 테스트

Note

백업 전략에는 백업 테스트가 포함되어야 합니다. 자세한 내용은 재해 복구 테스트 (p. 23) 단원을 참조하세요. 구현 실습 데모는 AWS Well-Architected 랩: 데이터 백업 및 복원 테스트를 참조하세요.

파일럿 라이트

파일럿 라이트 방식을 사용하면 한 리전에서 다른 리전으로 데이터를 복제하고 핵심 워크로드 인프라의 복사본을 프로비저닝할 수 있습니다. 데이터베이스 및 객체 스토리지와 같이 데이터 복제 및 백업을 지원하는 데 필요한 리소스는 항상 켜져 있습니다. 애플리케이션 서버와 같은 다른 요소는 애플리케이션 코드 및 구성과 함께 로드되지만 꺼져 있으며 테스트 중에 또는 재해 복구 장애 조치가 호출될 때만 사용됩니다. 백업 및 복원 방식과 달리 핵심 인프라는 항상 사용할 수 있으며 애플리케이션 서버를 켜고 확장하여 전체 규모의 프로덕션 환경을 신속하게 프로비저닝할 수 있는 옵션이 항상 있습니다.

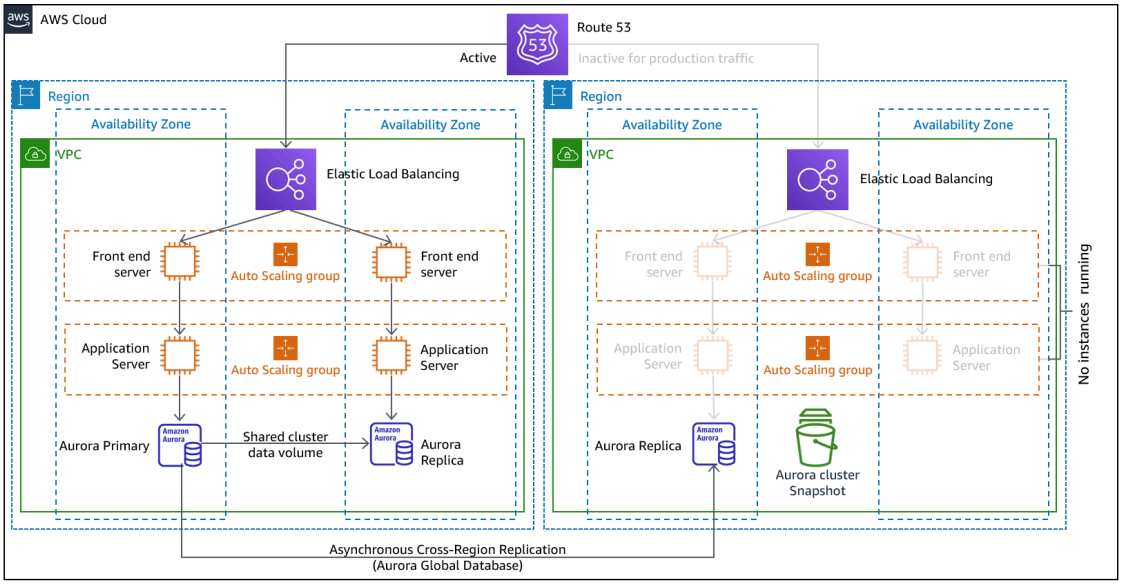

그림 9 - 파일럿 라이트 아키텍처

파일럿 라이트 방식은 활성 리소스를 최소화하여 지속적인 재해 복구 비용을 최소화하며 핵심 인프라 요구 사항이 모두 갖추어져 있기 때문에 재해 발생 시 복구를 단순화합니다. 이 복구 옵션을 사용하려면 배포 방식을 변경해야 합니다. 각 리전에서 핵심 인프라를 변경하고 워크로드(구성, 코드) 변경 사항을 각 리전에 동시에 배포해야 합니다. 배포를 자동화하고 코드형 인프라(IaC)를 사용하여 여러 계정 및 리전에 인프라를 배포하면 이 단계를 간소화할 수 있습니다(전체 인프라를 기본 리전에 배포하고 DR 리전으로의 인프라 배포는 축소하거나 끔). 리전별로 다른 계정을 사용하여 최고 수준의 리소스 및 보안 격리를 제공하는 것이 좋습니다(손상된 자격 증명이 재해 복구 계획의 일부인 경우).

