Redis용 Amazon ElastiCache

Redis용 ElastiCache 사용 설명서 API 버전 2015-02-02



Redis용 Amazon ElastiCache: Redis용 ElastiCache 사용 설명서

Copyright © 2020 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Redis용 ElastiCache란 무엇입니까?	1
일반적인 ElastiCache 사용 사례 및 ElastiCache 활용 방법	
인 메모리 데이터 스토어	2
게임 리더보드(Redis 정렬 세트)	
메시징(Redis Pub/Sub)	4
권장 데이터(Redis 해시)	
기타 Redis 사용	
ElastiCache 고객 추천사	
ElastiCache 리소스	
구성 요소 및 기능	
노드	
모드 Redis용 ElastiCache 샤드	
Redis용 ElastiCache 클러스터	
Redis용 ElastiCache 볼데스더	
AWS 리전 및 가용 영역	12
Redis용 ElastiCache 엔드포인트	
파라미터 그룹	12
Redis용_ElastiCache 보안	
보안 그룹	
서브넷 그룹	
Redis용 ElastiCache 백업	14
Events	14
Redis용 ElastiCache 용어	14
구현 관리용 도구	15
AWS Management 콘솔 사용	16
AWS CLI 사용	
AWS SDK 사용	
ElastiCache API 사용	
추가 리소스	
# 기 더보고	
Redis용 ElastiCache 시작하기	
Redises Elasticache 시작하기	
설정	
AWS 계정 생성	20
권한 설정(신규 ElastiCache 사용자에게만 해당)	. 21
1단계: 클러스터 시작	
2단계: 액세스 자격 부여	
3단계: 클러스터 노드에 연결	
3.1단계: 노드 엔드포인트 찾기	
3.2단계: Redis 클러스터 또는 복제 그룹에 연결(Linux)	. 24
3.2단계: Redis 클러스터 또는 복제 그룹에 연결(Windows)	. 25
4단계: 클러스터 삭제	26
추가 정보	
자습서 및 동영상	_
동영상	
입문자용 동영상	
고급 동영상	
고급 등당당 ElastiCache로 온라인 마이그레이션	
개요	
마이그레이션 단계	
마이그레이션을 위한 소스 및 대상 Redis 노드 준비	
마이그레이션 시작	
데이터 마이그레이션 진행 상황 확인	
데이터 마이그레이션 완료	33

	을 사용해 온라인 데이터 마이그레이션 수행	
	lastiCache 구현 관리	
엔진	버전 및 업그레이드	. 37
	지원되는 Redis 버전	. 37
	엔진 버전 업그레이드	. 45
리전	및 가용 영역 선택	
—	지원되는 리전 및 엔드포인트	47
	노드 찾기	
유지	관리 관리 중	
	- 관리	
	Redis 노드 및 샤드	
	노드에 연결	
	예약 노드	
	지원되는 노드 유형	
	노드 교체	
Elas	tiCache 클러스터 관리	
	지원되는 Redis 버전	. 64
	기타 ElastiCache 클러스터 작업	. 65
	클러스터 생성	. 65
	클러스터 세부 정보 보기	. 78
	클러스터 수정	
	클러스터 재부팅	
	클러스터에 노드 추가	
	클러스터에서 노드 제거	
	대기 중인 노드 추가 또는 삭제 작업 취소	101
	클러스터 삭제하기	101
= -1	글더스더 국제하기스터 또는 복제 그룹에 액세스	
글더		
	클러스터의 플랫폼 확인	104
	클러스터 또는 복제 그룹에 액세스 권한 부여	
샤드		
	샤드 ID를 검색하는 방법	112
글로	벌 데이터 스토어를 사용한 AWS 리전 간 복제	
	개요	
	사전 조건 및 제한 사항	
	글로벌 데이터 스토리지 사용(콘솔)	117
	글로벌 데이터 저장소(CLI) 사용	125
고가	용성을 위한 복제 그룹 사용	
	Redis 복제 이해	
	복제: Redis(클러스터 모드 비활성화됨) 대 Redis(클러스터 모드 활성화됨)	130
	작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ	132
	동기화 및 백업 구현 방법	
	복제 그룹 생성	
	복제 그룹의 세부 정보 보기	
	복제 그룹 엔드포인트 확인	
	복제 그룹 수정	
	복제 그룹 삭제	
	복제본 수 변경	
0.17	위기 전용 복제본 승격	
연결	엔드포인트 찾기	
	Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기(콘솔)	181
	Redis(클러스터 모드 활성화됨) 엔드포인트(콘솔)의 엔드포인트 찾기	
	엔드포인트 찾기 (AWS CLI)	
	엔드포인트 찾기 (ElastiCache API)	186
백업	및 복원	187
	Constraints	188
	비용	
	백업의 성능 영향	

	지도 배어 제어	400
	자동 백업 예약	
	수동 백업 만들기	190
	최종 백업 생성	
	백업 설명	198
	백업 복사	199
	백업 내보내기	
	백업에서 복원	208
	백업으로 클러스터 시드	211
	백업 태그 지정	
	백업 삭제	
	Redis용 ElastiCache의 AOF(Append Only Files)	218
	Redis용 ElastiCache 클러스터 조정	
	Neuiso Liasiloadile aujumi 140	. 213
	Redis(클러스터 모드 비활성화됨)에 대한 클러스터 조정	
	Redis(클러스터 모드 활성화됨)에서 클러스터 조정	241
	파라미터 그룹을 사용해 엔진 파라미터 구성	260
	파라미터 관리	
	파라미터 그룹 티어	261
	파라미터 그룹 생성	262
	이름별로 파라미터 그룹 목록 조회	
	파라미터 그룹의 값 목록 조회	
	파라미터 그룹 수정	271
	파라미터 그룹 삭제	
	Redis 특정 파라미터	
보안		298
	데이터 보호	208
	A Flatto-the Oldby Hol	200
	Amazon ElastiCache의 데이터 보안	
	인터네트워크 트래픽 개인 정보	315
	Amazon VPC 및 ElastiCache 보안	
	서브넷 및 서브넷 그룹	
	보안 그룹: EC2-Classic	
	ID 및 액세스 관리	341
	인증	
	액세스 제어	
	액세스 관리 개요	. 342
	자격 증명 기반 정책(IAM 정책) 사용	345
	서비스 연결 역할 사용	
	ElastiCache API 권한 참조	355
	Elasticache에서 로깅 및 모니터링	359
	모니터링 사용	
	이벤트 모니터링	
	태그를 사용한 비용 모니터링	375
	예약 노드를 통해 비용 관리	
	규정 준수 확인	
	Redis용 ElastiCache 규정 준수	391
	복원성	395
	장애 완화	
	인프라 보안	
캐싱	전략 및 모범 사례	399
	지연 로딩	
	라이트-스루	401
	TTL 추가	
	관련 주제	
	제한된 Redis 명령	
	충분한 메모리를 확보하여 Redis 스냅샷 생성	404
	백그라운드 쓰기 프로세스 및 메모리 사용량	
	백그라운드 쓰기를 실행할 때 메모리 부족 방지	405

	세야토 메드 그 교니	405
,	예약된 메모리 관리	405
	필요한 예약된 메모리의 양	405
	예약된 메모리를 관리하기 위한 파라미터	406
	예약된 메모리 관리 파라미터 지정	408
	Redis AOF를 사용할 때 디스크 공간 부족 문제 완화	409
	내결함성 향상을 위한 Redis 다중 AZ 활성화	
	모범 사례: 온라인 클러스터 크기 조정	
	모범 사례: 유지 관리 중 가동 중지 최소화	
	서비스 업데이트	
	서비스 업데이트 관리	
	셀프 서비스 업데이트 적용	
	셀프 서비스 업데이트 중지	
참조 .		
	ElastiCache API 사용	
	Query API 사용	
	사용 가능한 라이브러리	
	애플리케이션 문제 해결	
	AWS CloudTrail을 사용하여 Amazon ElastiCache API 호출 로깅	
	ElastiCache용 AWS CLI 설정	
!	사전 요구사항	
	명령행 도구 얻기	
	도구 설정	
	도구에 대한 자격 증명 제공	
	포기에 대한 자극 8.8 제 8 환경 변수	
	오류 메시지	
	알림	
	일반 ElastiCache 알림	
D!:	Redis용 ElastiCache 관련 알림	
	용 ElastiCache 설명서 기록	
AWS	Glossary	442

Redis용 Amazon ElastiCache란 무엇입니까?

Redis용 Amazon ElastiCache 사용 설명서에 오신 것을 환영합니다. Amazon ElastiCache는 클라우드에서 분산된 인 메모리 데이터 스토어 또는 캐시 환경을 손쉽게 설정, 관리 및 확장할 수 있는 웹 서비스입니다. 확장 가능하고 비용 효율적인 고성능 캐싱 솔루션을 제공합니다. 또한 분산된 캐시 환경의 배포 및 관리와 관련된 복잡성을 해소할 수 있습니다.

Note

Amazon ElastiCache는 Redis 엔진 및 Memcached 엔진 모두와 함께 작동합니다. 관심 있는 엔진에 대해 설명한 가이드를 사용하십시오. 사용하고 싶은 엔진을 결정하기 어렵다면 이 가이드의 Memcached와 Redis 비교 (p. 17) 단원을 참조하십시오.

Redis를 사용하는 기존 애플리케이션은 거의 수정하지 않고 ElastiCache를 사용할 수 있습니다. 애플리케이션에서는 사용자가 배포한 ElastiCache 노드의 호스트 이름과 포트 번호에 관한 정보만 필요합니다.

Redis용 ElastiCache는 중요한 프로덕션 배포를 위해 서비스를 보다 안정적으로 제공하는 여러 가지 기능을 갖추고 있습니다.

- 캐시 노드 실패에서 자동 감지 및 복구를 합니다.
- 복제를 지원하는 Redis 클러스터에 있는 읽기 전용 복제본으로 실패한 기본 클러스터를 자동 장애 조치하는 다중 AZ입니다.
- Redis(클러스터 모드 활성화됨)는 최대 90개 샤드로의 데이터 분할을 지원합니다.
- Redis 버전 3.2 이상에서는 모든 버전이 전송 중 데이터 암호화와 인증으로 미사용 데이터 암호화를 지원합니다. 이 지원을 통해 HIPAA 규정 준수 애플리케이션을 구축할 수 있습니다.
- 내결함성 향상을 위한 노드 및 클러스터의 유연한 가용 영역 배치.
- Amazon EC2, Amazon CloudWatch, AWS CloudTrail, Amazon SNS 등의 AWS 서비스와 통합 이러한 통합을 통해 고성능과 높은 보안성을 갖춘 관리되는 인메모리 캐싱 솔루션을 제공합니다.

주제

- 일반적인 ElastiCache 사용 사례 및 ElastiCache 활용 방법 (p. 1)
- Amazon ElastiCache 리소스 (p. 7)
- Redis용 ElastiCache 구성 요소 및 기능 (p. 8)
- Redis용 ElastiCache 용어 (p. 14)
- 구현 관리용 도구 (p. 15)

일반적인 ElastiCache 사용 사례 및 ElastiCache 활용 방법

최신 뉴스, 10대 리더보드, 제품 카탈로그를 게재할 때나 이벤트 티켓을 판매할 때 가장 중요한 것은 속도입니다. 웹 사이트와 비즈니스의 성공 여부는 콘텐츠를 제공하는 속도에 상당한 영향을 받습니다.

뉴욕 타임즈의 "For Impatient Web Users, an Eye Blink Is Just Too Long to Wait(참을성 없는 웹 사용자에게는 눈 깜박하는 시간조차 너무 길게 느껴져)"라는 기사에 따르면 사용자는 경쟁 사이트 간의 250밀리

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 인 메모리 데이터 스토어

초(4/4초) 차이를 인지합니다. 사용자는 결국 속도가 느린 사이트를 떠나 빠른 사이트로 이동합니다. How Webpage Load Time Is Related to Visitor Loss에 따르면 Amazon에서 실시한 테스트에서 로드 시간이 100 밀리초(10/10초) 증가할 때마다 매출이 1% 감소하는 결과가 나왔습니다.

다른 사용자가 데이터를 원할 경우 캐시된 데이터를 훨씬 더 빠르게 제공할 수 있습니다. 웹 페이지든 비즈니스 결정을 주도하는 보고서용이든 이는 동일하게 적용되는 사실입니다. 귀사에서는 웹 페이지를 캐시하지 않고 지연 시간을 최소화하여 제공할 여유가 있습니까?

가장 많이 요청되는 항목을 캐시하려 할 것이라는 사실은 어쩌면 자명한 것일 수 있습니다. 요청 빈도가 낮은 항목을 캐시하려고 하지 않는 이유는 무엇입니까? 가장 최적화된 데이터베이스 쿼리나 원격 API 호출이라 할지라도 인 메모리 캐시에서 플랫 키를 검색하는 것보다 현저히 느려집니다. 현저히 느려지면 고객이 다른 곳으로 떠나는 경향이 있습니다.

다음 예제에서는 ElastiCache를 사용하여 애플리케이션의 전반적인 성능을 개선할 수 있는 몇 가지 방법을 보여줍니다.

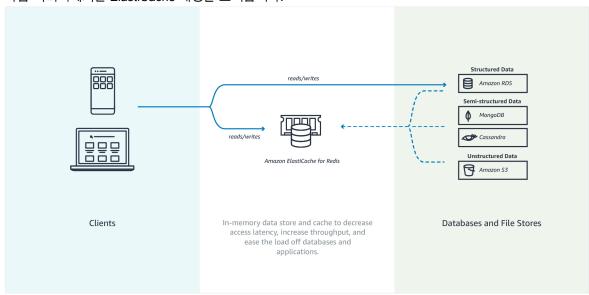
주제

- 인 메모리 데이터 스토어 (p. 2)
- 게임 리더보드(Redis 정렬 세트) (p. 3)
- 메시징(Redis Pub/Sub) (p. 4)
- 권장 데이터(Redis 해시) (p. 6)
- 기타 Redis 사용 (p. 7)
- ElastiCache 고객 추천사 (p. 7)

인 메모리 데이터 스토어

인 메모리 키-값 스토어의 주된 목적은 지연 시간이 1밀리초 미만인 매우 빠른 속도와 저렴한 비용으로 데이터 복사본에 액세스할 수 있게 하는 것입니다. 대부분의 데이터 스토어에는 자주 액세스하지만 거의 업데이트하지 않는 데이터 영역이 있습니다. 또한 데이터베이스를 쿼리하면 키-값 페어 캐시에서 키를 찾는 것에 비해 확실히 속도가 느리고 비용이 많이 듭니다. 일부 데이터베이스 쿼리는 특히 수행하는 데 있어 많은 비용이듭니다. 예로는 여러 표에 걸친 조인이나 복잡한 계산이 포함된 쿼리를 들 수 있습니다. 이러한 쿼리 결과를 캐시하여 쿼리 가격을 한 번만 지불합니다. 그런 다음 쿼리를 다시 실행할 필요 없이 여러 번 데이터를 빠르게 검색할 수 있습니다.

다음 이미지에서는 ElastiCache 캐싱을 보여줍니다.



어떻게 캐시해야 합니까?

캐시할 데이터를 결정할 때 다음과 같은 요인을 고려합니다.

속도 및 비용 – 캐시가 아닌 데이터베이스에서 데이터를 얻는 것이 항상 더 느리고 비용도 많이 듭니다. 다른데이터베이스 쿼리에 비해 원래 느리고 비용이 많이 드는 데이터베이스 쿼리도 있습니다. 예를 들어, 여러 테이블에서 조인을 수행하는 쿼리는 간단한 싱글 테이블 쿼리에 비해 훨씬 더 느리고 비용이 많이 소요됩니다. 속도가 느리고 비용이 많이 드는 쿼리를 통해서만 얻을 수 있는 데이터라면 캐시가 필요합니다. 비교적 빠르고 간단한 쿼리로 얻을 수 있는 데이터라도 다른 요인에 따라 캐싱이 필요할 수 있습니다.

데이터 및 액세스 패턴 – 캐시할 항목을 결정할 때는 데이터 자체와 액세스 패턴을 이해해야 합니다. 예를 들어, 빠르게 변하거나 거의 액세스하지 않는 데이터는 캐시할 필요가 없습니다. 캐시를 통해 실질적인 이점을 누리려면 비교적 정적이고 자주 액세스하는 데이터여야 합니다. 소셜 미디어 사이트의 개인 프로필을 예로들 수 있습니다. 반대로, 캐시해도 속도나 비용이 나아지지 않는다면 데이터를 캐시할 필요가 없습니다. 예를들어 검색 결과를 반환하는 웹 페이지의 쿼리와 결과는 대부분 고유하기 때문에 그러한 웹 페이지를 캐시하는 것은 의미가 없습니다.