이 방식을 사용하면 데이터 재해에도 대비해야 합니다. 지속적인 데이터 복제는 일부 유형의 재해로부터 사용자를 보호하지만 저장된 데이터의 버전 관리나 특정 시점으로 복구 옵션까지 전략에 포함하지 않으면 데이터 손상 또는 파괴로부터 사용자를 보호하지 못할 수 있습니다. 재해 리전에서 복제한 데이터를 백업하여 동일한 리전에 특정 시점으로 백업을 생성할 수 있습니다.

AWS 서비스

백업 및 복원 (p. 13) 단원에서 다루는 AWS 서비스를 사용하여 특정 시점으로 백업을 생성하는 것 외에도 파일럿 라이트 전략을 사용할 때 다음 서비스도 고려하는 것이 좋습니다.

파일럿 라이트의 경우 DR 리전의 라이브 데이터베이스 및 데이터 스토어로의 지속적인 데이터 복제는 낮은 RPO에 가장 적합한 방법입니다(앞에서 설명한 특정 시점으로 백업과 함께 사용하는 경우). AWS는 다음 서비스 및 리소스를 사용하여 데이터에 대한 지속적인 리전 간 비동기 데이터 복제를 제공합니다.

- Amazon Simple Storage Service(Amazon S3) 복제
- Amazon RDS 읽기 전용 복제본
- Amazon Aurora Global Database
- Amazon DynamoDB 글로벌 테이블

연속 복제를 사용하면 DR 리전에서 거의 즉시 데이터 버전을 사용할 수 있습니다. 실제 복제 시간은 S3 객체용 S3 Replication Time Control(S3 RTC) 및 Amazon Aurora Global Database의 관리 기능과 같은 서비스를 사용하여 모니터링할 수 있습니다.

재해 복구 리전에서 읽기/쓰기 워크로드를 실행하기 위해 장애 조치를 수행하는 경우 RDS 읽기 전용 복제본을 기본 인스턴스로 승격해야 합니다. Aurora가 아닌 DB 인스턴스의 경우 프로세스를 완료하는 데 몇 분이

걸리며 재부팅도 프로세스의 일부입니다. RDS를 통한 교차 리전 복제(CRR) 및 장애 조치의 경우 Amazon Aurora Global Database를 사용하면 몇 가지 이점이 있습니다. Global Database는 전용 인프라를 사용하여 애플리케이션을 지원하는 데 데이터베이스를 완전히 사용할 수 있으며, 일반적인 대기 시간이 1초 미만인 보조 리전(AWS 리전 내에서는 100밀리초 미만)에 복제할 수 있습니다. Amazon Aurora Global Database를 사용하면 기본 리전의 성능 저하 또는 중단이 발생하는 경우 리전 전체가 중단되더라도 1분 이내에 보조 리전 중 하나를 승격하여 읽기/쓰기 작업을 맡도록 할 수 있습니다. 승격은 자동으로 수행될 수 있으며 재부팅할 필요가 없습니다.

리소스가 더 적거나 작은 핵심 워크로드 인프라의 축소 버전을 DR 리전에 배포해야 합니다. AWS CloudFormation을 사용하면 인프라를 정의하고 AWS 계정과 AWS 리전 전체에 일관되게 배포할 수 있습니다. AWS CloudFormation에서는 사전 정의된 의사 파라미터를 사용하여 AWS 계정 및 해당 계정이 배포된 AWS 리전을 식별합니다. 따라서 CloudFormation 템플릿에 조건 로직을 구현하여 DR 리전에 축소 버전의 인프라만 배포할 수 있습니다. EC2 인스턴스 배포의 경우 Amazon Machine Image(AMI)가 하드웨어 구성 및 설치된 소프트웨어와 같은 정보를 제공합니다. 필요한 AMI를 생성하는 Image Builder 파이프라인을 구현하고 이를 기본 리전과 백업 리전 모두에 복사할 수 있습니다. 이렇게 하면 재해 발생 시 새 리전에서 워크로드를 재배포하거나 확장하는 데 필요한 모든 것을 이 골든 AMI에 포함할 수 있습니다. Amazon EC2 인스턴스는 축소된 구성(기본 리전보다 인스턴스 수가 적음)으로 배포됩니다. 최대 절전 모드를 사용하여 EC2 인스턴스를 중지된 상태로 전환하면 EC2 비용을 지불하지 않고 사용한 스토리지에 대해서만 비용을 지불하면 됩니다. EC2 인스턴스를 시작하려면 AWS Command Line Interface(CLI) 또는 AWS SDK를 사용하여 스크립트를 생성하면 됩니다. 프로덕션 트래픽을 지원하도록 인프라를 확장하려면 웜 스탠바이 (p. 19) 단원의 AWS Auto Scaling을 참조하세요.

파일럿 라이트와 같은 활성/대기 구성의 경우 모든 트래픽은 처음에 기본 리전으로 이동하고 기본 리전을 더 이상 사용할 수 없는 경우 재해 복구 리전으로 전환됩니다. AWS 서비스 사용을 고려할 트래픽 관리 옵션에는 두 가지가 있습니다. 첫 번째 옵션은 Amazon Route 53을 사용하는 것입니다. Amazon Route 53을 사용하면 하나 이상의 AWS 리전에 있는 여러 IP 엔드포인트를 하나의 Route 53 도메인 이름과 연결할 수 있습니다. 그런 다음 해당 도메인 이름의 적절한 엔드포인트로 트래픽을 라우팅할 수 있습니다. Amazon Route 53 상태 확인이 이러한 엔드포인트를 모니터링합니다. 이러한 상태 확인을 사용하여 트래픽이 정상 상태의 엔드포인트로 전송되도록 DNS 장애 조치를 구성할 수 있습니다.

두 번째 옵션은 AWS Global Accelerator를 사용하는 것입니다. AnyCast IP를 사용하면 하나 이상의 AWS 리전에 있는 여러 엔드포인트를 동일한 고정 IP 주소로 연결할 수 있습니다. 그런 다음 AWS Global Accelerator는 해당 주소와 연결된 적절한 엔드포인트로 트래픽을 라우팅합니다. Global Accelerator 상태 확인 기능이 엔드포인트를 모니터링합니다. AWS Global Accelerator는 이러한 상태 확인 기능을 사용하여 애플리케이션의 상태를 자동으로 확인하고 사용자 트래픽을 정상적인 애플리케이션 엔드포인트로만 라우팅합니다. Global Accelerator는 광범위한 AWS 엣지 네트워크를 사용하여 트래픽을 가능한 한 빨리 AWS 네트워크 백본에 배치하므로 애플리케이션 엔드포인트에 대한 대기 시간을 줄입니다. Global Accelerator는 Route 53과 같은 DNS 시스템에서 발생할 수 있는 캐싱 문제도 방지합니다.

CloudEndure Disaster Recovery

AWS Marketplace에서 사용할 수 있는 CloudEndure Disaster Recovery는 기본 서버의 블록 수준 복제를 사용하여 서버에서 호스팅되는 애플리케이션 및 서버에서 호스팅되는 모든 소스의 데이터베이스를 지속적으로 복제합니다. CloudEndure Disaster Recovery를 사용하면 AWS 클라우드를 온프레미스 워크로드 및 해당 환경의 재해 복구 리전으로 사용할 수 있습니다. AWS에서 호스팅되는 워크로드가 EC2(RDS 아님)에서 호스팅되는 애플리케이션 및 데이터베이스로만 구성된 경우 해당 워크로드의 재해 복구에도 CloudEndure Disaster Recovery를 사용할 수 있습니다. CloudEndure Disaster Recovery는 파일럿 라이트 전략을 사용하여 준비 영역으로 사용되는 Amazon Virtual Private Cloud(Amazon VPC)에서 데이터 및 꺼진 리소스의 복사본을 유지 관리합니다. 장애 조치 이벤트가 트리거되면 준비된 리소스를 사용하여 복구 위치로 사용되는 대상 Amazon VPC에 전체 용량 배포를 자동으로 생성합니다.

그림 10 - CloudEndure Disaster Recovery 아키텍처

웜 스탠바이

웜 스탠바이 방식에는 축소되었지만 완전히 작동하는 프로덕션 환경의 복사본이 다른 리전에 있는지 확인하는 작업이 포함됩니다. 이 방식은 워크로드가 다른 리전에서 항상 켜져 있기 때문에 파일럿 라이트 개념을 확장하고 복구 시간을 단축합니다. 또한 이 방식을 사용하면 보다 쉽게 테스트를 수행하거나 지속적인 테스트를 구현하여 재해 복구 역량에 대한 확신을 높일 수 있습니다.