기한 경과 – 정의하자면 캐시된 데이터는 기한이 경과한 데이터입니다. 경우에 따라 데이터 기한이 지나지 않았더라도 언제나 기한이 지난 데이터로 간주하고 취급해야 합니다. 데이터를 캐시할지 여부를 결정할 때는 애플리케이션의 데이터 기한 경과 허용 범위를 확인해야 합니다.

애플리케이션에서 기한이 지난 데이터를 허용할 수 있는 상황도 있고 그렇지 않은 상황도 있습니다. 예를 들어 사이트에서 공개적으로 거래된 주가를 제공한다고 가정합니다. 고객이 가격이 n분 지연될 수 있다는 고지사항과 함께 어느 정도의 기한이 경과할 수 있음을 받아들일 수 있습니다. 하지만 주식을 매각하거나 매입하는 중개인에게 주식의 가격을 제공할 때는 실시간 데이터가 필요합니다.

다음에 해당하는 경우 데이터 캐시를 고려하십시오.

- 캐시 검색에 비해 느린 속도와 높은 비용으로 데이터를 얻습니다.
- 사용자가 자주 데이터에 액세스합니다.
- 데이터가 비교적 동일하게 유지되거나 빠르게 변경되어도 기한 경과가 큰 문제가 되지 않습니다.

자세한 내용은 다음 자료를 참조하십시오.

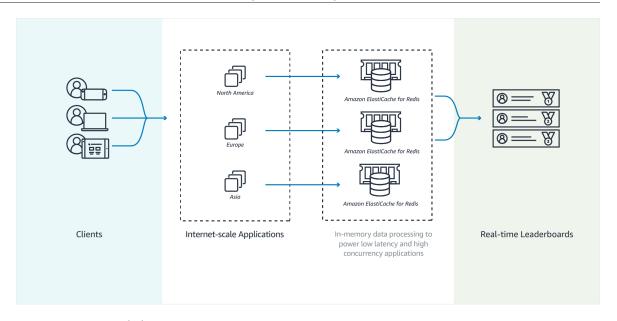
• Redis용 ElastiCache 사용 설명서의 캐싱 전략

게임 리더보드(Redis 정렬 세트)

Redis 정렬 세트는 리더보드의 컴퓨팅 복잡성을 애플리케이션에서 Redis 클러스터로 옮깁니다.

게임의 상위 10개 점수와 같은 리더보드는 계산적으로 복잡합니다. 이는 특히 동시 플레이어가 많고 점수가 지속적으로 바뀌는 경우에 그렇습니다. Redis 정렬 세트는 고유성 및 요소 순서를 둘 다 보장합니다. Redis 정렬 세트를 사용하면 정렬 세트에 새 요소가 추가될 때마다 실시간으로 순위를 다시 매깁니다. 그다음에 올바른 숫자 순서의 세트에 해당 요소가 추가됩니다.

다음 다이어그램을 통해 Redis용 ElastiCache 게임 리더보드가 어떻게 작동하는지 알 수 있습니다.



Example - Redis 리더보드

이 예제에서는 ZADD를 사용하여 게이머 4명과 이들의 점수를 정렬 목록에 입력합니다. ZREVRANGEBYSCORE 명령이 높은 점수에서 낮은 점수 순서로 플레이어를 나열합니다. 그런 다음 ZADD를 사용해 기존의 항목을 덮어써 June의 점수를 업데이트합니다. 마지막으로, ZREVRANGEBYSCORE에서 높은 점수에서 낮은 점수 순서로 플레이어를 나열합니다. 이 목록에서는 June이 순위가 상승했음을 알 수 있습니다.

```
ZADD leaderboard 132 Robert
ZADD leaderboard 231 Sandra
ZADD leaderboard 32 June
ZADD leaderboard 381 Adam

ZREVRANGEBYSCORE leaderboard +inf -inf
1) Adam
2) Sandra
3) Robert
4) June

ZADD leaderboard 232 June

ZREVRANGEBYSCORE leaderboard +inf -inf
1) Adam
2) June
3) Sandra
4) Robert
```

다음 명령을 통해 June은 모든 플레이어 중 자신의 순위를 알 수 있습니다. 순위는 0부터 시작하므로 ZREVRANK가 두 번째 위치에 있는 June에 대해 1을 반환합니다.

```
ZREVRANK leaderboard June
1
```

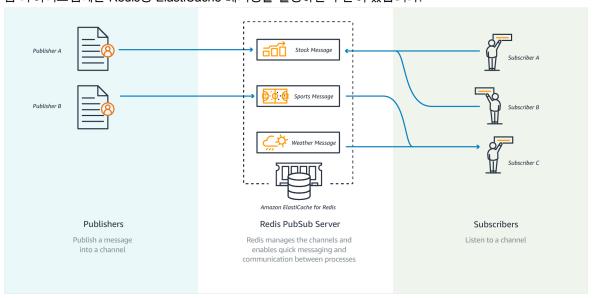
자세한 내용은 정렬 세트에 대한 Redis 설명서를 참조하십시오.

메시징(Redis Pub/Sub)

이메일 메시지를 보낼 때 지정된 수신자 한 명 이상에게 메시지가 전송됩니다. pub/sub 패러다임에서는 메시지를 받을 사람을 모르는 상태에서 특정 채널로 메시지를 보냅니다. 메시지를 받는 사람은 채널을 구독하는

사람입니다. 예를 들어 news.sports.golf 채널을 구독한다고 가정해 보겠습니다. news.sports.golf 채널을 구 독하는 모든 사람이 news.sports.golf에 게시되는 메시지를 받게 됩니다.

Redis pub/sub 기능은 키 스페이스와 아무 관계가 없으므로 어떤 수준에서도 간섭이 발생하지 않습니다. 다음 다이어그램에는 Redis용 ElastiCache 메시징을 설명하는 부분이 있습니다.



구독

채널의 메시지를 받으려면 해당 채널을 구독합니다. 단일 채널, 지정된 여러 채널 또는 패턴에 맞는 모든 채널을 구독할 수 있습니다. 구독을 취소하려면 구독할 때 지정한 채널에서 구독을 취소해야 합니다. 또는 패턴일치를 사용하여 구독한 경우 이전에 사용했던 패턴과 동일한 패턴으로 구독을 취소합니다.

Example - 단일 채널 구독

단일 채널을 구독하려면 구독할 채널을 지정하는 SUBSCRIBE 명령을 사용하십시오. 다음 예제에서는 클라 이언트가 news.sports.golf 채널을 구독합니다.

SUBSCRIBE news.sports.golf

얼마 후 클라이언트가 구독을 취소할 채널을 지정하는 UNSUBSCRIBE 명령을 사용하여 채널 구독을 취소합 니다.

UNSUBSCRIBE news.sports.golf

Example - 지정된 여러 채널 구독

특정한 여러 채널을 구독하려면 SUBSCRIBE 명령으로 채널을 나열하십시오. 다음 예제에서는 클라이언트 가 news.sports.golf, news.sports.soccer 및 news.sports.skiing 채널을 구독합니다.

SUBSCRIBE news.sports.golf news.sports.soccer news.sports.skiing

특정 채널 구독을 취소하려면 UNSUBSCRIBE 명령을 사용하고 구독을 취소할 채널을 지정합니다.

UNSUBSCRIBE news.sports.golf

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 권장 데이터(Redis 해시)

여러 채널의 구독을 취소하려면 UNSUBSCRIBE 명령을 사용하고 구독을 취소할 채널을 지정합니다.

UNSUBSCRIBE news.sports.golf news.sports.soccer

모든 구독을 취소하려면 UNSUBSCRIBE를 사용하고 각 채널을 지정합니다. 또는 UNSUBSCRIBE를 사용하고 채널을 지정하지 않습니다.

UNSUBSCRIBE news.sports.golf news.sports.soccer news.sports.skiing

또는

UNSUBSCRIBE

Example - 패턴 일치를 사용하는 구독

클라이언트는 PSUBSCRIBE 명령을 사용하여 패턴과 일치하는 모든 채널을 구독할 수 있습니다.

다음 예제에서는 클라이언트가 모든 스포츠 채널을 구독합니다. SUBSCRIBE를 사용할 때처럼 모든 스포츠 채널을 개별적으로 나열하는 것은 아닙니다. 대신 PSUBSCRIBE 명령을 사용하여 패턴 일치를 사용합니다.

PSUBSCRIBE news.sports.*

Example 구독 취소

이 채널의 구독을 취소하려면 PUNSUBSCRIBE 명령을 사용하십시오.

PUNSUBSCRIBE news.sports.*

Important

[P]SUBSCRIBE 명령과 [P]UNSUBSCRIBE 명령에 보낸 채널 문자열이 일치해야 합니다. news.*에 대한, 그리고 PUNSUBSCRIBE news.sports.* 또는 UNSUBSCRIBE news.sports.golf에서의 PSUBSCRIBE가 불가능합니다.

게시

채널의 모든 구독자에게 메시지를 보내려면 채널과 메시지를 지정하는 PUBLISH 명령을 사용하십시오. 다음 예제에서는 "It's Saturday and sunny. I'm headed to the links."라는 메시지를 news.sports.golf 채널에 게시합 니다.

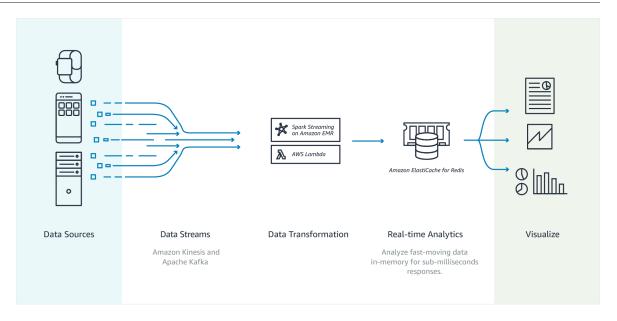
PUBLISH news.sports.golf "It's Saturday and sunny. I'm headed to the links."

클라이언트는 구독한 채널에 게시할 수 없습니다.

자세한 내용은 Redis 설명서의 Pub/Sub을 참조하십시오.

권장 데이터(Redis 해시)

Redis의 INCR 또는 DECR을 사용하면 컴파일 권장 사항이 단순화됩니다. 사용자가 제품을 "좋아할" 때마다 item:productID:like 카운터가 증가하고 "싫어할" 때마다 item:productID:dislike 카운터가 증가합니다. Redis 해시를 사용하여 제품을 좋아하거나 싫어한 모든 사람의 목록을 보존할 수도 있습니다. 다음 다이어그램에서는 Redis용 ElastiCache 실시간 분석 스토어를 보여줍니다.



Example - 호불호

```
INCR item:38923:likes
HSET item:38923:ratings Susan 1
INCR item:38923:dislikes
HSET item:38923:ratings Tommy -1
```

기타 Redis 사용

Salvatore Sanfilippo의 블로그 게시글(How to take advantage of Redis just adding it to your stack(단순히 Redis를 스택에 추가하여 Redis를 이용하는 방법))에서는 여러 가지 일반적인 데이터베이스 문제와 이를 Redis를 사용해 쉽게 해결하는 방법을 설명합니다. 이 접근 방식을 사용하면 데이터베이스에서 로드할 필요가 없어 성능이 향상됩니다.

ElastiCache 고객 추천사

Airbnb, PBS, Esri 등의 기업이 어떻게 Amazon ElastiCache을 활용해 고객 환경을 개선하여 비즈니스 성장을 도모하고 있는지에 관해서는 추천사Amazon ElastiCache를 참조하십시오.

다른 ElastiCache 고객 사용 사례는 ElastiCache 동영상 (p. 28)에서 보실 수 있습니다.

Amazon ElastiCache 리소스

먼저 다음의 단원을 읽은 다음, 필요에 따라 참조하는 것이 좋습니다.

- 서비스 하이라이트 및 요금 제품 세부 정보 페이지는 일반적인 ElastiCache 제품 개요, 서비스 하이라이 트 및 요금 정보를 제공합니다.
- ElastiCache 동영상 ElastiCache 동영상 (p. 28) 섹션에는 Amazon ElastiCache를 소개하는 동영상이 있습니다. 이 동영상에서는 ElastiCache에 대한 일반 사용 사례를 다루며, ElastiCache를 사용하여 지연 시간을 줄이고 애플리케이션의 처리량을 향상하는 방법을 설명합니다.
- 시작하기 Redis용 Amazon ElastiCache 시작하기 (p. 20) 섹션에는 캐시 클러스터 생성 프로세스를 안내하는 예가 포함되어 있습니다. 이 예제에는 캐시 클러스터에 대한 액세스할 수 있는 권한을 부여하는 방법과 캐시 노드에 연결하고 캐시 클러스터를 삭제하는 방법이 포함되어 있습니다.

• 규모에 따른 성능 – Amazon ElastiCache의 규모에 따른 성능 백서에서는 규모에 따라 애플리케이션이 원활하게 수행할 수 있는 캐싱 전략에 대해 설명합니다.

이전 단원을 완료한 후에는 다음 단원을 읽어 보십시오.

• 노드 크기 선택 (p. 67)

캐시하려는 모든 데이터를 수용할 만큼의 충분한 노드를 원하며, 동시에 필요한 것보다 더 많은 캐시의 비용을 지불하고 싶지 않은 경우 이 항목을 사용하여 최적의 노드 크기를 선택할 수 있습니다.

• 캐싱 전략 및 모범 사례 (p. 399)

클러스터의 효율성에 영향을 줄 수 있는 문제를 확인하고 해결하십시오.

AWS Command Line Interface (AWS CLI)를 사용할 경우 시작하는 데 도움이 되도록 다음 문서를 사용할 수 있습니다.

• AWS Command Line Interface 설명서

이 섹션에서는 AWS CLI 다운로드, 시스템에서 작동 중인 AWS CLI 가져오기 및 AWS 자격 증명 제공에 대한 정보를 제공합니다.

• AWS CLI ElastiCache용 설명서

이 별도의 문서는 구문 및 예제를 포함한 모든 ElastiCache용 AWS CLI 명령를 다룹니다.

ElastiCache API를 널리 사용되는 다양한 프로그래밍 언어로 사용하도록 애플리케이션 프로그램을 작성할 수 있습니다. 다음과 같은 몇 가지 리소스가 있습니다.

• Amazon Web Services용 도구

Amazon Web Services은 ElastiCache에 대한 지원과 더불어 다양한 SDK(소프트웨어 개발 키트)를 제공합니다. Java, .NET, PHP, Ruby 및 기타 언어를 사용하여 ElastiCache에 대해 코딩할 수 있습니다. 이러한 SDK는 ElastiCache에 대한 요청을 포맷하고 응답을 분석하며 재시도 논리 및 오류 처리를 제공함으로써 애플리케이션 개발을 상당히 간소화할 수 있습니다.

• ElastiCache API 사용 (p. 421)

AWS SDK를 사용하지 않으려는 경우 Query API를 직접 사용하면서 ElastiCache와 상호 작용할 수 있습니다. 이 단원에서 요청 생성과 인증 및 응답 처리에 대한 정보와 문제 해결 팁을 찾을 수 있습니다.

Amazon ElastiCache API Reference

이 별도의 문서는 구문 및 예제를 포함한 모든 ElastiCache API 작업을 다룹니다.

Redis용 ElastiCache 구성 요소 및 기능

다음은 Amazon ElastiCache 배포의 주요 구성 요소를 개괄적으로 설명한 것입니다.

주제

- ElastiCache 노드 (p. 9)
- Redis용 ElastiCache 샤드 (p. 9)
- Redis용 ElastiCache 클러스터 (p. 9)
- Redis용 ElastiCache 복제 (p. 10)
- AWS 리전 및 가용 영역 (p. 12)
- Redis용 ElastiCache 엔드포인트 (p. 12)

노드

- ElastiCache 파라미터 그룹 (p. 12)
- Redis용 ElastiCache 보안 (p. 13)
- ElastiCache 보안 그룹 (p. 13)
- ElastiCache 서브넷 그룹 (p. 14)
- Redis용 ElastiCache 백업 (p. 14)
- ElastiCache 이벤트 (p. 14)

ElastiCache 노드

노드는 ElastiCache 배포의 가장 작은 구성 요소입니다. 노드는 다른 노드와 독립적으로 존재하거나 다른 노드와 일부 관련되어 존재할 수 있습니다.

노드는 안전한 네트워크에 연결된 RAM의 크기가 고정된 청크입니다. 각 노드는 클러스터를 생성할 때 선택한 엔진 및 버전의 인스턴스를 실행합니다. 필요한 경우 클러스터의 노드를 다른 인스턴스 유형으로 스케일업하거나 스케일 다운할 수 있습니다. 자세한 내용은 Redis용 ElastiCache 클러스터 조정 (p. 219) 단원을참조하십시오.