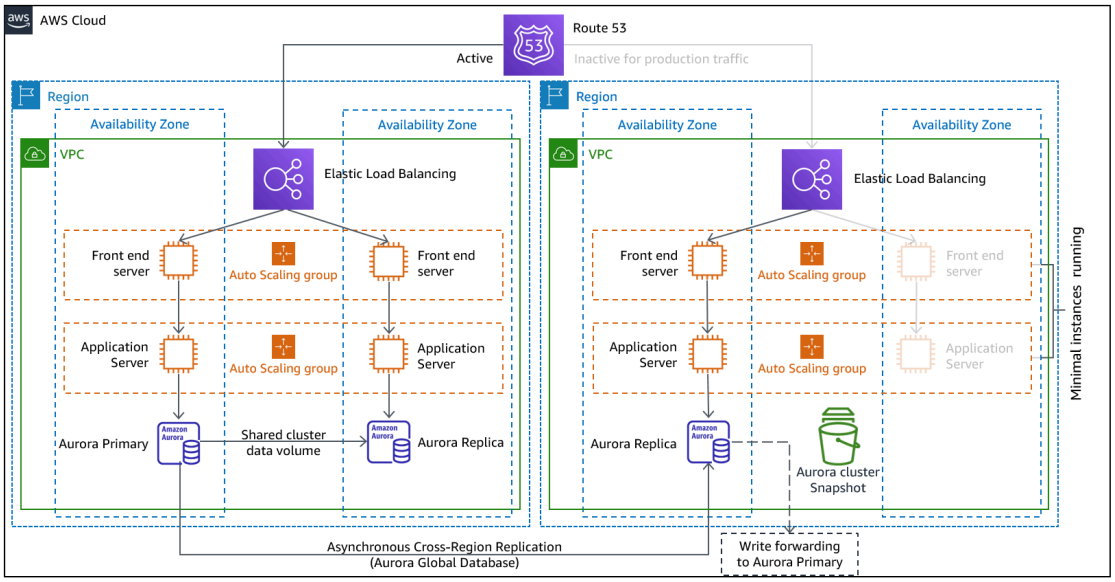

그림 11 - 웜 스탠바이 아키텍처

참고: 파일럿 라이트 (p. 16)와 웜 스탠바이 (p. 19)의 차이를 이해하기 어려울 수 있습니다. 둘 다 기본 리전 자산의 복사본이 있는 DR 리전의 환경을 포함합니다. 차이점은 파일럿 라이트는 먼저 추가 조치를 취하지 않으면 요청을 처리할 수 없는 반면, 웜 스탠바이는 트래픽을 (감소된 용량 수준에서) 즉시 처리할 수 있다는 것입니다. 파일럿 라이트 방식에서는 서버를 '켜고', 추가(비 코어) 인프라를 배포하고 확장해야 하는 반면, 웜 스탠바이 방식에서는 확장만 하면 됩니다(모든 것이 이미 배포되어 실행 중임). RTO 및 RPO 요구 사항을 검토하여 두 방식 중에서 선택할 수 있습니다.

AWS 서비스

백업 및 복원 (p. 13)과 파일럿 라이트 (p. 16)가 적용되는 모든 AWS 서비스는 데이터 백업, 데이터 복제, 활성/대기 트래픽 라우팅, EC2 인스턴스를 포함한 인프라 배포를 위해 웜 스탠바이 모드에서도 사용됩니다.

AWS Auto Scaling은 AWS 리전 내에서 Amazon EC2 인스턴스, Amazon ECS 작업, Amazon DynamoDB 처리량, Amazon Aurora 복제본을 포함한 리소스를 확장하는 데 사용됩니다. Amazon EC2 Auto Scaling은 AWS 리전 내의 가용 영역 전체에 EC2 인스턴스 배포를 확장하여 해당 리전 내에서 복원성을 제공합니다. 파일럿 라이트 또는 웜 스탠바이 전략의 일환으로 Auto Scaling을 사용하여 DR 리전을 전체 프로덕션 기능으로 확장할 수 있습니다. 예를 들어 EC2의 경우 Auto Scaling 그룹에서 원하는 용량 설정을 늘립니다. AWS Management Console을 통해 수동으로 이 설정을 조정하거나, AWS SDK를 통해 자동으로 조정하거나, 원하는 새 용량 값으로 AWS CloudFormation 템플릿을 재배포하여 조정할 수 있습니다. AWS CloudFormation 파라미터를 사용하여 CloudFormation 템플릿을 더 쉽게 재배포할 수 있습니다. 프로덕션 용량으로 확장되는 것을 제한하지 않도록 DR 리전의 서비스 할당량이 충분히 높게 설정되어 있는지 확인하는 것이 좋습니다.

다중 사이트 활성/활성

다중 사이트 활성/활성 또는 상시 대기 활성/활성 전략의 일부로 워크로드를 여러 리전에서 동시에 실행할 수 있습니다. 다중 사이트 활성/활성 방식에서는 배포된 모든 리전의 트래픽을 처리하는 반면, 상시 대기 전략에서는 단일 리전의 트래픽만 처리하고 다른 리전은 재해 복구에만 사용됩니다. 다중 사이트 활성/활성 방식을 통해 사용자는 워크로드가 배포된 모든 리전에서 워크로드에 액세스할 수 있습니다. 이 방법은 가장 복잡하고 비용이 많이 드는 재해 복구 방식이지만 올바른 기술 선택 및 구현을 통해 대부분의 재해에 대한 복구 시간을 거의 제로에 가깝게 줄일 수 있습니다. 그러나 데이터 손상은 백업에 의존해야 하므로 일반적으로 복구 시점은 0이 아닙니다. 상시 대기는 사용자가 단일 리전으로만 연결되고 DR 리전은 트래픽을 받지 않는 활성/비활성 구성을 사용합니다. 대부분의 고객은 두 번째 리전에서 전체 환경을 구축하려는 경우 활성/활성 환경을 사용하는 것이 합리적이라는 것을 알게 됩니다. 또는 사용자 트래픽을 처리하는 데 두 리전을 모두 사용하지 않으려는 경우 웜 스탠바이는 경제적이며 운영 면에서 덜 복잡한 방식을 제공합니다.

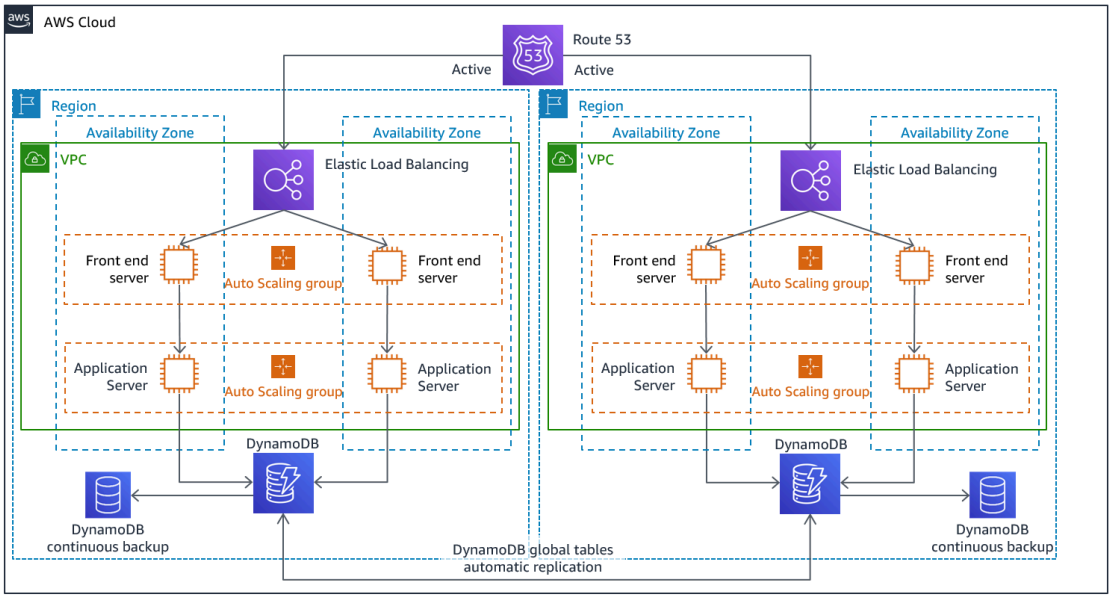

그림 12 - 다중 사이트 활성/활성 아키텍처(상시 대기의 경우 활성 경로 하나를 비활성으로 변경)