클러스터 내 모든 노드는 인스턴스 유형이 동일하며, 동일한 캐시 엔진을 실행합니다. 각 캐시 노드에는 고유한 DNS(Domain Name Service) 이름 및 포트가 있습니다. 여러 유형의 캐시 노드가 지원되며 연결된 메모리양이 각각 다릅니다. 지원되는 노드 인스턴스 유형의 목록은 지원되는 노드 유형 (p. 54) 단원을 참조하십시오.

사용한 노드에 대해서만 비용을 지불하는 "사용한 만큼 지불"을 기준으로 노드를 구입할 수 있습니다. 또는 대폭 인하된 시간당 요금으로 예약 노드를 구입할 수 있습니다. 사용률이 높은 경우, 예약 노드를 구입하면 비용을 절약할 수 있습니다. 클러스터가 항상 사용 중에 있고 사용량 폭증을 처리하기 위해 가끔 노드를 추가하는 경우를 생각해 봅시다. 이 경우, 대부분의 시간을 실행하기 위해 여러 개의 예약된 노드를 구입할 수 있습니다. 그런 다음 노드를 가끔 추가하는 경우에 대해 "사용한 만큼 지불"을 기준으로 노드를 구입할 수 있습니다. 예약 노드에 대한 자세한 내용은 ElastiCache 예약 노드 (p. 54) 단원을 참조하십시오.

노드에 대한 자세한 내용은 노드 관리 (p. 51) 단원을 참조하십시오.

Redis용 ElastiCache 샤드

Redis 샤드(API 및 CLI에서 노드 그룹이라고 함)는 1~ six 개의 관련 노드 그룹입니다. Redis(클러스터 모드 비활성화됨) 클러스터에는 항상 한 개의 샤드가 있습니다. Redis(클러스터 모드 활성화됨) 클러스터 하나에 는 1 ~ –90개의 샤드가 포함될 수 있습니다.

다중 노드 샤드는 읽기/쓰기 기본 노드 하나와 1 ~5 개의 복제본 노드를 통해 복제를 구현합니다. 자세한 내용은 고가용성을 위한 복제 그룹 사용 (p. 127) 단원을 참조하십시오.

샤드에 대한 자세한 내용은 샤드 작업 (p. 111) 단원을 참조하십시오.

Redis용 ElastiCache 클러스터

Redis 클러스터는 Redis용 ElastiCache 샤드 (p. 9) 하나 이상의 논리적 그룹입니다. 데이터는 Redis(클러스터 모드 활성화됨) 클러스터의 샤드 간에 파티셔닝됩니다.

여러 ElastiCache 작업은 클러스터에서 다음 사항을 대상으로 합니다.

- 클러스터 생성
- 클러스터 수정
- 클러스터의 스냅샷 생성(모든 Redis 버전)
- 클러스터 삭제
- 클러스터의 요소 보기

• 비용 할당 태그를 클러스터에 추가 및 클러스터에서 삭제

자세한 내용은 다음 관련 항목을 참조하십시오.

• ElastiCache 클러스터 관리 (p. 64) 및 노드 관리 (p. 51)

클러스터, 노드 및 관련 작업에 대한 정보입니다.

• AWS 서비스 제한: Amazon ElastiCache

노드 또는 클러스터의 최대 수와 같은 ElastiCache 제한에 대한 정보입니다. 이러한 제한을 초과해야 할 경우 Amazon ElastiCache캐시 노드 요청 양식을 사용하여 요청하십시오.

• 장애 완화 (p. 395)

클러스터 및 복제 그룹의 내결함성 향상에 대한 정보입니다.

일반적인 클러스터 구성

Redis 클러스터에는 1 ~ -90개의 샤드(API에서 노드 그룹이라고 함)가 포함되며, 각 샤드는 데이터의 파티션입니다. Redis(클러스터 모드 비활성화됨)에는 항상 한 개의 샤드만 있습니다.

다음은 일반적인 클러스터 구성입니다.

Redis 클러스터

Redis(클러스터 모드 활성화됨) 클러스터에는 1 ~ –90개의 샤드(API 및 CLI에서 노드 그룹이라고 함)가 포함됩니다. Redis(클러스터 모드 비활성화됨) 클러스터에는 항상 한 개의 샤드(API 및 CLI에서 노드 그룹 하나)만 있습니다. Redis 샤드에는 1 ~ six개의 노드가 있습니다. 한 샤드에 노드가 두 개 이상 있는 경우 샤드에서는 복제를 지원합니다. 이 경우 한 노드는 읽기/쓰기 기본 노드이고 다른 노드는 읽기 전용 복제 노드입니다.

향상된 내결함성을 위해 Redis 클러스터에 최소 두 개의 노드가 있으며, 자동 장애 조치가 있는 다중 AZ를 활성화하는 것이 좋습니다. 자세한 내용은 장애 완화 (p. 395) 단원을 참조하십시오.

Redis(클러스터 모드 비활성화됨) 클러스터 수요가 변경되면 확장 또는 축소할 수 있습니다. 이렇게 하려면 클러스터를 다른 노드 인스턴스 유형으로 이동합니다. 애플리케이션이 읽기 집약적인 경우 읽기 전용 복제본 Redis(클러스터 모드 비활성화됨) 클러스터를 추가하는 것이 좋습니다 이렇게 하면 보다 적절한 수의 노드로 읽기를 분산할 수 있습니다.

ElastiCache는 Redis(클러스터 모드 비활성화됨) 클러스터의 노드 유형을 더 큰 노드 유형으로 동적으로 변경하는 것을 지원합니다. 스케일 업 또는 스케일 다운에 대한 정보는 Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 조정 (p. 220) 또는 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터 조정 (p. 230) 단원을 참조하십시오.

Redis용 ElastiCache 복제

ElastiCache 콘솔과 ElastiCache API 및 AWS CLI 간의 기술 차이를 보다 정확하게 이해하려면 계속 읽기 전에 Redis용 ElastiCache 용어 (p. 14) 단원을 참조하십시오.

한 샤드(API 및 CLI에서 노드 그룹이라고 함)에서 2~six개의 노드를 그룹화하여 복제를 구현합니다. 이러한 노드 중 하나는 읽기/쓰기 기본 노드입니다. 다른 모든 노드는 읽기 전용 복제본 노드입니다.

각 복제본 노드는 기본 노드에서 데이터 사본을 유지합니다. 복제본 노드는 비동기식 복제 메커니즘을 사용하여 기본 노드와의 동기화를 유지합니다. 애플리케이션은 클러스터에 있는 모든 노드에서 읽을 수 있지만 기본 노드에만 쓸 수 있습니다. 읽기 전용 복제본은 여러 엔드포인트에 읽기를 분산하여 확장성을 향상합니다. 또한 읽기 전용 복제본은 여러 데이터 사본을 유지 관리하여 내결함성을 개선합니다. 다중 가용 영역에서 읽기 전용 복제본을 찾으면 내결함성이 더욱 개선됩니다. 내결함성에 대한 자세한 내용은 장애 완화 (p. 395) 단원을 참조하십시오.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Redis용 ElastiCache 복제

Redis(클러스터 모드 비활성화됨) 클러스터는 하나의 샤드(API 및 CLI에서 노드 그룹이라고 함)를 지원합니다. Redis(클러스터 모드 활성화됨) 클러스터는 1~90개의 샤드(API 및 CLI에서 노드 그룹이라고 함)를 지원합니다.

API 및 CLI 각각에서의 복제는 다른 용어를 사용하여 이전 버전과의 호환성을 유지하지만, 결과는 동일합니다. 다음 표는 복제 구현을 위한 API 및 CLI 용어를 보여줍니다.

복제 비교: Redis(클러스터 모드 비활성화됨) 및 Redis(클러스터 모드 활성화됨)

다음 표에서 Redis(클러스터 모드 비활성화됨) 복제 그룹과 Redis(클러스터 모드 활성화됨) 복제 그룹의 기능을 비교할 수 있습니다.

	Redis(클러스터 모드 비활성화됨)	Redis(클러스터 모드 활성화됨)
샤드 (노드 그룹)	1	1–90
각 샤드(노드 그룹)의 복제본 수	0–5	0–5
데이터 파티셔닝	아니요	예
복제본 추가/삭제	예	예
노드 그룹 추가/삭제	아니요	예
확장 지원	예	예
엔진 업그레이드 지원	예	예
복제본을 기본 노드로 승격	예	자동
다중 AZ 및 자동 장애 조치	선택	필수 여부
백업/복구	예	예
☆ト つ・		

참고:

복제본이 없는 기본 노드에서 장애가 발생할 경우 기본 노드에 저장된 데이터가 모두 손실됩니다.

백업 및 복원을 사용하여 Redis(클러스터 모드 활성화됨)로 마이그레이션할 수 있습니다.

백업 및 복원을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터 크기를 조정할 수 있습니다.

모든 샤드(API 및 CLI에서 노드 그룹) 및 노드가 동일한 AWS 리전에 있어야 합니다. 그러나 해당 AWS 리전의 여러 가용 영역에서 개별 노드를 프로비저닝할 수 있습니다.

읽기 전용 복제본은 데이터가 둘 이상의 노드—(기본 및 하나 이상의 읽기 전용 복제본)에 복제되므로 잠재적인 데이터 손실이 방지됩니다. 안정성과 복구 속도를 높이려면 여러 가용 영역에 읽기 전용 복제본을 하나이상 생성하는 것이 좋습니다. 또한, AOF를 사용하는 대신 자동 장애 조치가 포함된 다중 AZ를 활성화합니다. 자동 장애 조치가 포함된 다중 AZ가 활성화될 경우 AOF가 비활성화됩니다. 자세한 내용은 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132) 단원을 참조하십시오.

복제: 제한 및 제외

- 노드 유형 cache.t1.micro 및 cache.t2.에 대해서는 AOF가 지원되지 않습니다. 이러한 유형의 노드에 서는 appendonly 파라미터 값이 무시됩니다.
- 자동 장애 조치가 포함된 다중 AZ는 노드 유형 T1에서 지원되지 않습니다.

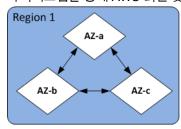
AOF 및 다중 AZ에 대한 자세한 내용은 장애 완화 (p. 395) 단원을 참조하십시오.

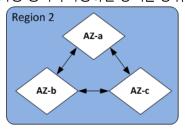
AWS 리전 및 가용 영역

Amazon ElastiCache은 전 세계의 여러 AWS 리전에서 사용할 수 있습니다. 따라서 비즈니스 요구 사항에 적합한 위치에서 ElastiCache 클러스터를 시작할 수 있습니다. 예를 들어 고객과 가장 가깝거나 특정 법적 요구사항을 준수하는 AWS 리전에서 시작할 수 있습니다.

기본적으로 AWS SDK, AWS CLI, ElastiCache API, ElastiCache 콘솔은 미국 서부(오레곤) 리전을 참조합니다. ElastiCache를 새 AWS 리전에서 사용할 수 있게 되면서 이러한 AWS 리전의 새 엔드포인트도 사용할 수 있습니다. HTTP 요청, AWS SDK, AWS CLI 및 ElastiCache 콘솔에서 이를 사용할 수 있습니다.

각 AWS 리전은 다른 AWS 리전에서 완전히 격리되도록 설계되었습니다. 각 리전 내에는 가용 영역이 여러개 있습니다. 서로 다른 가용 영역에서 노드를 시작하면 가능한 최고 수준의 내결함성을 갖출 수 있습니다. AWS 리전 및 가용 영역에 대한 자세한 내용은 리전 및 가용 영역 선택 (p. 47) 단원을 참조하십시오. 다음 다이어그램을 통해 AWS 리전 및 가용 영역이 작동하는 방식을 상위 수준에서 볼 수 있습니다.





ElastiCache에서 지원되는 AWS 리전 및 해당 엔드포인트에 대한 자세한 내용은 지원되는 리전 및 엔드포인트 (p. 47) 단원을 참조하십시오.

Redis용 ElastiCache 엔드포인트

엔드포인트는 애플리케이션이 ElastiCache 노드 또는 클러스터에 연결하는 데 사용하는 고유한 주소입니다.

Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 엔드포인트

단일 노드 Redis 클러스터에 대한 엔드포인트는 읽기 및 쓰기 모두를 위해 클러스터에 연결하는 데 사용됩니다.

Redis(클러스터 모드 비활성화됨)에 대한 다중 노드 엔드포인트

다중 노드 Redis(클러스터 모드 비활성화됨) 클러스터에는 두 가지 유형의 엔드포인트가 있습니다. 기본 엔드포인트는 기본 역할의 특정 노드가 변경된 경우에도 항상 클러스터의 기본 노드에 연결됩니다. 클러스터에 대한 모든 쓰기를 위해 기본 엔드포인트를 사용합니다.

Redis(클러스터 모드 비활성화됨) 클러스터의 읽기 엔드포인트는 항상 특정 노드를 가리킵니다. 읽기 전용 복제본을 추가 또는 제거할 때마다 애플리케이션에서 관련된 노드 엔드포인트를 업데이트해야 합니다.

Redis(클러스터 모드 활성화됨) 엔드포인트

Redis(클러스터 모드 활성화됨) 클러스터에는 단일 구성 엔드포인트가 있습니다. 구성 엔드포인트에 연결되면 애플리케이션이 클러스터의 각 샤드에 대한 기본 및 읽기 엔드포인트를 찾을 수 있습니다.

자세한 내용은 연결 엔드포인트 찾기 (p. 180) 단원을 참조하십시오.

ElastiCache 파라미터 그룹

캐시 파라미터 그룹은 지원되는 엔진 소프트웨어에 대한 런타임 설정을 관리하는 간단한 방법입니다. 메모리 사용량, 제거 정책, 항목 크기 등을 제어하는 데 여러 가지 파라미터가 사용됩니다. ElastiCache 파라미터 그

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Redis용 ElastiCache 보안

룹이란 클러스터에 적용할 수 있는 엔진별 파라미터의 이름이 지정된 모음을 말합니다. 이를 통해 해당 클러 스터에 있는 모든 노드가 정확히 동일한 방법으로 구성되게 할 수 있습니다.

지원되는 파라미터의 목록, 해당 기본값 및 수정할 수 있는 사항은 DescribeEngineDefaultParameters 단원 (CLI: describe-engine-default-parameters)을 참조하십시오.

ElastiCache 파라미터 그룹에 대한 자세한 내용은 파라미터 그룹을 사용해 엔진 파라미터 구성 (p. 260) 단원을 참조하십시오.

Redis용 ElastiCache 보안

향상된 보안을 위해 Redis용 ElastiCache 노드 액세스는 허용하는 Amazon EC2 인스턴스에 실행되는 애플리케이션으로 제한됩니다. 서브넷 그룹 또는 보안 그룹을 사용하여 클러스터에 액세스할 수 있는 Amazon EC2 인스턴스를 제어할 수 있습니다.

기본적으로 모든 새 Redis용 ElastiCache 클러스터는 Amazon Virtual Private Cloud(Amazon VPC) 환경에서 시작됩니다. 특정 서브넷에서 실행 중인 Amazon EC2 인스턴스로부터 클러스터 액세스를 부여하려면 서브 넷 그룹을 사용하면 됩니다. Amazon VPC 외부에서 클러스터를 실행하도록 선택한 경우 보안 그룹을 생성할 수 있습니다. 이를 통해 특정 Amazon EC2 보안 그룹 내에서 실행되는 Amazon EC2 인스턴스를에 권한을 부여할 수 있습니다.

Redis용 ElastiCache는 노드 액세스를 제한하는 것 외에 지정된 버전의 Redis용 ElastiCache를 실행 중인 노드의 TLS 및 실행 중 암호화를 지원합니다. 자세한 내용은 다음 자료를 참조하십시오.

- Amazon ElastiCache의 데이터 보안 (p. 299)
- HIPAA 자격 획득 (p. 393)
- Redis AUTH 명령을 사용하여 사용자 인증 (p. 313)

ElastiCache 보안 그룹

Note

ElastiCache 보안 그룹은 Amazon Virtual Private Cloud(Amazon VPC) 환경에서 실행되지 않는 클러스터에만 적용됩니다. Amazon VPC를 기반으로 Virtual Private Cloud (VPC)에서 ElastiCache 노드를 실행하는 경우 Amazon VPC 보안 그룹으로 캐시 클러스터에 대한 액세스를 제어할 수 있습니다. 이는 ElastiCache 보안 그룹과는 다릅니다. Amazon VPC와 함께 ElastiCache를 사용하는 방법에 대한 자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

ElastiCache를 사용하면 보안 그룹을 통해 클러스터에 대한 액세스를 제어할 수 있습니다. 보안 그룹은 클러스터에 대한 네트워크 액세스를 제어하는 방화벽 역할을 합니다. 기본적으로 클러스터로 액세스하는 네트워크는 해제되어 있습니다. 애플리케이션이 클러스터에 액세스하도록 하려면 특정 Amazon EC2 보안 그룹의호스트에서 액세스를 명시적으로 활성화해야 합니다. 수신 규칙이 구성되면 동일한 규칙이 해당 보안 그룹과관련된 모든 클러스터에 적용됩니다.