다중 사이트 활성/활성의 경우 워크로드가 둘 이상의 리전에서 실행 중이므로 이 시나리오에서는 장애 조치와 같은 상황이 없습니다. 이 경우 재해 복구 테스트는 워크로드가 리전의 손실에 어떻게 반응하는지에 초점을 맞춥니다. 가령 장애가 발생한 리전에서 먼 곳으로 트래픽이 라우팅되는지와 같은 사항에 중점을 둡니다. 다른 리전에서 모든 트래픽을 처리할 수 있는지와 같은 사항도 고려합니다. 데이터 재해에 대한 테스트도 필요합니다. 백업 및 복구는 여전히 필요하며 정기적으로 테스트해야 합니다. 또한 데이터 손상, 삭제 또는 난독화와 관련된 데이터 재해의 복구 시간은 항상 0보다 길며 복구 지점은 항상 재해가 발견되기 전의 특정 시점입니다. 복구 시간을 0에 가깝게 유지하기 위해 다중 사이트 활성/활성(또는 상시 대기) 방식에 추가적인 복잡성과 비용이 필요한 경우 보안을 유지하고 인적 오류를 방지하여 인적 재해를 완화하기 위해 추가적인 노력을 기울여야 합니다.

AWS 서비스

백업 및 복원 (p. 13), 파일럿 라이트 (p. 16) 및 웜 스탠바이 (p. 19)가 적용되는 모든 AWS 서비스는 특정 시점으로 데이터 백업, 데이터 복제, 활성/활성 트래픽 라우팅, EC2 인스턴스를 포함한 인프라 배포 및 확장을 위해 다중 사이트 활성/활성 모드에서도 사용됩니다.

앞에서 설명한 활성/비활성 시나리오(파일럿 라이트 및 웜 스탠바이)의 경우 네트워크 트래픽을 활성 리전으로 라우팅하는 데 Amazon Route 53과 AWS Global Accelerator를 모두 사용할 수 있습니다. 또한 활성/활성 전략의 경우 이 두 서비스는 어떤 사용자를 어떤 활성 리전 엔드포인트로 연결할지 결정하는 정책을 정의할

수 있습니다. AWS Global Accelerator를 사용하여 각 애플리케이션 엔드포인트로 향하는 트래픽 다이얼을 설정함으로써 트래픽의 비율을 제어할 수 있습니다. Amazon Route 53은 이 백분율 방식과 지리 근접성 및 대기 시간 기반 방식을 포함한 여러 다른 사용 가능한 정책도 지원합니다. Global Accelerator는 AWS 엣지 서버의 광범위한 네트워크를 자동으로 활용하여 가능한 한 빨리 트래픽을 AWS 네트워크 백본으로 온보딩하므로 요청 대기 시간이 단축됩니다.

이 전략을 사용한 데이터 복제는 0에 가까운 RPO를 가능하게 합니다. Amazon Aurora Global Database와 같은 AWS 서비스는 데이터베이스를 애플리케이션 지원에 전적으로 사용할 수 있는 전용 인프라를 사용하며, 일반적으로 1초 미만의 대기 시간으로 1개의 보조 리전에 복제할 수 있습니다. 활성/비활성 전략을 사용하면 기본 리전에만 쓰기가 발생합니다. 활성/활성와의 차이점은 각 활성 리전에 대한 쓰기가 처리되는 방식을 설계하는 데 있습니다. 사용자와 가장 가까운 리전에서 사용자 읽기가 처리되도록 설계하는 것이 일반적이며 이를 로컬 읽기라고 합니다. 쓰기 방식을 설계할 때는 다음과 같은 몇 가지 옵션이 있습니다.