클러스터에 대한 네트워크 액세스를 허용하려면 먼저 보안 그룹을 생성합니다. 그런 다음 AuthorizeCacheSecurityGroupIngress API 작업 또는 authorize-cache-security-group-ingress AWS CLI 명령을 사용하여 원하는 Amazon EC2 보안 그룹을 승인합니다. 이렇게 하면 결과적으로 허용되는 Amazon EC2 인스턴스가 지정됩니다. 생성 시 보안 그룹을 클러스터에 연결할 수 있습니다. ElastiCache 관리 콘솔, ModifyCacheCluster API 작업 또는 modify-cache-cluster AWS CLI 명령을 사용하여 이 작업을 수행할 수도 있습니다.

Important

IP 범위 기반 액세스 제어는 클러스터에 사용할 수 없습니다. 클러스터에 액세스하는 모든 클라이언 트는 Amazon EC2 네트워크 내에 있어야 하며, 이전에 설명한 대로 보안 그룹을 사용해 승인을 받 아야 합니다. 보안 그룹에 대한 자세한 내용은 보안 그룹: EC2-Classic (p. 334) 단원을 참조하십시오.

ElastiCache 서브넷 그룹

서브넷 그룹은 Amazon VPC 환경에서 실행 중인 클러스터에 대해 지정할 수 있는 서브넷(일반적으로 프라이빗 서브넷) 모음입니다.

Amazon VPC에서 클러스터를 생성할 경우 캐시 서브넷 그룹을 지정해야 합니다. ElastiCache는 해당 캐시 서브넷 그룹을 사용하여 캐시 노드에 연결된 서브넷 내의 서브넷 및 IP 주소를 선택합니다.

Amazon VPC 환경에서 캐시 서브넷 그룹을 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- Amazon VPC 및 ElastiCache 보안 (p. 316)
- 2단계: 액세스 자격 부여 (p. 23)
- 서브넷 및 서브넷 그룹 (p. 329)

Redis용 ElastiCache 백업

백업은 Redis 클러스터의 특정 시점 복사입니다. 백업을 사용하여 기존 클러스터를 복원하거나 새 클러스터 를 시드할 수 있습니다. 백업은 클러스터의 모든 데이터와 일부 메타데이터로 구성됩니다.

클러스터에서 실행되는 Redis 버전에 따라 백업 프로세스를 완료하는 데 필요한 예약된 메모리의 양이 달라집니다. 자세한 내용은 다음을 참조하십시오.

- Redis용 ElastiCache 백업 및 복원 (p. 187)
- 동기화 및 백업 구현 방법 (p. 143)
- 백업의 성능 영향 (p. 189)
- 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)

ElastiCache 이벤트

캐시 클러스터에서 중요 이벤트가 발생하면 ElastiCache는 특정 Amazon SNS 주제에 알림을 전송합니다. 이러한 이벤트로는 노드 추가 실패 또는 성공, 보안 그룹 수정 등을 들 수 있습니다. 주요 이벤트를 모니터링 하면 클러스터의 현재 상태를 파악할 수 있으며, 많은 경우에 교정 작업을 수행할 수도 있습니다.

ElastiCache 이벤트에 대한 자세한 내용은 ElastiCache 이벤트 모니터링 (p. 367) 단원을 참조하십시오.

Redis용 ElastiCache 용어

2016년 10월에 Amazon ElastiCache는 Redis 3.2에 대한 지원을 시작했습니다. 이 시점에 최대 90개의 샤드 (ElastiCache API 및 AWS CLI에서 노드 그룹이라고 함)로의 데이터 분할에 대한 지원이 추가되었습니다. 이전 버전과의 호환성을 유지하기 위해 새 Redis 기능을 포함하도록 API 버전 2015-02-02 작업을 확장했습니다.

동시에 ElastiCache 콘솔에서 이 새 기능과 업계에서 일반적으로 사용되는 용어를 사용하기 시작했습니다. 이러한 변경 사항은 어떤 경우에 API 및 CLI에서 사용되는 용어가 콘솔에서 사용되는 용어와 다를 수 있다는 것을 의미합니다. 다음 목록에는 API 및 CLI와 콘솔 간에 다를 수 있는 용어가 식별되어 있습니다.

캐시 클러스터 또는 노드와 노드 비교

복제본 노드가 없는 경우 노드와 캐시 클러스터 사이에 일대일 관계가 있습니다. 따라서 ElastiCache 콘솔은 종종 상호 교환적으로 용어를 사용했습니다. 콘솔에서는 이제 노드 처리량이라는 용어를 사용합니

다. 한 가지 예외는 [Create Cluster] 버튼입니다. 이 버튼은 복제본 노드를 포함 또는 포함하지 않고 클러 스터를 생성하는 프로세스를 시작합니다.

ElastiCache API 및 AWS CLI는 과거의 용어를 계속 사용합니다.

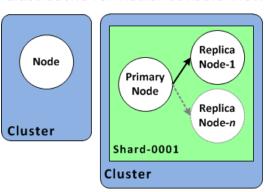
클러스터와 복제 그룹 비교

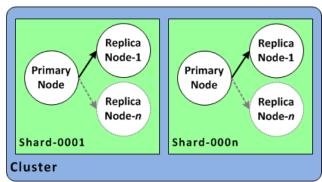
콘솔에서는 이제 모든 Redis용 ElastiCache 클러스터에 대해 클러스터라는 용어를 사용합니다. 콘솔에서는 다음과 같은 모든 환경에서 클러스터 용어를 사용합니다.

- 클러스터가 단일 노드 Redis 클러스터인 경우
- 클러스터가 단일 샤드 내에서 복제를 지원하는 Redis(클러스터 모드 비활성화됨) 클러스터인 경우 (API 및 CLI에서 노드 그룹이라고 함)
- 클러스터가 1 ~ 90개의 샤드를 지원하는 Redis(클러스터 모드 활성화됨) 클러스터인 경우

다음 다이어그램은 콘솔의 관점에서 바라본 Redis용 ElastiCache 클러스터의 다양한 토폴로지를 나타낸 것입니다.

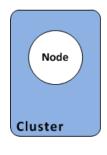
ElastiCache for Redis: Console View

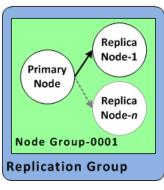


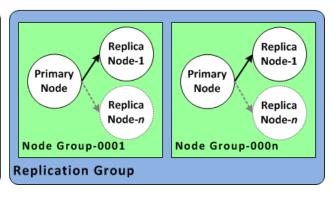


ElastiCache API 및 AWS CLI 작업에서는 여전히 단일 노드 Redis용 ElastiCache 클러스터를 다중 노드 복제 그룹과 다른 것으로 식별합니다. 다음 다이어그램은 ElastiCache API 및 AWS CLI 관점에서 바라본 다양한 Redis용 ElastiCache 토폴로지를 나타낸 것입니다.

ElastiCache for Redis: API/CLI View







구현 관리용 도구

ElastiCache 클러스터에 대한 Amazon EC2 인스턴스 액세스를 허용하면 ElastiCache 클러스터를 관리할 수 있는 네 가지 수단(AWS Management 콘솔, ElastiCache용 AWS CLI, ElastiCache용 AWS SDK 및 ElastiCache API)이 제공됩니다.

AWS Management 콘솔 사용

가장 간편한 Amazon ElastiCache 관리 방법은 AWS Management 콘솔을 사용하는 것입니다. 콘솔을 사용하면 캐시 클러스터를 생성하고 캐시 노드를 추가 및 제거하며 코드를 작성하지 않고도 다른 관리 작업을 수행할 수 있습니다. 콘솔은 또한 CloudWatch의 캐시 노드 성능 그래프를 제공합니다. 이러한 그래프는 캐시엔진 활동, 메모리 및 CPU 사용률 및 기타 측정치를 보여 줍니다. 자세한 내용은 이 사용 설명서의 특정 항목을 참조하십시오.

AWS CLI 사용

ElastiCache에서 AWS Command Line Interface(AWS CLI)를 사용해도 됩니다. AWS CLI를 사용하면 캐시 클러스터 시작 또는 중지와 같은 한 번에 한 작업을 쉽게 수행할 수 있습니다. 선택한 스크립팅 언어에서 ElastiCache용 AWS CLI 명령를 호출하여 반복 작업을 자동화할 수도 있습니다. AWS CLI에 대한 자세한 내용은 사용 설명서 및 AWS CLI Command Reference를 참조하십시오.

AWS SDK 사용

애플리케이션에서 ElastiCache에 액세스하려면 AWS SDK(소프트웨어 개발 키트) 중 하나를 사용할 수 있습니다. SDK는 ElastiCache API 호출을 래핑하고 ElastiCache API의 하위 수준 세부 정보로부터 애플리케이션을 보호합니다. 사용자는 자격 증명을 제공하고 SDK 라이브러리가 인증 및 요청 서명을 담당합니다. AWS SDK에 대한 자세한 내용은 Amazon Web Services용 도구를 참조하십시오.

ElastiCache API 사용

또한 ElastiCache 웹 서비스 API에 대해 직접 애플리케이션 코드를 작성할 수 있습니다. API를 사용할 때 HTTP 요청을 구성하고 인증하기 위해 필요한 코드를 작성해야 합니다. 또한, ElastiCache의 결과를 분석하고 오류를 처리하기 위해 코드를 작성해야 합니다. API에 대한 자세한 내용은 ElastiCache API 사용 (p. 421)을 참조하십시오.

추가 리소스

Redis용 Amazon ElastiCache 배포 관리에 대한 자세한 내용은 다음을 참조하십시오.

- Redis용 ElastiCache 구현 관리 (p. 37)
- 인터네트워크 트래픽 개인 정보 (p. 315)
- Elasticache에서 로깅 및 모니터링 (p. 359)

Memcached와 Redis 비교

Amazon ElastiCache는 Memcached 및 Redis 캐시 엔진을 지원합니다. 각 엔진에는 몇 가지 장점이 있습니다. 이 항목의 정보를 활용하면 요구 사항에 가장 잘 맞는 엔진과 버전을 선택하는 데 도움이 됩니다.

Important

캐시 클러스터 또는 복제 그룹을 생성한 후에는 최신 엔진 버전으로 업그레이드할 수 있지만, 이전 엔진 버전으로 다운그레이드할 수는 없습니다. 이전 엔진 버전을 사용하려면 기존 캐시 클러스터 또 는 복제 그룹을 삭제하고 이전 엔진 버전을 통해 다시 생성해야 합니다.

표면적으로는 엔진이 유사하게 보입니다. 각 엔진은 인 메모리 키-값 저장소입니다. 그러나 실제로 상당한 차이점이 있습니다.

다음과 같은 경우 Memcached를 선택합니다.

- 가능한 가장 단순한 모델이 필요한 경우
- 여러 코어 또는 스레드가 있는 큰 노드를 실행해야 하는 경우
- 시스템의 요구 사항이 증가하고 감소함에 따라 노드를 추가 및 제거하는 확장 및 축소 기능이 필요한 경우
- 데이터베이스와 같은 객체를 캐시에 저장해야 하는 경우

다음과 같은 경우 Redis용 ElastiCache 버전이 있는 Redis를 선택합니다.

• Redis용 ElastiCache 버전 5.0.0(개선됨)

생산자가 실시간으로 새 항목을 추가하고 소비자가 차단 또는 비 차단 방식으로 메시지를 사용할 수 있도록 지원하는 로그 데이터 구조인 Redis 스트림을 사용할 수 있습니다.

자세한 내용은 Redis 버전 5.0.0(개선됨)을 참조하십시오.

• Redis용 ElastiCache 버전 4.0.10(개선됨)

암호화 및 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드의 동적으로 추가 또는 제거를 지원합니다.

자세한 내용은 Redis 버전 4.0.10(개선됨)을 참조하십시오.

• Redis용 ElastiCache 버전 3.2.10(개선됨)

Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 동적으로 추가 또는 제거하는 기능을 지원합니다.

Important

Redis용 ElastiCache 3.2.10에서는 현재 암호화가 지원되지 않습니다.

자세한 내용은 다음 자료를 참조하십시오.

- Redis 버전 3.2.10(개선됨)
- Redis에 대한 온라인 리샤딩 모범 사례에 대한 자세한 내용은 다음을 참조하십시오.
 - 모범 사례: 온라인 리샤딩
 - Redis(클러스터 모드 활성화됨)를 위한 온라인 리샤딩 및 샤드 재분배
- Redis 클러스터 조정에 대한 자세한 내용은 조정 단원을 참조하십시오.
- Redis용 ElastiCache 버전 3.2.6(개선됨)

다음 기능 이외에 이전 Redis 버전의 기능이 필요하면 Redis용 ElastiCache 3.2.6을 선택하십시오.

- 전송 중 데이터 암호화. 자세한 내용은 Redis용 Amazon ElastiCache 전송 중인 데이터 암호화 단원을 참조하십시오.
- 미사용 데이터 암호화. 자세한 내용은 Redis용 Amazon ElastiCache미사용 데이터 암호화 단원을 참조 하십시오.
- HIPAA 자격 획득 인증 자세한 내용은 Redis용 Amazon ElastiCache에 대한 HIPAA 자격 획득 단원을 참 조하십시오.
- Redis용 ElastiCache (클러스터 모드 활성화됨) 버전 3.2.4

다음 기능 이외에 Redis 2.8.x의 기능이 필요하면 Redis 3.2.4(클러스터 모드)를 선택합니다.

- 2~90개의 노드 그룹으로 데이터를 분할해야 하는 경우(클러스터 모드에만 해당)
- 지역 검색 인덱싱이 필요한 경우(클러스터 모드 또는 비클러스터 모드)
- 여러 데이터베이스를 지원할 필요가 없는 경우

Important

Redis(클러스터 모드 활성화됨)에는 다음과 같은 제한 사항이 있습니다.

- 더 큰 노드 유형으로 확장할 수 없음
- 노드 그룹의 복제본 수를 변경할 수 없음(분할)
- Redis용 ElastiCache(비 클러스터 모드) 2.8x 및 3.2.4(개선됨)

다음과 같은 경우 Redis 2.8.x 또는 Redis 3.2.4(비클러스터 모드)를 선택합니다.

- 문자열, 해시, 목록, 세트, 정렬된 세트 및 비트맵과 같은 복잡한 데이터 유형이 필요한 경우
- 인 메모리 데이터 세트를 정렬하거나 순위를 지정해야 하는 경우
- 키 저장소의 지속성을 원할 경우
- 읽기 집약적 애플리케이션을 위해 기본 항목에서 하나 이상의 읽기 전용 복제본으로 데이터를 복제해야 하는 경우
- 기본 노드가 실패할 때 자동 장애 조치가 필요한 경우
- 서버에 대한 이벤트를 클라이언트에 알리기 위해 게시 및 구독(게시/구독) 기능—이 필요합니다.
- 백업 및 복원 기능이 필요한 경우
- 여러 데이터베이스를 지원해야 하는 경우

Memcached, Redis(클러스터 모드 비활성화됨) 및 Redis(클러스터 모드 활성화됨)의 비교 요약

	Memcached	Redis(클러스터 모드 비활성화됨)	Redis(클러스터 모드 활성화됨)
엔진 버전	1.4.x	2.8.x 이상	3.2.x 이상
데이터 유형	간단함	2.8.x - 복합 *	3.2.x 이상 - 복합
데이더 휴영		복합	
데이터 파티셔닝	예	아니요	예
클러스터 수정 가능	예	예	3.2.10이상 - 제한
온라인 리샤딩	아니요	아니요	3.2.10 이상
암호화	아니요	3.2.6, 4.0.10 및 이상	3.2.6, 4.0.10 및 이상
규정 준수 인증			,
규정 준수 인증			
FedRAMP	예외전 2015-0	2-0 3 .2.6, 4.0.10 및 이상	3.2.6, 4.0.10 및 이상

	Memcached	Redis(클러스터 모드 비활성화됨)	Redis(클러스터 모드 활성화됨)
HIPAA	아니요	3.2.6, 4.0.10 및 이상	3.2.6, 4.0.10 및 이상
PCI DSS	아니요	3.2.6, 4.0.10 및 이상	3.2.6, 4.0.10 및 이상
다중 스레드	예	아니요	아니요
노드 유형 업그레이드	아니요	예	예
엔진 업그레이드	예	예	예
고가용성(복제)	아니요	예	예
자동 장애조치(failover)	아니요	선택	필수
게시/구독 기능	아니요	예	예
정렬된 세트	아니요	예	예
백업 및 복원	아니요	예	예
지역 검색 인덱싱	아니요	2.8.x - 아니요	예
~ ~ ~ ~ ~		3.2.x 이상 - 예) VII
참고:			
무지역 개체(에: 데이터베이스)			

문자열, 객체(예: 데이터베이스)

* 문자열, 세트, 정렬된 세트, 목록, 해시, 비트맵, HyperLogLog

문자열, 세트, 정렬된 세트, 목록, 해시, 비트맵, hyperloglog, 지역 검색 인덱스

클러스터에 대한 엔진을 선택한 후 해당 엔진의 최신 버전을 사용하는 것이 좋습니다. 자세한 내용은 지원 ElastiCache for Memcached 버전 또는 지원 Redis용 ElastiCache 버전을 참조하십시오.

Redis용 Amazon ElastiCache 시작하기

아래 단원에서는 ElastiCache 관리 콘솔을 사용하여 Redis(클러스터 모드 비활성화됨) 클러스터를 생성하는 것부터 이 클러스터에 액세스 권한을 부여하고 연결하며 마지막으로 이 클러스터를 삭제하는 데까지 안내해 드립니다. 그 일환으로 첫 부분은 클러스터에 대한 요구 사항을 확인하고 AWS 계정을 생성하는 데 도움을 드리는 것부터 시작합니다.

Amazon ElastiCache는 Redis 복제 그룹을 사용하여 고가용성을 지원합니다. Redis 복제 그룹과 이를 생성하는 방법에 대한 정보는 고가용성을 위한 복제 그룹 사용 (p. 127)를 참조하십시오.

Redis 버전 3.2부터 ElastiCache Redis는 각 노드 그룹이 복제 그룹을 구현하는 여러 노드 그룹에서의 데이터 분할을 지원합니다. 이 연습에서는 독립 실행형 Redis 클러스터를 생성합니다.

주제

- 요구 사항 결정 (p. 20)
- 설정 (p. 20)
- 1단계: 클러스터 시작 (p. 21)
- 2단계: 액세스 자격 부여 (p. 23)
- 3단계: 클러스터 노드에 연결 (p. 23)
- 4단계: 클러스터 삭제(불필요한 요금 부과 방지) (p. 26)
- 추가 정보 (p. 26)

요구 사항 결정

클러스터 또는 복제 그룹을 생성할 때 비즈니스 요구를 충족하고 다시 실행할 필요가 없도록 Redis 클러스터 또는 복제 그룹을 생성하기 전에는 항상 클러스터 요구 사항을 확인해야 합니다. 이 연습에서는 대체로 클러스터에 기본값을 적용할 것이므로 요구 사항 확인은 생략하겠습니다. 자세한 내용은 요구 사항 결정 (p. 65) 단원을 참조하십시오.

설정

다음 주제에서는 ElastiCache 사용을 시작하기 위해 수행해야 하는 1회성 작업에 관해 설명합니다.

주제

- AWS 계정 생성 (p. 20)
- 권한 설정(신규 ElastiCache 사용자에게만 해당) (p. 21)

AWS 계정 생성

Amazon ElastiCache를 사용하려면 활성 AWS 계정과 ElastiCache 및 기타 AWS 리소스에 액세스할 수 있는 권한이 있어야 합니다.

아직 AWS 계정이 없다면 사용할 계정을 하나 생성해야 합니다. AWS 계정은 무료입니다. AWS 서비스에 가입한 것 자체로는 요금이 청구되지 않고 사용한 AWS 서비스에 대해서만 청구됩니다.

AWS 계정을 생성하려면

- 1. https://portal.aws.amazon.com/billing/signup을 엽니다.
- 2. 온라인 지시 사항을 따릅니다.

등록 절차 중 전화를 받고 전화 키패드를 사용하여 확인 코드를 입력하는 과정이 있습니다.

권한 설정(신규 ElastiCache 사용자에게만 해당)

Amazon ElastiCache에서는 서비스 연결 역할을 생성 및 사용해 리소스를 프로비저닝하고 사용자를 대신해 다른 AWS 리소스 및 서비스에 액세스합니다. ElastiCache에서 사용자를 위해 서비스 연결 역할을 생성하도록 하려면 AmazonElastiCacheFullAccess라는 AWS 관리형 정책을 사용하십시오. 이 역할은 서비스가 사용자를 대신해 서비스 연결 역할을 생성하는 데 필요한 권한으로 사전에 프로비저닝되어 있습니다.

사용자는 기본 정책을 사용하는 대신 사용자 지정 관리형 정책을 사용하는 쪽을 선택할 수 있습니다. 이 경우 iam:createServiceLinkedRole을 호출할 권한이 있거나 ElastiCache 서비스 연결 역할을 생성했어야 합니다.

자세한 내용은 다음을 참조하십시오.

- 새 정책 생성(IAM)
- Amazon ElastiCache에 대한 AWS 관리형(미리 정의된) 정책 (p. 347)
- Amazon ElastiCache에 서비스 연결 역할 사용 (p. 349)

1단계: 클러스터 시작

시작하려는 클러스터가 활성화되고 샌드박스에서 실행되지 않습니다. 삭제하기 전까지 인스턴스에 대해 표준 ElastiCache 사용 요금이 청구됩니다. 여기에 설명된 연습을 한 번에 끝내고 연습을 마칠 때 클러스터를 삭제하면 총 청구 비용이 가장 적게 듭니다(일반적으로 1달러 미만). ElastiCache 사용 요금에 대한 자세한 내용은 https://aws.amazon.com/elasticache/를 참조하십시오.

Important

클러스터는 Amazon VPC에서 시작됩니다. 클러스터 생성을 시작하기 전에 서브넷 그룹을 생성해야 합니다. 자세한 내용은 서브넷 그룹 생성 (p. 329) 단원을 참조하십시오.

독립 실행형 Redis(클러스터 모드 비활성화됨) 클러스터를 생성하려면

- 1. AWS Management 콘솔에 로그인한 후 https://console.aws.amazon.com/elasticache/에서 Amazon ElastiCache 콘솔을 엽니다.
- 2. [Get Started Now]를 선택합니다.

사용할 수 있는 클러스터가 이미 있는 경우 [Launch Cluster]를 선택합니다.

- 3. 상단 오른쪽 모서리의 목록에서 이 클러스터를 시작할 AWS 리전을 선택합니다.
- 4. [Cluster engine]에 대해 [Redis]를 선택합니다.
- 5. Cluster Mode enabled(Scale Out)(클러스터 모드 활성화(확장))를 선택하지 않았는지 확인합니다.
- 6. 다음과 같이 [Redis settings] 단원을 완료합니다.
 - a. [Name]의 경우 클러스터 이름을 입력합니다.

클러스터 명명 제약 조건은 다음과 같습니다.

• 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 1단계: 클러스터 시작

- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.
- b. [Engine version compatibility] 목록에서 이 클러스터를 실행할 Redis 엔진 버전을 선택합니다. 이전 버전을 실행할 특별한 이유가 없는 한 최신 버전을 선택하는 것이 좋습니다.
- c. [Port]에서 기본 포트 6379를 허용합니다. 다른 포트를 사용해야 하는 경우 포트 번호를 입력합니다.
- d. [Parameter group]에서 이 클러스터에서 사용할 파라미터 그룹을 선택하거나 "새로 생성"을 선택하여 새 파라미터 그룹을 생성하여 해당 클러스터에서 사용합니다. 이 연습에서는 기본 파라미터 그룹을 사용합니다.

자세한 내용은 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

e. [Node type]에 대해 이 클러스터에 사용할 노드 유형을 선택합니다. 이 연습에서는 테이블 위의 [t2] 인스턴스 패밀리, [cache.t2.small], [Save]를 순차적으로 선택합니다.

자세한 내용은 노드 크기 선택 (p. 67) 단원을 참조하십시오.

f. [Number of replicas]에서 이 클러스터의 읽기 전용 복제본 개수를 선택합니다. 이 연습에서는 독립 실행형 클러스터를 생성할 것이므로 [None]을 선택합니다.

[없음]을 선택하면 [Replication group description] 필드가 사라집니다.

7. 다음과 같이 [Advanced Redis settings]을 선택하고 완료합니다.

Note

Redis(클러스터 모드 활성화됨) 복제 그룹을 생성하면 Advanced Redis settings(고급 Redis 설정) 세부 정보가 약간 다릅니다. Redis(클러스터 모드 활성화됨) 복제 그룹을 생성하는 단계별절차는 Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (p. 153) 단원을참조하십시오.

a. [Subnet group] 목록에서 클러스터에 적용할 서브넷을 선택합니다. 이 예제에서는 [default]를 선택합니다.

자세한 내용은 서브넷 및 서브넷 그룹 (p. 329) 단원을 참조하십시오.

- b. [Availability zone(s)]에 대해 다음 두 가지 옵션이 있습니다.
 - 기본 설정 없음 ElastiCache에서 가용 영역을 선택합니다.
 - Specify availability zones(가용 영역 설정) 사용자 클러스터의 가용 영역을 지정합니다.

이 연습에서는 [Specify availability zones]를 선택한 다음 [Primary] 아래의 목록에서 가용 영역을 선택합니다.

자세한 내용은 리전 및 가용 영역 선택 (p. 47) 단원을 참조하십시오.

c. [Security groups] 목록에서 이 클러스터에서 사용할 보안 그룹을 선택합니다. 이 예제에서는 [default]를 선택합니다.

자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

d. .RDB 파일의 데이터로 클러스터를 시드하려면 Seed RDB file S3 location(RDB 파일 S3 위치 지정) 상자에 .RDB 파일의 Amazon S3 위치를 입력합니다.

자세한 내용은 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211) 단원을 참조하십시오.

e. 프로덕션 클러스터가 아니기 때문에 [Enable automatic backups] 확인란의 선택을 취소합니다.

Redis 백업 및 복원에 대한 자세한 내용은 Redis용 ElastiCache 백업 및 복원 (p. 187)을 참조하십시오.

f. 유지 관리 기간은 ElastiCache가 벌점 2015 02 이 유지 관리를 예약하는 시간이며 일반적으로 매주 한 시간입니다. ElastiCache가 유지 관리 기간의 요일과 시간을 지정하도록(기본 설정 없음) 허

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 2단계: 액세스 자격 부여

용하거나 요일과 시간을 직접 지정할 수 있습니다(Specify maintenance window(유지 관리 기간 지정)). 유지 관리 기간 지정을 선택한 경우 유지 관리 기간의 [Start day], [Start time] 및 [Duration](시간)을 지정합니다. 이 예제에서는 기본 설정 없음을 선택합니다.

자세한 내용은 유지 관리 관리 중 (p. 50) 단원을 참조하십시오.

- g. [Notifications]를 [Disabled]로 둡니다.
- 8. [Create cluster]를 선택하여 클러스터를 시작하거나 [Cancel]을 선택해 작업을 취소합니다.

2단계: 액세스 자격 부여

이 단원에서는 Amazon EC2 인스턴스를 시작하고 연결하는 것에 익숙하다고 가정합니다. 자세한 내용은 Amazon EC2 Getting Started Guide섹션을 참조하십시오.

모든 ElastiCache 클러스터는 Amazon EC2 인스턴스에서 액세스하도록 설계되었습니다. 가장 일반적인 시나리오는 동일한 Amazon Virtual Private Cloud(Amazon VPC)의 Amazon EC2인스턴스에서 ElastiCache 클러스터에 액세스하는 것입니다. 이 항목에서는 시나리오를 다룹니다. 다른 Amazon VPC, 다른 AWS 리전 또는 기업 네트워크에서 ElastiCache 클러스터를 액세스에 대한 정보는 다음을 참조하십시오.

- Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴 (p. 320)
- AWS 외부에서 ElastiCache 리소스 액세스 (p. 107)

기본적으로 클러스터에 대한 네트워크 액세스는 이를 시작하는 데 사용한 사용자 계정으로 제한됩니다. EC2 인스턴스에서 클러스터에 연결하려면 EC2 인스턴스가 클러스터에 액세스하도록 권한을 부여해야 합니다. 필요한 단계는 클러스터를 EC2-VPC 또는 EC2-Classic으로 시작했는지에 따라 다릅니다.

클러스터에 액세스할 수 있는 권한을 부여하는 절차는 클러스터 또는 복제 그룹에 액세스 (p. 104) 단원을 참조하십시오.

3단계: 클러스터 노드에 연결

계속하기 전에 2단계: 액세스 자격 부여 (p. 23) 단계를 완료하십시오.

이 단원에서는 Amazon EC2 인스턴스를 생성했고 이 인스턴스에 연결할 수 있다고 가정합니다. 작업 방법에 대한 지침은 Amazon EC2 Getting Started Guide 단원을 참조하십시오.

Amazon EC2 인스턴스는 권한을 부여한 경우에만 클러스터 노드에 연결할 수 있습니다. 자세한 내용은 2단계: 액세스 자격 부여 (p. 23) 단원을 참조하십시오.

3 1단계 노드 엔드포인트 찾기

클러스터가 사용 가능한 상태이고 사용자가 이 클러스터에 액세스할 수 있는 권한을 부여받았다면(2단계: 액세스 자격 부여 (p. 23)) Amazon EC2 인스턴스에 로그인하여 클러스터에 연결할 수 있습니다. 이를 수행하려면 먼저 엔드포인트를 결정해야 합니다.

사용자의 엔드포인트를 찾으려면 엔진과 실행 중인 클러스터 유형에 관한 주제를 참조하십시오. 필요한 엔드 포인트를 찾은 경우 3.2단계에서 사용할 수 있도록 클립보드에 복사합니다.

- 연결 엔드포인트 찾기 (p. 180)
- Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 181)—복제 그룹의 기본 엔드포 인트 또는 독립형 노드의 노드 엔드포인트가 필요합니다.
- Redis(클러스터 모드 활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 182)—클러스터의 구성 엔드포인 트가 필요합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 3.2단계: Redis 클러스터 또는 복제 그룹에 연결(Linux)

- 엔드포인트 찾기 (AWS CLI) (p. 184)
- 엔드포인트 찾기 (ElastiCache API) (p. 186)

3.2단계: Redis 클러스터 또는 복제 그룹에 연결(Linux)

이제 필요한 엔드포인트가 있으므로 EC2 인스턴스에 로그인하여 클러스터 또는 복제 그룹에 연결할 수 있습니다.

다음 예제에서는 redis-cli 유틸리티를 사용하여 암호화가 활성화되지 않았고 Redis를 실행하는 클러스터에 연결합니다. Redis 및 사용 가능한 Redis 명령에 대한 자세한 내용은 Redis 명령 웹페이지를 참조하십시오.

redis-cli를 사용하여 암호화가 활성화되지 않은 Redis 클러스터에 연결하려면

- 1. 선택한 연결 유틸리티를 사용하여 Amazon EC2 인스턴스에 연결합니다. Amazon EC2 인스턴스에 연결하는 방법에 대한 지침은 Amazon EC2 Getting Started Guide 단원을 참조하십시오.
- 2. GNU 컴파일러 모음(gcc)을 다운로드하여 설치합니다.

EC2 인스턴스의 명령 프롬프트에서 다음 명령을 입력하고 확인 프롬프트에서 y을 입력합니다.

```
sudo yum install gcc
```

이렇게 하면 다음과 비슷하게 출력됩니다.

```
Loaded plugins: priorities, security, update-motd, upgrade-helper
Setting up Install Process
Resolving Dependencies
--> Running transaction check
...(output omitted)...
Total download size: 27 M
Installed size: 53 M
Is this ok [y/N]: y
Downloading Packages:
(1/11): binutils-2.22.52.0.1-10.36.amzn1.x86_64.rpm
                                                        | 5.2 MB
                                                        4.8 MB
(2/11): cpp46-4.6.3-2.67.amzn1.x86_64.rpm
                                                                     00:00
(3/11): qcc-4.6.3-3.10.amzn1.noarch.rpm
                                                       | 2.8 kB
                                                                     00:00
...(output omitted)...
Complete!
```

3. redis-cli 유틸리티를 다운로드하고 컴파일합니다. 이 유틸리티는 Redis 소프트웨어 배포에 포함되어 있습니다.

EC2 인스터스의 명령 프롬프트에 다음 명령을 입력합니다.

```
wget http://download.redis.io/redis-stable.tar.gz
tar xvzf redis-stable.tar.gz
cd redis-stable
make distclean // Ubuntu systems only
make
```

4. EC2 인스턴스의 명령 프롬프트에서 다음 명령을 입력하고 클러스터 및 포트의 엔드포인트를 이 예제에 표시된 것으로 대체합니다.

```
src/redis-cli -c -h mycachecluster.eaogs8.0001.usw2.cache.amazonaws.com -p 6379
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 3.2단계: Redis 클러스터 또는 복제 그룹에 연결(Windows)

그러면 다음과 유사한 Redis 명령 프롬프트가 나타납니다.

```
redis mycachecluster.eaogs8.0001.usw2.cache.amazonaws.com 6379>
```

5. Redis 명령을 실행합니다.

이제 클러스터에 연결되어 다음과 같이 Redis 명령을 실행할 수 있습니다.