- 글로벌 쓰기 전략은 모든 쓰기를 단일 리전으로 라우팅합니다. 해당 리전에서 장애가 발생할 경우 다른 리전이 승격되어 쓰기를 처리합니다. Aurora Global Database는 리전 전체에서 읽기 복제본과의 동기화를 지원하며 읽기/쓰기 작업을 처리하도록 보조 리전 중 하나를 1분 이내에 승격할 수 있으므로 글로벌 쓰기에 적합합니다.
- 로컬 쓰기 전략은 읽기와 마찬가지로 가장 가까운 리전으로 쓰기를 라우팅합니다. Amazon DynamoDB 글로벌 테이블은 이러한 전략을 지원하여 글로벌 테이블이 배포된 모든 리전에서 읽고 쓸 수 있도록 합니다. Amazon DynamoDB 글로벌 테이블은 동시 업데이트 간에 last writer wins 조정을 사용합니다.
- 쓰기 파티션 전략은 쓰기 충돌을 피하기 위해 파티션 키(예: 사용자 ID)를 기반으로 특정 리전에 쓰기를 할당합니다. 이 경우 양방향으로 구성된 Amazon S3 복제를 사용할 수 있으며 현재 두 리전 간 복제를 지원합니다. 이 방식을 구현할 때는 A와 B 두 버킷 모두에서 복제본 수정 동기화를 활성화하여 복제된 객체에 대한 객체 ACL(액세스 제어 목록), 객체 태그 또는 객체 잠금과 같은 복제본 메타데이터 변경 사항을 복제해야 합니다. 활성 리전의 버킷 간에 삭제 마커를 복제할지 여부도 구성할 수 있습니다. 복제 외에도 데이터 손상 또는 파괴 이벤트로부터 보호하기 위해 특정 시점으로 백업도 전략에 포함해야 합니다.

AWS CloudFormation은 여러 AWS 리전의 AWS 계정 간에 일관되게 배포된 인프라를 적용할 수 있는 강력한 도구입니다. AWS CloudFormation StackSets는 단일 작업으로 여러 계정과 리전에서 CloudFormation 스택을 생성, 업데이트 또는 삭제할 수 있도록 하여 이 기능을 확장합니다. AWS CloudFormation은 YAML 또는 JSON을 사용하여 코드형 인프라를 정의하지만 AWS Cloud Development Kit (AWS CDK)를 사용하면 익숙한 프로그래밍 언어로 코드형 인프라를 정의할 수 있습니다. 코드는 CloudFormation으로 변환된 후 AWS에서 리소스를 배포하는 데 사용됩니다.

탐지

워크로드가 제공해야 하는 비즈니스 성과를 제공하지 못할 경우 이를 가능한 한 빨리 파악하는 것이 중요합니다. 이렇게 하면 재해를 신속하게 선언하고 사고로부터 복구할 수 있습니다. 적극적인 복구 목표의 경우 이러한 대응 시간은 적절한 정보와 결합되어 복구 목표를 달성하는 데 매우 중요한 역할을 합니다. 복구 시점 목표가 1시간이면 사고를 탐지하고, 담당 직원에게 알리고, 에스컬레이션 프로세스를 수행하고, 복구 예상 시간에 대한 정보(있는 경우)를 평가하고(DR 계획은 실행하지 않음), 재해를 선언하고, 1시간 이내에 복구해야 합니다.

> Note
>
> RTO가 위험에 처하더라도 이해 관계자가 DR을 호출하지 않기로 결정할 경우 DR 계획과 목표를 재평가합니다. DR 계획을 호출하지 않기로 한 결정은 계획이 부적절하거나 실행에 대한 확신이 부족하기 때문일 수 있습니다.

비즈니스 가치를 제공하는 현실적이고 달성 가능한 목표를 제공하기 위해서는 사고 탐지, 알림, 에스컬레이션, 발견 및 선언을 계획 및 목표에 반영하는 것이 중요합니다.

AWS는 Service Health Dashboard에 서비스 가용성에 대한 최신 정보를 게시합니다. 언제든지 확인하여 최신 정보를 얻거나, RSS 피드를 구독하여 각 개별 서비스에 대한 중단 알림을 받을 수 있습니다. Service Health Dashboard에 표시되지 않는 서비스 중 하나에서 실시간 운영 문제가 발생하는 경우 지원 요청을 생성할 수 있습니다.

AWS Health Dashboard는 계정에 영향을 줄 수 있는 AWS Health 이벤트에 대한 정보를 제공합니다. 이 정보는 최근 이벤트와 예정된 이벤트를 범주별로 보여 주는 대시보드 및 지난 90일간의 모든 이벤트를 보여 주는 전체 이벤트 로그의 두 가지 방법으로 표시됩니다.