```
set a "hello"
                        // Set key "a" with a string value and no expiration
OK
get a
                        // Get value for key "a"
"hello"
get b
                        // Get value for key "b" results in miss
(nil)
{\tt set}\ b "Good-bye" EX 5 \ \ // Set key "b" with a string value and a 5 second expiration
"Good-bye"
get b
                        // Get value for key "b"
"Good-bye"
                        // wait >= 5 seconds
get b
                        // key has expired, nothing returned
(nil)
quit
                        // Exit from redis-cli
```

3.2단계: Redis 클러스터 또는 복제 그룹에 연결 (Windows)

Redis CLI를 사용하여 EC2 Windows 인스턴스에서 Redis 클러스터에 연결하려면 redis-cli 패키지를 다운로 드하고 redis-cli.exe를 사용하여 EC2 Windows 인스턴스에서 Redis 클러스터에 연결해야 합니다.

다음 예제에서는 redis-cli 유틸리티를 사용하여 암호화가 활성화되지 않았고 Redis를 실행하는 클러스터에 연결합니다. Redis 및 사용 가능한 Redis 명령에 대한 자세한 내용은 Redis 명령 웹페이지를 참조하십시오.

redis-cli를 사용하여 암호화가 활성화되지 않은 Redis 클러스터에 연결하려면

- 1. 선택한 연결 유틸리티를 사용하여 Amazon EC2 인스턴스에 연결합니다. Amazon EC2 인스턴스에 연결하는 방법에 대한 지침은 Amazon EC2 Getting Started Guide 단원을 참조하십시오.
- 2. https://github.com/microsoftarchive/redis/releases/download/win-3.0.504/Redis-x64-3.0.504.zip 링크를 복사하고 인터넷 브라우저에 붙여 넣어 Github https://github.com/microsoftarchive/redis/releases/tag/win-3.0.504의 사용 가능한 릴리스에서 Redis 클라이언트에 대한 zip 파일을 다운로드합니다.

원하는 폴더/경로에 zip 파일의 압축을 풉니다.

명령 프롬프트를 열고 Redis 디렉터리로 변경한 다음 c:\Redis>redis-cli -h Redis_Cluster_Endpoint -p 6379 명령을 실행합니다.

예:

```
c:\Redis>redis-cli -h cmd.xxxxxxx.ng.0001.usw2.cache.amazonaws.com -p 6379
```

3. Redis 명령을 실행합니다.

이제 클러스터에 연결되어 다음과 같이 Redis 명령을 실행할 수 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 4단계: 클러스터 삭제

4단계: 클러스터 삭제(불필요한 요금 부과 방지)

Important

활발하게 사용하지 않는 클러스터는 삭제하는 것이 거의 언제나 바람직합니다. 클러스터의 상태가 [deleted]가 되기 전까지 요금이 계속 부과됩니다.

계속하기 전에 최소한 1단계: 클러스터 시작 (p. 21) 단계를 완료해야 합니다.

클러스터를 삭제하려면

- 1. AWS Management 콘솔에 로그인한 후 https://console.aws.amazon.com/elasticache/에서 Amazon ElastiCache 콘솔을 엽니다.
- Redis를 실행 중인 모든 클러스터의 목록을 보려면 탐색 창에서 [Redis]를 선택합니다.
- 삭제할 클러스터를 선택하려면 클러스터 목록에서 클러스터의 이름을 선택합니다.

Tip

ElastiCache 콘솔에서는 클러스터를 한 번에 하나씩만 삭제할 수 있습니다. 여러 클러스터를 선택하면 삭제 버튼이 비활성화됩니다. 여러 클러스터를 삭제하려면 클러스터마다 이 절차를 반복하십시오. 클러스터 하나가 완전히 삭제되기를 기다렸다가 그다음 클러스터를 삭제할 필요는 없습니다.

- 4. [Actions]에 대해 [Delete]를 선택합니다.
- 5. 클러스터 삭제 확인 화면에서 삭제를 선택하여 클러스터를 삭제하거나 취소를 선택하여 클러스터를 유 지합니다.

삭제를 선택하면 클러스터의 상태가 삭제 중으로 바뀝니다.

클러스터가 클러스터 목록에서 제거되는 즉시 요금 부과가 중단됩니다.

이로써 Redis용 ElastiCache 클러스터를 시작하여 이에 대한 액세스 권한을 부여하고, 연결, 확인, 삭제하는 일련의 과정을 마쳤습니다.

추가 정보

이제 시작하기 연습을 끝마쳤으므로 다음 단원을 참조하여 ElastiCache 및 사용 가능한 도구에 대한 자세한 내용을 살펴볼 수 있습니다.

- AWS 시작
- Amazon Web Services용 도구
- AWS 명령줄 인터페이스
- Amazon ElastiCache API 참조

시작하기 연습을 마친 후에는 다음 단원을 읽어 ElastiCache 관리에 대해 좀 더 자세히 알아볼 수 있습니다.

• 노드 크기 선택 (p. 67)

캐시하려는 모든 데이터를 수용할 만큼의 충분한 캐시를 원하며, 동시에 필요한 것보다 더 많은 캐시의 비용을 지불하고 싶지 않은 경우 이 단원을 읽어보면 최적의 노드 크기를 선택하는 데 도움이 됩니다.

• 캐싱 전략 및 모범 사례 (p. 399)

클러스터의 효율성에 부정적인 영향을 미칠 수 있는 문제를 확인하고 해결하십시오.

ElastiCache 자습서 및 동영상

다음은 Amazon ElastiCache 사용자가 관심을 기울이는 작업을 해결하는 자습서입니다.

- ElastiCache 동영상 (p. 28)
- 자습서: Amazon VPC에서 Amazon ElastiCache에 액세스하도록 Lambda 함수 구성

ElastiCache 동영상

다음으로, 기본 및 고급 Amazon ElastiCache 개념을 학습할 수 있는 동영상을 찾을 수 있습니다. AWS 교육 에 대한 정보는 AWS 교육 및 자격증을 참조하십시오.

주제

- 입문자용 동영상 (p. 28)
- 고급 동영상 (p. 29)

입문자용 동영상

다음 비디오는 Amazon ElastiCache에 대해 소개합니다.

주제

- DAT204—AWS NoSQL 서비스에 확장 가능한 애플리케이션 구축(re:Invent 2015) (p. 28)
- DAT207—Amazon ElastiCache를 통해 애플리케이션 성능 가속화(AWS re:Invent 2013) (p. 28)

DAT204—AWS NoSQL 서비스에 확장 가능한 애플리케이션 구축 (re:Invent 2015)

이 세션에서는 NoSQL 데이터베이스에 대한 이점을 설명하고, AWS에서 제공한 기본 NoSQL 서비스 (Amazon DynamoDB 및 Amazon ElastiCache)에 대해 알아보겠습니다. 그런 다음, 두 선도적인 고객인 Expedia와 Mapbox로부터 사용 사례와 아키텍처 문제점 및 설계 패턴과 모범 사례를 비롯한 AWS NoSQL 서 비스를 사용하여 이를 해결한 방법에 대해 듣습니다. 이 세션을 통해 NoSQL과 NoSQL의 강력한 기능을 잘 이해하여 데이터베이스 문제점을 확실하게 처리할 준비를 갖추게 됩니다.

DAT204—AWS NoSQL 서비스에 확장 가능한 애플리케이션 구축(re:Invent 2015)

DAT207—Amazon ElastiCache를 통해 애플리케이션 성능 가속화 (AWS re:Invent 2013)

이 동영상에서는 애플리케이션 성능의 속도를 높이기 위해 Amazon ElastiCache를 사용하여 인 메모리 캐시를 쉽게 배포하는 방법에 대해 알아봅니다. Amazon ElastiCache를 사용하여 애플리케이션 지연 시간을 개선하고 데이터베이스 서버에 대한 부담을 줄이는 방법을 알아봅니다. 또한 애플리케이션이 성장함에 따라 쉽게 관리 및 조정할 수 있는 캐싱 레이어를 구축하는 방법도 알아봅니다. 이 세션 중 캐싱을 활성화하여 혜택을 받을 수 있는 다양한 시나리오 및 사용 사례를 살펴보고, Amazon ElastiCache에서 제공한 기능을 설명합니다

DAT207 - Amazon ElastiCache를 통해 애플리케이션 성능 가속화(re:Invent 2013)

고급 동영상

다음 비디오는 고급 Amazon ElastiCache 주제를 다룹니다.

주제

- DAT305—Amazon ElastiCache 심층 분석(re:Invent 2017) (p. 29)
- DAT306—Amazon ElastiCache 심층 분석(re:Invent 2016) (p. 29)
- DAT317—IFTTT에서 Redis용 ElastiCache를 사용하여 이벤트를 예측하는 방법(re:Invent 2016) (p. 29)
- DAT407—Amazon ElastiCache 심층 분석(re:Invent 2015) (p. 29)
- SDD402—Amazon ElastiCache 심층 분석(re:Invent 2014) (p. 30)
- DAT307—Amazon ElastiCache 아키텍처 및 설계 패턴에 대한 심층 분석(re:Invent 2013) (p. 30)

DAT305—Amazon ElastiCache 심층 분석(re:Invent 2017)

Amazon ElastiCache의 설계 및 아키텍처에 대한 뒷이야기를 살펴봅니다. Memcached 및 Redis 솔루션으로 일반적인 설계 패턴을 살펴보고, 고객이 인 메모리 작업에 이런 패턴을 사용하여 지연 시간을 단축하고 애플 리케이션 처리량을 개선한 방법을 알아봅니다. 이 비디오 중에는 ElastiCache 모범 사례, 설계 패턴 및 안티 패턴을 검토합니다.

비디오는 다음의 내용을 소개합니다.

- Redis용 ElastiCache 온라인 리샤딩
- ElastiCache 보안 및 암호화
- Redis용 ElastiCache 버전 3.2.10

DAT305—Amazon ElastiCache 심층 분석(re:Invent 2017)

DAT306—Amazon ElastiCache 심층 분석(re:Invent 2016)

Amazon ElastiCache의 설계 및 아키텍처에 대한 뒷이야기를 살펴봅니다. Memcached 및 Redis 솔루션으로 일반적인 설계 패턴을 살펴보고, 고객이 인 메모리 작업에 이런 패턴을 사용하여 지연 시간을 단축하고 애플 리케이션 처리량을 개선한 방법을 알아봅니다. 이 세션 중에는 ElastiCache 모범 사례, 설계 패턴 및 안티 패 턴을 검토합니다.

DAT306—Amazon ElastiCache 심층 분석(re:Invent 2016)

DAT317—IFTTT에서 Redis용 ElastiCache를 사용하여 이벤트를 예측하는 방법(re:Invent 2016)

IFTTT는 단순 작업 자동화에서 주택 제어 방식 개선 등에 이르기까지, 사람들이 좋아하는 서비스를 이용해 더 많은 일을 수행할 수 있도록 지원하는 무료 서비스입니다. IFTTT는Redis용 ElastiCache를 사용하여 Amazon S3에서 로그 문서를 위한 인덱스뿐 아니라, 트랜잭션 실행 내역 및 일정 예측을 저장합니다. 이 세션을 보고 LUA의 스크립팅 성능과 Redis의 데이터 유형이 다른 것을 이용하면 불가능한 일을 어떻게 달성하게 지원하는지 알아보십시오.

DAT317—IFTTT에서 Redis용 ElastiCache를 사용하여 이벤트를 예측하는 방법(re:Invent 2016)

DAT407—Amazon ElastiCache 심층 분석(re:Invent 2015)

Amazon ElastiCache의 설계 및 아키텍처에 대한 뒷이야기를 살펴봅니다. Memcached 및 Redis 솔루션의 일 반적인 설계 패턴을 살펴보고. 고객이 인 메모리 작업에 이런 패턴을 사용하여 애플리케이션에 대한 개선된

지연 시간 및 처리량을 실현한 방법을 알아봅니다. 이 세션 중에는 Amazon ElastiCache와 관련된 모범 사례, 설계 패턴, 안티 패턴을 검토합니다.

DAT407—Amazon ElastiCache 심층 분석(re:Invent 2015)

SDD402—Amazon ElastiCache 심층 분석(re:Invent 2014)

이 동영상에서는 일반적인 캐싱 사용 사례, Memcached 및 Redis 엔진, 요구 사항에 적합한 엔진을 결정하는 데 도움이 되는 패턴, 일관적 해싱 및 기타 빠르고 확장 가능한 애플리케이션을 구축하는 수단을 살펴봅니다. Adobe의 최고 책임 과학자인 Frank Wiebe는 Adobe에서 Amazon ElastiCache를 사용하여 고객 경험을 향상하고 비즈니스를 확장하는 방법을 자세히 설명합니다.

DAT402—Amazon ElastiCache 심층 분석(re:Invent 2014)

DAT307—Amazon ElastiCache 아키텍처 및 설계 패턴에 대한 심 층 분석(re:Invent 2013)

이 동영상에서는 캐싱, 캐싱 전략, 확장, 모니터링을 살펴봅니다. 또한 Memcached 및 Redis 엔진을 비교합니다. 이 세션 중에는 Amazon ElastiCache와 관련된 모범 사례 및 설계 패턴도 검토합니다.

DAT307 - Amazon ElastiCache 아키텍처 및 설계 패턴에 대한 심층 분석(AWS re:Invent 2013).

ElastiCache로 온라인 마이그레이션

온라인 마이그레이션을 사용하면 Amazon EC2의 자체 호스팅된 Redis에서 Amazon ElastiCache로 데이터를 마이그레이션할 수 있습니다.

개요

Amazon EC2에서 실행 중인 Redis에서 Amazon ElastiCache로 데이터를 마이그레이션하려면 기존 또는 새로 생성된 Amazon ElastiCache 배포가 필요합니다. 이 배포는 즉시 마이그레이션이 가능한 구성을 가지고 있어야 합니다. 또한 인스턴스 유형 및 복제본 개수 같은 속성을 포함하여 원하는 구성과 비슷한 구성을 가져야 합니다.

Important

온라인 마이그레이션 프로세스를 시작하기 전에 다음 단원을 전부 검토하는 것이 좋습니다

StartMigration API 작업 또는 AWS CLI 명령을 호출하면 마이그레이션이 시작됩니다. 마이그레이션 프로세스는 Redis용 ElastiCache 클러스터의 마스터 노드를 EC2의 소스 Redis 클러스터에 대한 복제본으로 만듭니다. Redis 복제를 사용하면 소스 Redis와 ElastiCache 간에 데이터가 동기화됩니다. 데이터가 동기화되고 나면 ElastiCache으로 전환할 준비가 거의 완료됩니다. 이때 애플리케이션이 마이그레이션 이후에 ElastiCache를 호출할 수 있도록 애플리케이션 측에서 변경을 수행합니다.

클라이언트 측 변경을 수행할 준비가 끝나면 CompleteMigration API 작업을 호출합니다. 이 API 작업은 본 및 복제 노드를 사용하여 ElastiCache 배포를 기본 Redis 배포로 승격합니다(해당되는 경우). 이제 클라 이언트 애플리케이션을 리디렉션하여 ElastiCache에 대한 데이터 쓰기를 시작할 수 있습니다. 마이그레이션 전반에 걸쳐 Redis on EC2 노드 및 ElastiCache 기본 노드에서 redis-cli INFO 명령을 실행하여 복제 상태를 확인할 수 있습니다.

마이그레이션 단계

다음 주제에서는 데이터 마이그레이션을 위한 프로세스에 대해 개략적으로 알아봅니다.

- 마이그레이션을 위한 소스 및 대상 Redis 노드 준비 (p. 31)
- 마이그레이션 시작 (p. 32)
- 데이터 마이그레이션 진행 상황 확인 (p. 33)
- 데이터 마이그레이션 완료 (p. 33)

마이그레이션을 위한 소스 및 대상 Redis 노드 준비

ElastiCache 콘솔, API 또는 AWS CLI에서 마이그레이션을 시작하기 전에 아래 언급된 4개 사전 요구 사항모두 충족되어야 합니다.

자세한 내용은 동영상의 17:08에서 Redis on Amazon EC2에서 완전 관리형 Amazon ElastiCache로 온라인 으로 데이터 마이그레이션 시작하기를 참조하십시오.

마이그레이션을 위한 소스 및 대상 Redis 노드를 준비하려면

1. 대상 ElastiCache 배포를 식별하여 여기로 데이터를 마이그레이션할 수 있는지 확인합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 마이그레이션 시작

기존 또는 새로 생성된 ElastiCache 배포는 마이그레이션을 위해 다음과 같은 요구 사항을 충족해야 합니다.

- 사용하는 Redis 엔진 버전이 5.0.5 이상이고 클러스터 모드가 비활성화 상태여야 합니다.
- 전송 중 데이터 암호화 또는 저장 시 데이터 암호화가 활성화 되어서는 안 됩니다.
- 자동 장애 조치가 활성화된 다중 AZ가 있어야 합니다.
- Redis on EC2 인스턴스에서 데이터를 수용하기에 충분한 메모리가 있어야 합니다. 예약 메모리 설정을 올바르게 구성하는 방법은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.
- Redis 버전 2.8.21에서 Redis 버전 5.0.5 이상으로 직접 마이그레이션 할 수 있습니다.
- 2. Redis on EC2의 구성과 Redis용 ElastiCache 배포의 구성이 서로 호환되는지 확인합니다.

최소한 대상 ElastiCache 배포의 모든 구성은 Redis 복제를 위한 Redis 구성과 호환되어야 합니다.

- Redis 클러스터는 클러스터 모드가 비활성화된 구성이어야 합니다.
- Redis on EC2 인스턴스에서 Redis AUTH가 활성화 되어서는 안 됩니다.
- Redis 구성 protected-mode가 no로 설정되어 있어야 합니다.
- Redis 구성에 bind 구성이 있는 경우에는 ElastiCache 노드에서의 요청을 허용하도록 구성이 업데이 트 되어야 합니다.
- 논리적 데이터베이스의 개수는 ElastiCache 노드와 Redis on EC2 인스턴스 간에 동일해야 합니다. 이 값은 Redis 구성에서 databases를 사용해 설정됩니다.
- 데이터 복제가 성공하려면 데이터 수정을 수행하는 Redis 명령의 이름을 변경해서는 안 됩니다.
- Redis 클러스터에서 ElastiCache로 데이터를 복제하려면 이러한 추가 로드를 처리하기에 충분한 CPU 및 메모리가 있는지 확인합니다. 이러한 로드는 Redis 클러스터에서 생성된 RDB 파일에서 나와 서 네트워크를 경유해 ElastiCache 노드로 전달됩니다.
- 3. 다음과 같은 작업을 수행하여 EC2 인스턴스를 ElastiCache에 연결할 수 있는지 확인합니다.
 - EC2 인스턴스의 IP 주소가 프라이빗 주소인지 확인합니다.
 - Redis on EC2 인스턴스와 동일한 VPC에서 ElastiCache 배포를 할당 또는 생성합니다(권장 사항).
 - VPC가 다른 경우에는 두 노드 간의 액세스를 허용하도록 VPC 피어링을 설정합니다. VPC 피어링에 대한 자세한 내용은 에서 Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴 (p. 320) 단원을 참조하십시오.
 - Redis on EC2 인스턴스에 연결된 보안 그룹은 ElastiCache 노드에서의 인바운드 트래픽을 허용해야 합니다.
- 4. 데이터 마이그레이션이 완료된 이후에 애플리케이션이 ElastiCache 노드로 트래픽을 보낼 수 있는지 확인합니다. 자세한 내용은 Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴 (p. 320) 단원을 참조하십시오.

마이그레이셔 시작

모든 사전 요구 사항이 완료된 이후에 AWS Management 콘솔, ElastiCache API 또는 AWS CLI을 사용해 데이터 마이그레이션을 시작할 수 있습니다. 다음은 CLI를 사용하는 경우를 보여주는 예제입니다.

start-migration 명령을 다음 파라미터로 호출하여 마이그레이션을 시작합니다.

- --replication-group-id 대상 ElastiCache 복제 그룹의 식별자
- --customer-node-endpoint-list DNS 또는 IP 주소를 가진 엔드포인트와 소스 Redis on EC2 클 러스터가 실행 중인 포트의 목록입니다. ElastiCache에서는 현재 클러스터 모드가 비활성화된 구성만 지 원되기 때문에 이 목록에는 하나의 항목만 포함되어야 합니다. 연결 복제가 활성화된 경우에는 엔드포인트 가 Redis 클러스터의 마스터 노드 대신에 복제본을 가리킬 수 있습니다.

다음은 CLI 사용을 보여주는 예제입니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 데이터 마이그레이션 진행 상황 확인

aws elasticache start-migration --replication-group-id test-cluster -customer-node-endpoint-list "Address='10.0.0.241',Port=6379"

이 명령을 실행하면 ElastiCache 기본 노드가 Redis on EC2 인스턴스의 복제본이 되도록 자체 구성됩니다. ElastiCache 클러스터의 상태가 migrating(마이그레이션 중)으로 바뀌면서 데이터가 Redis on EC2 인스턴스에서 ElastiCache 기본 노드로 마이그레이션되기 시작합니다. Redis 인스턴스의 데이터 및 로드 크기에 따라 마이그레이션이 완료되는 데 다소 시간이 걸릴 수 있습니다. Redis on EC2 인스턴스 및 ElastiCache 기본 노드에서 redis-cli INFO 명령을 실행하여 마이그레이션 진행 상황을 확인할 수 있습니다.

복제가 성공하고 나면 Redis on EC2 인스턴스에 대한 모든 쓰기가 ElastiCache 클러스터로 전파됩니다. 읽기에는 ElastiCache 노드를 사용할 수 있습니다. 하지만 ElastiCache 클러스터에 대한 쓰기는 불가능합니다. ElastiCache 기본 노드에 다른 복제본 노드가 연결되어 있는 경우에는 이러한 복제본 노드가 ElastiCache 기본 노드에서의 복제를 계속 수행합니다. 따라서 Redis on EC2 인스턴스에서 나온 모든 데이터가 ElastiCache 클러스터의 모든 노드에 복제됩니다.

ElastiCache 기본 노드가 Redis on EC2 인스턴스의 복제본이 될 수 없는 경우에는 몇 차례 시도한 이후에야 마스터로 다시 승격이 가능합니다. ElastiCache 클러스터의 상태가 사용 가능으로 변경되고, 마이그레이션 개시 실패에 대한 복제 그룹 이벤트가 전송됩니다. 이러한 문제를 해결하려면 다음과 같이 하십시오.

- 복제 그룹 이벤트를 확인합니다. 이벤트에서 나온 모든 구체적인 정보를 이용해 마이그레이션 실패 문제를 해결합니다.
- 이벤트가 구체적인 정보를 제공하지 않는 경우에는 마이그레이션을 위한 소스 및 대상 Redis 노드 준비 (p. 31)의 지침을 준수했는지 확인합니다.
- VPC 및 서브넷을 위한 라우팅 구성이 ElastiCache 노드와 Redis on EC2 인스턴스 간의 트래픽을 허용하는지 확인합니다.
- Redis on EC2 인스턴스에 연결된 보안 그룹이 ElastiCache 노드에서의 인바운드 트래픽을 허용하는지 확 인합니다.
- 복제 실패에 대한 구체적인 정보는 Redis on EC2 인스턴스에 대한 Redis 로그를 확인합니다.

데이터 마이그레이션 진행 상황 확인

마이그레이션이 시작된 이후에 다음을 통해 진행 상황을 확인할 수 있습니다.

- Redis master_link_status가 ElastiCache 기본 노드의 INFO 명령에서 up인지 확인합니다. 이 정보는 ElastiCache 콘솔에서 확인할 수 있습니다. 클러스터를 선택하고 CloudWatch 지표에서 마스터 링크 상태를 관찰합니다. 값이 1에 도달하면 데이터가 동기화됩니다.
- Redis on EC2 인스턴스에서 INFO 명령을 실행하여 ElastiCache 복제본의 상태가 online(온라인)인지 확인할 수 있습니다. 이렇게 하면 복제 지연 시간에 대한 정보도 제공됩니다.
- Redis on EC2 인스턴스에서 CLIENT LIST Redis 명령을 사용하여 하위 클라이언트 출력 버퍼를 확인합니다.

마이그레이션이 완료되고 나면 ElastiCache 클러스터의 상태가 동기화로 표시됩니다. 이는 이제 모든 데이터가 복제되었음을 의미합니다. 데이터가 동기화되는 중에 새로운 쓰기는 Redis 인스턴스의 마스터 노드로전달됩니다.

데이터 마이그레이션 완료

ElastiCache 클러스터로 전환할 준비가 되면 다음 파라미터와 함께 complete-migration CLI 명령을 사용합니다.

• --replication-group-id- 복제 그룹의 식별자입니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 콘솔을 사용해 온라인 데이터 마이그레이션 수행

• --force - 데이터가 동기화 중인지 확인하지 않고 마이그레이션을 강제 중단하는 값입니다.

다음은 예제입니다.

aws elasticache complete-migration --replication-group-id test-cluster

이 명령을 실행하고 나면 ElastiCache 기본 노드가 Redis 인스턴스에서의 복제를 중단하고 이를 기본 노드로 승격시킵니다. 이러한 승격은 보통 몇 분 내에 완료됩니다. 기본 노드로의 승격을 확정하려면 이벤트 Complete Migration successful for test-cluster를 확인합니다. 이때 애플리케이션을 ElastiCache 쓰기 및 읽기로 보낼 수 있습니다. ElastiCache 클러스터 상태는 migrating(마이그레이션 중)에서 사용 가능으로 바뀌어야 합니다.

마스터로의 승격이 실패한 경우에도 ElastiCache 기본 노드는 Redis on EC2 인스턴스에서의 복제를 계속 수 행합니다. ElastiCache 클러스터의 상태는 계속 migrating(마이그레이션 중)으로 유지되고, 실패에 대한 복제 그룹 이벤트 메시지가 전송됩니다. 이 문제를 해결하려면 다음과 같이 하십시오.

- 복제 그룹 이벤트를 확인합니다. 이벤트에서 나온 구체적인 정보를 이용해 마이그레이션 실패 문제를 해결합니다.
- 데이터가 동기화 중이 아니라는 이벤트 메시지가 나타날 수 있습니다. 이 경우에는 ElastiCache 기본 노드가 Redis on EC2 인스턴스에서 복제를 수행할 수 있고 둘 모두가 동기화 중인지 확인합니다. 여전히 동기화를 중단하고 싶은 마음이 있으면 –force 옵션을 통해 이전의 명령을 실행할 수 있습니다.
- 노드가 대체 중이라는 이벤트 메시지가 나타날 수도 있습니다. 이는 ElastiCache 노드 중 하나에서 대체가 진행 중이었던 경우입니다. 대체 작업이 완료되고 난 후에 전체 마이그레이션 단계를 다시 시도할 수 있습니다.

콘솔을 사용해 온라인 데이터 마이그레이션 수행

AWS Management 콘솔을 사용해 EC2 인스턴스에서 Redis 클러스터로 데이터를 마이그레이션할 수 있습니다.

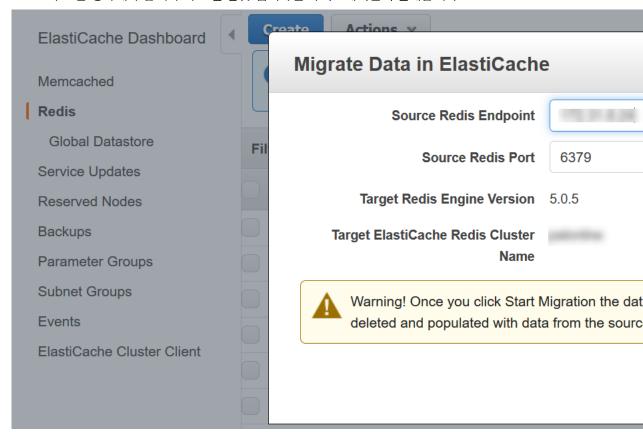
콘솔을 사용해 온라인 데이터 마이그레이션을 수행하려면

- 1. 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 새 Redis 클러스터를 생성하거나 기존 클러스터를 선택합니다. 클러스터가 다음 요구 사항을 충족하는 지 확인합니다.
 - Redis 엔진 버전은 5.0.5 이상이어야 합니다.
 - Redis 클러스터는 클러스터 모드가 비활성화된 구성이어야 합니다.
 - Redis on EC2 인스턴스에서 Redis AUTH가 활성화 되어서는 안 됩니다.
 - Redis 구성 protected-mode가 no로 설정되어 있어야 합니다.
 - Redis 구성에 bind 구성이 있는 경우에는 ElastiCache 노드에서의 요청을 허용하도록 구성이 업데이 트 되어야 합니다.
 - 데이터베이스의 개수는 ElastiCache 노드와 Redis on EC2 인스턴스 간에 동일해야 합니다. 이 값은 Redis 구성에서 databases를 사용해 설정됩니다.
 - 데이터 복제가 성공하려면 데이터 수정을 수행하는 Redis 명령의 이름을 변경해서는 안 됩니다.
 - Redis 클러스터에서 ElastiCache로 데이터를 복제하려면 이러한 추가 로드를 처리하기에 충분한 CPU 및 메모리가 있는지 확인합니다. 이러한 로드는 Redis 클러스터에서 생성된 RDB 파일에서 나와 서 네트워크를 경유해 ElastiCache 노드로 전달됩니다.
 - 클러스터는 사용 가능 상태입니다.
- 3. 클러스터를 선택했으면 작업에서 Migrate Data from Endpoint(엔드포인트에서 데이터 마이그레이션)를 선택합니다.

4. Migrate Data from Endpoint(엔드포인트에서 데이터 마이그레이션) 대화 상자에서 EC2 인스턴스의 IP 주소 또는 이름과 Redis on EC2 인스턴스를 사용할 수 있는 포트를 입력합니다.

Important

IP 주소는 정확해야 합니다. 주소를 잘못 입력하면 마이그레이션이 실패합니다.



5. Start Migration(마이그레이션 시작)을 선택합니다.

클러스터에서 마이그레이션이 시작되면 상태가 설정 변경으로 바뀐 다음, Migrating(마이그레이션 중)으로 바뀝니다.

6. 탐색 창에서 이벤트를 선택해 마이그레이션 진행 상황을 모니터링합니다.



Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 콘솔을 사용해 온라인 데이터 마이그레이션 수행

마이그레이션이 진행되는 동안 언제라도 마이그레이션을 중단할 수 있습니다. 이렇게 하려면 클러스터를 선택한 다음, 작업에서 Stop Data Migration(데이터 마이그레이션 중단)을 선택합니다. 그러면 클러스터의 상태가 사용 가능로 바뀝니다.

마이그레이션이 성공하면 클러스터의 상태가 사용 가능으로 바뀌고 이벤트 로그에 다음과 같은 내용이 표시됩니다.

Migration operation succeeded for replication group ElastiCacheClusterName.

마이그레이션이 실패하면 클러스터의 상태가 사용 가능으로 바뀌고 이벤트 로그에 다음과 같은 내용이 표시됩니다.

Migration operation failed for replication group ElastiCacheClusterName.

Redis용 ElastiCache 구현 관리

이 단원에서는 ElastiCache 구현의 다양한 구성 요소를 관리하는 방법에 관한 세부 정보를 얻을 수 있습니다. 여기에는 노드나 클러스터를 생성, 업데이트 및 삭제하는 등의 작업이 포함됩니다.

주제

- 엔진 버전 및 업그레이드 (p. 37)
- 리전 및 가용 영역 선택 (p. 47)
- 유지 관리 관리 중 (p. 50)
- 노드 관리 (p. 51)
- ElastiCache 클러스터 관리 (p. 64)
- 클러스터 또는 복제 그룹에 액세스 (p. 104)
- 샤드 작업 (p. 111)
- 글로벌 데이터 스토어를 사용한 AWS 리전 간 복제 (p. 114)
- 고가용성을 위한 복제 그룹 사용 (p. 127)
- 연결 엔드포인트 찾기 (p. 180)
- Redis용 ElastiCache 백업 및 복원 (p. 187)
- Redis용 ElastiCache 클러스터 조정 (p. 219)
- 파라미터 그룹을 사용해 엔진 파라미터 구성 (p. 260)

엔진 버전 및 업그레이드

이 단원에서는 지원되는 Redis 엔진 버전과 이를 업그레이드하는 방법을 설명합니다.

주제

- 지원되는 Redis용 ElastiCache 버전 (p. 37)
- 엔진 버전 업그레이드 (p. 45)

지원되는 Redis용 ElastiCache 버전

Redis용 Amazon ElastiCache를 사용해 HIPAA 규정 준수 애플리케이션을 구축할 수 있습니다. 이렇게 하려면, Redis 클러스터를 생성할 때 Redis용 ElastiCache 버전 3.2.6, 4.0.10 이상을 사용하여 Redis 클러스터를 생성할 때 미사용 데이터 암호화, 전송 중 데이터 암호화 및 Redis AUTH를 활성화할 수 있습니다. 예를 들어 AWS와 체결하여 이행 중인 비즈니스 제휴 계약(AWS)에 따라 보호 대상 건강 정보(PHI)를 비롯한 의료 관련 정보를 저장할 수 있습니다. AWS 범위 내 서비스는 외부 감사 기관의 철저한 평가를 거쳐 인증, 규정 준수 증명 또는 운영 권한(ATO)을 받았습니다. 자세한 내용은 다음 항목을 참조하십시오.

- AWS 클라우드 규정 준수
- HIPAA 규정 준수
- 규정 준수 프로그램 제공 AWS 범위 내 서비스
- Redis용 ElastiCache 규정 준수 (p. 391)
- Amazon ElastiCache의 데이터 보안 (p. 299)
- Redis AUTH 명령을 사용하여 사용자 인증 (p. 313)

지원되는 Redis용 ElastiCache 버전

• Redis용 ElastiCache 버전 5.0.6(개선됨) (p. 39)

- Redis용 ElastiCache 버전 5.0.5(개선됨) (p. 39)
- Redis용 ElastiCache 버전 5.0.4(개선됨) (p. 39)
- Redis용 ElastiCache 버전 5.0.3(개선됨) (p. 39)
- Redis용 ElastiCache 버전 5.0.0(개선됨) (p. 40)
- Redis용 ElastiCache 버전 4.0.10(개선됨) (p. 41)
- Redis용 ElastiCache 버전 3.2.10(개선됨) (p. 41)
- Redis용 ElastiCache 버전 3.2.6(개선됨) (p. 42)
- Redis용 ElastiCache 버전 3.2.4(개선됨) (p. 42)
- Redis용 ElastiCache 버전 2.8.24(개선됨) (p. 43)
- Redis용 ElastiCache 버전 2.8.23(개선됨) (p. 43)
- Redis용 ElastiCache 버전 2.8.22(개선됨) (p. 44)
- Redis용 ElastiCache 버전 2.8.21 (p. 44)
- Redis용 ElastiCache 버전 2.8.19 (p. 44)
- Redis용 ElastiCache 버전 2.8.6 (p. 44)
- Redis용 ElastiCache 버전 2.6.13 (p. 45)

Note

새로운 Redis 버전은 보다 향상되고 안정적인 사용자 환경을 제공하므로 ElastiCache 콘솔을 사용할 경우 Redis 버전 2.6.13, 2.8.6 및 2.8.19는 더 이상 지원되지 않습니다. 이러한 Redis 버전은 사용하지 않을 것을 권장합니다. 이러한 버전 중 하나를 사용해야 하는 경우 AWS CLI 또는 ElastiCache API에서 작업하십시오.

자세한 정보는 다음 항목을 참조하십시오.

	AWS CLI	ElastiCache API
클러스터 생성	클러스터 생성 (AWS CLI) (p. 76)	클러스터 생성 (ElastiCache API) (p. 77)
	클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.	클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.
클러스터 수정	AWS CLI 사용 (p. 88) 클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.	ElastiCache API 사용 (p. 89) 클러스터 모드가 활성화된 상태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니다.
복제 그룹 생성	Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 149) Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154)	Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성 (p. 151) Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158)
복제 그룹 수정	AWS CLI 사용 (p. 166)	ElastiCache API 사용 (p. 166)

Redis용 ElastiCache 버전 5.0.6(개선됨)

Redis용 Amazon ElastiCache에서는 Amazon ElastiCache가 지원하는 Redis 엔진의 다음 버전을 새로 제공합니다. 여기에는 버그 수정이 포함됩니다.

자세한 내용은 GitHub의 Redis에서 Redis 5.0.6 출시 정보를 참조하십시오.

Redis용 ElastiCache 버전 5.0.5(개선됨)

Redis 용 Amazon ElastiCache는 Amazon ElastiCache가 지원하는 다음 버전의 Redis 엔진을 새로 제공합니다. 이 업데이트에는 모든 계획된 작업 중에 자동 장애 조치 클러스터의 Redis용 ElastiCache에 대한 온라인 구성 변경 사항이 포함되어 있습니다. 이제 클러스터가 온라인 상태에서 들어오는 요청을 계속 처리하는 동안 클러스터의 규모를 조정하고, Redis 엔진 버전을 업그레이드하고, 패치 및 유지 관리 업데이트를 적용할수 있습니다. 여기에는 버그 수정도 포함되어 있습니다.

자세한 내용은 GitHub의 Redis의 Redis 5.0.5 출시 정보를 참조하십시오.

Redis용 ElastiCache 버전 5.0.4(개선됨)

Redis용 Amazon ElastiCache는 Amazon ElastiCache가 지원하는 다음 버전의 Redis 엔진을 새로 제공합니다. 다음과 같은 향상된 기능을 포함합니다.

- 특별한 조건에서 엔진 안정성 보장.
- 향상된 Hyperloglog 오류 처리.
- 안정적인 복제를 위한 향상된 핸드셰이크 명령
- XCLAIM 명령을 통한 일관된 메시지 배달 추적.
- 객체에서의 향상된 LFU 필드 관리.
- ZPOP 사용 시 향상된 트랜잭션 관리.

자세한 내용은 GitHub의 Redis의 Redis 5.0.4 릴리스 노트를 참조하십시오.

Redis용 ElastiCache 버전 5.0.3(개선됨)

Redis용 Amazon ElastiCache는 Amazon ElastiCache가 지원하는 다음 버전의 Redis 엔진을 새로 제공합니다. 다음과 같은 향상된 기능을 포함합니다.

- 정렬된 세트 엣지 케이스, 정확한 메모리 사용량 등을 개선하는 버그 픽스. 자세한 내용은 Redis 5.0.3 출시 정보를 참조하십시오.
- 명령 이름 변경 기능: ElastiCache for Redis 5.0.3에는 위험할 수 있거나 비용이 높은 Redis 명령 (FLUSHALL 또는 FLUSHDB 등과 같이 데이터 손실 사고를 유발할 수 있는 명령)을 이름을 변경할 수 있는 rename-commands라는 파라미터가 새로 포함되었습니다. 이것은 오픈 소스 Redis의 rename-command 구성과 비슷합니다. 하지만 ElastiCache는 완전 관리형 워크플로를 제공함으로써 보다 향상된 경험을 제공합니다. 명령 이름 변경은 즉시 적용되며, 명령 목록을 포함하는 클러스터의 모든 노드에 자동으로 전파됩니다. 사용자의 개입(노드 재부팅 등)은 필요 없습니다.

다음 예제에서는 기존 파라미터 그룹을 수정하는 방법을 보여줍니다. 이러한 예제에는 이름을 변경하려는 명령 목록(공백으로 구분)인 rename-commands 파라미터가 포함됩니다.

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-
name <a href="mailto:custom_param_group">custom_param_group</a>
--parameter-name-values "ParameterName=rename-commands, ParameterValue='flushall restrictedflushall'" --region <a href="mailto:region">region</a>
```