매우 엄격한 RTO 요구 사항이 있는 경우 상태 확인을 기반으로 자동 장애 조치를 구현할 수 있습니다. 사용자 경험을 대표하고 핵심 성과 지표를 기반으로 하는 상태 확인을 설계하세요. 심층 상태 확인은 워크로드의 주요 기능을 실행하며, 간단한 수준의 하트비트 검사 이상으로 검사합니다. 여러 신호를 기반으로 심층 상태 확인을 사용하세요. 필요가 없을 때 장애 조치를 수행하면 그 자체로 가용성 위험을 야기할 수 있으므로 이러한 방식에서는 잘못된 경보를 트리거하지 않도록 주의해야 합니다.

재해 복구 테스트

재해 복구 구현을 테스트하여 구현을 검증하고 워크로드 DR 리전으로의 장애 조치를 정기적으로 테스트하여 RTO 및 RPO가 충족되는지 확인합니다.

거의 실행되지 않는 복구 경로를 개발하는 것은 피해야 할 패턴입니다. 읽기 전용 쿼리에 사용되는 보조 데이터 스토어를 예로 들 수 있습니다. 데이터 스토어에 데이터를 쓸 때 기본 스토어에서 장애가 발생하면 보조 데이터 스토어로 장애 조치를 진행할 수 있습니다. 이 장애 조치를 자주 테스트하지 않으면 보조 데이터 스토어의 기능에 대한 가정이 잘못될 수 있습니다. 예를 들어 마지막으로 테스트했을 때는 보조 용량이 충분했지만 이 시나리오에서는 더 이상 로드를 모두 처리하지 못하거나 보조 리전의 서비스 할당량이 충분하지 않을 수 있습니다.

경험에 따르면 자주 테스트하는 경로만이 유일하게 효과가 있는 오류 복구 방법입니다. 이러한 이유로 인해 복구 경로를 적게 갖는 것이 가장 좋습니다.

복구 패턴을 설정하고 정기적으로 테스트할 수 있습니다. 복잡하거나 중요한 복구 경로가 있는 경우 해당 복구 경로의 작동을 검증하기 위해 프로덕션 환경에서 해당 장애를 정기적으로 실행해야 합니다.

DR 리전에서 구성 드리프트 관리 DR 리전에 필요한 인프라, 데이터 및 구성이 갖추어져 있는지 확인합니다. 예를 들어 AMI와 서비스 할당량이 최신 상태인지 확인합니다.

AWS Config를 사용하여 AWS 리소스 구성을 지속적으로 모니터링하고 기록할 수 있습니다. AWS Config는 드리프트를 탐지하고 AWS Systems Manager Automation을 트리거하여 드리프트를 수정하고 경보를 발생시킬 수 있습니다. AWS CloudFormation은 사용자가 배포한 스택의 드리프트를 추가로 탐지할 수 있습니다.

결론

고객은 클라우드에서 애플리케이션의 가용성에 대한 책임이 있습니다. 재해가 무엇인지 정의하고 이러한 정의와 재해가 비즈니스 성과에 미칠 수 있는 영향을 반영하는 재해 복구 계획을 수립하는 것이 중요합니다. 영향 분석 및 위험 평가를 기반으로 복구 시간 목표(RTO) 및 복구 시점 목표(RPO)를 수립한 후 재해에 대비할 수 있는 적절한 아키텍처를 선택하세요. 재해를 적시에 탐지할 수 있는지 확인하세요. 언제 목표가 위험에 처하는지 파악하는 것이 중요하기 때문입니다. 계획을 수립하고 테스트를 통해 계획을 검증하세요. 검증되지 않은 재해 복구 계획은 신뢰 부족이나 재해 복구 목표 달성 실패로 인해 구현되지 않을 위험이 있습니다.

기여자

이 문서를 작성하는 데 도움을 주신 분들입니다.

- Alex Livingstone, AWS Enterprise Support 클라우드 운영 실무 책임자
- Seth Eliot, Amazon Web Services 수석 안정성 솔루션스 아키텍트

추가 자료

자세한 내용은 다음 리소스를 참조하세요.

- 안정성 원칙 - AWS Well-Architected Framework
- 재해 복구 계획 체크리스트
- 상태 확인 구현
- AWS 솔루션 구현: Multi-Region Application Architecture
- AWS re:Invent 2018: 다중 리전 활성/활성 애플리케이션을 위한 아키텍처 패턴(ARC209-R2)

문서 기록

변경 사항	설명	날짜
첫 게시	최초 게시 날짜	2021년 2월 12일

이 백서의 업데이트에 대한 알림을 받으려면 RSS 피드를 구독하세요.

고지 사항

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