이 예제에서는 rename-commands 파라미터를 사용하여 flushall 명령을 restrictedflushall로 이름 변경합니다.

여러 명령의 이름을 변경하려면 다음을 사용하십시오.

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-
name custom_param_group
--parameter-name-values "ParameterName=rename-commands, ParameterValue='flushall
restrictedflushall flushdb restrictedflushdb''" --region region
```

변경을 되돌리려면 다음과 같이 명령을 다시 실행하고, 유지하려는 ParameterValue 목록에서 이름 변경된 값을 제외시킵니다.

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-
name custom_param_group
--parameter-name-values "ParameterName=rename-commands, ParameterValue='flushall
restrictedflushall'" --region region
```

이 경우, flushall 명령은 restrictedflushall로 이름이 변경되고 이름 변경된 다른 명령은 원래 명령 이름으로 되돌려집니다.

Note

명령 이름 변경 시 다음과 같은 제한이 따릅니다.

- 이름 변경된 모든 명령은 영숫자여야 합니다.
- 새 명령 이름의 최대 길이는 20자(영숫자)입니다.
- 명령 이름을 변경할 경우 해당 클러스터와 연결된 파라미터 그룹을 업데이트해야 합니다.
- 명령 사용을 전체적으로 차단하려면 다음과 같이 blocked 키워드를 사용합니다.

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-
name <u>custom_param_group</u>
--parameter-name-values "ParameterName=rename-commands,
ParameterValue='flushall blocked'" --region <u>region</u>
```

파라미터 변경에 대한 정보와 이름을 변경할 수 있는 명령 목록을 보려면 Redis 5.0.3 파라미터 변경 사항 (p. 275) 단원을 참조하십시오.

Redis용 ElastiCache 버전 5.0.0(개선됨)

Redis용 Amazon ElastiCache에서는 Amazon ElastiCache가 지원하는 Redis 엔진의 다음 메이저 버전을 도입합니다. Redis용 ElastiCache 5.0.0은 다음 개선 사항을 지원합니다.

- Redis 스트림: 이 모델에서는 생산자가 실시간으로 새 항목을 추가할 수 있는 로그 데이터 구조를 모델링합니다. 또한 소비자가 차단 또는 차단하지 않는 방식으로 메시지를 소비할 수 있습니다. 또한 스트림을 사용하여 클라이언트 그룹을 대표하는 소비자 그룹이 Apache Kafka와 비슷한 메시지 스트림의 서로 다른 부분을 공동으로 사용할 수 있습니다. 자세한 내용은 Introduction to Redis Streams를 참조하십시오.
- XADD, XRANGE 및 XREAD와 같은 스트림 명령군 지원. 자세한 내용은 Redis Streams Commands를 참조하십시오.
- 새 파라미터 및 이름이 변경된 파라미터의 수. 자세한 내용은 Redis 5.0.0 파라미터 변경 사항 (p. 276) 단원을 참조하십시오.
- 새로운 Redis 지표인 StreamBasedCmds.
- Redis 노드의 스냅샷 시간 단축.

Important

Redis용 Amazon ElastiCache는 Redis 오픈 소스 버전 5.0.1의 두 가지 중요한 버그 수정을 백포팅 했습니다. 해당되는 사항은 다음과 같습니다.

- 특정 키가 이미 만료되면 RESTORE 불일치가 회신됩니다.
- XCLAIM 명령은 잠재적으로 잘못된 항목을 반환하거나 프로토콜을 동기화 해제할 수 있습니다.

이 두 가지 버그 수정은 Redis 엔진 버전 5.0.0의 Redis용 ElastiCache 지원에 포함되어 있으며 향후 버전 업데이트에서 사용됩니다.

Redis용 ElastiCache 버전 4.0.10(개선됨)

Redis용 Amazon ElastiCache에서는 Amazon ElastiCache가 지원하는 Redis 엔진의 다음 메이저 버전을 도입합니다. Redis용 ElastiCache 4.0.10은 다음 개선 사항을 지원합니다.

- 단일 Redis용 ElastiCache 버전에서 온라인 클러스터 크기 조정 및 암호화가 모두 지원. 자세한 내용은 다음 자료를 참조하십시오.
 - Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241)
 - Redis(클러스터 모드 활성화됨)를 위한 온라인 리샤딩 및 샤드 재분배 (p. 243)
 - Amazon ElastiCache의 데이터 보안 (p. 299)
- 새 파라미터의 수입니다. 자세한 내용은 Redis 4.0.10 파라미터 변경 사항 (p. 279) 단원을 참조하십시오.
- MEMORY와 같은 메모리 명령군 지원. 자세한 내용은 Redis Commands(MEMO에서 검색)를 참조하십시오.
- 온라인 상태에서 메모리 조각 모음을 지원하여 더욱 효율적인 메모리 사용률과 데이터에 대해 더 많이 사용가능한 메모리가 허용됩니다.
- 비동기식 플러시 및 삭제를 지원합니다. Redis용 ElastiCache에서는 UNLINK, FLUSHDB 및 FLUSHALL과 같은 명령을 지원하여 기본 스레드와 다른 스레드에서 실행할 수 있습니다. 이렇게 하면 비동기식으로 메모리를 확보하여 애플리케이션의 성능 및 응답 시간을 향상시킬 수 있습니다.
- 새로운 Redis 지표인 ActiveDefragHits. 자세한 내용은 Redis 지표 단원을 참조하십시오.

Redis 버전 3.2.10을 실행하는 Redis(클러스터 모드 비활성화됨) 사용자는 콘솔을 사용하여 온라인 업그레이드를 통해 클러스터를 업그레이드할 수 있습니다.

Redis용 ElastiCache 클러스터 크기 조정 및 암호화 지원 비교

기능	3.2.6	3.2.10	4.0.10 이상
온라인 클러스터 크기 조정 *	아니요	예	예
전송 중 데이터 암호화 **	예	아니요	예
미사용 데이터 암호화 **	예	아니요	예

^{*} 샤드 추가, 제거 및 재분배

Redis용 ElastiCache 버전 3.2.10(개선됨)

Redis용 Amazon ElastiCache에서는 Amazon ElastiCache가 지원하는 Redis 엔진의 다음 메이저 버전을 도입합니다. Redis용 ElastiCache 3.2.10에서는 온라인 클러스터 크기 조정을 도입해 클러스터에서 샤드를 추가 또는 제거하고 동시에 수신되는 I/O 요청을 계속해서 처리합니다. Redis용 ElastiCache 3.2.10 사용자는

^{**} FedRAMP, HIPAA 및 PCI DSS 준수 애플리케이션에 필요합니다. 자세한 내용은 Redis용 ElastiCache 규정 준수 (p. 391) 단원을 참조하십시오.

데이터를 암호화하는 옵션을 제외한 이전 Redis 버전의 모든 기능을 사용할 수 있습니다. 이 기능은 현재 버전 3.2.6에서만 사용할 수 있습니다.

Redis용 ElastiCache 버전 3.2.6 및 3.2.10 비교

기능	3.2.6	3.2.10	
온라인 클러스터 크기 조정 *	아니요	예	
전송 중 데이터 암호화 **	예	아니요	
미사용 데이터 암호화 **	예	아니요	
- 나는 중의 게이 마케팅에			

^{*} 샤드 추가, 제거 및 재분배

자세한 내용은 다음 자료를 참조하십시오.

- Redis(클러스터 모드 활성화됨)를 위한 온라인 리샤딩 및 샤드 재분배 (p. 243)
- 모범 사례: 온라인 클러스터 크기 조정 (p. 410)

Redis용 ElastiCache 버전 3.2.6(개선됨)

Redis용 Amazon ElastiCache에서는 Amazon ElastiCache가 지원하는 Redis 엔진의 다음 메이저 버전을 도 입합니다. Redis용 ElastiCache 3.2.6 사용자는 데이터를 암호화하는 옵션 이외에도 이전 Redis 버전의 모든 기능을 사용할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.

- Redis용 ElastiCache 전송 중 데이터 암호화(TLS) (p. 300)
- Redis용 ElastiCache에서 미사용 데이터 암호화 (p. 306)
- HIPAA 자격 획득 (p. 393)

Redis용 ElastiCache 버전 3.2.4(개선됨)

Redis용 Amazon ElastiCache 버전 3.2.4에서는 Amazon ElastiCache가 지원하는 Redis 엔진의 다음 메이저 버전을 도입합니다. Redis용 ElastiCache 3.2.4 사용자는 사용할 수 있는 이전 Redis 버전의 모든 기능과 클 러스터 모드 또는 비클러스터 모드에서 실행할 수 있는 옵션을 보유합니다. 다음 표에는 이에 대해 요약되어 있습니다.

Redis 3.2.4 비클러스터 모드와 클러스터 모드 비교

기능	비클러스터 모드	클러스터 모드
데이터 파티셔닝	아니요	예
지역 검색 인덱싱	예	예
노드 유형 변경	예	여 *
복제본 조정	예	여 *
스케일 아웃	아니요	여 *
데이터베이스 지원	Multiple	단일
파라미터 그룹	default.redis3.2 ** d	efault.redis3.2.cluster.on *

^{**} FedRAMP, HIPAA 및 PCI DSS 준수 애플리케이션에 필요합니다. 자세한 내용은 Redis용 ElastiCache 규정 준수 (p. 391) 단원을 참조하십시오.

기능	클러스터 모드			
* 선택적으로 클러스터 크기를 조정				
** 또는 해당 그룹에서 파생된 파라미터.				

참고:

- 분할 각 노드 그룹에 대한 복제 지원을 통해 데이터를 2 ~ 90개의 노드 그룹(샤드)으로 분할할 수 있는 기능입니다.
- 지역 검색 인덱싱 Redis 3.2에서는 GEO 명령 6개를 통한 지역 검색 인덱싱의 지원을 도입합니다. 자세한 내용은 Redis 명령 페이지의 Redis GEO* 명령 설명서 Redis 명령: GEO를 참조하십시오(GEO에 대해 필터링됨).

추가 Redis 3 기능에 대한 자세한 내용은 Redis 3.2 출시 정보 및 Redis 3.0 출시 정보를 참조하십시오.

현재 ElastiCache 관리형 Redis(클러스터 모드 활성화됨)는 다음 Redis 3.2 기능을 지원하지 않습니다.

- 복제본 마이그레이션
- 클러스터 재분배
- Lua 디버거

ElastiCache는 다음 Redis 3.2 관리 명령을 비활성화합니다.

- · cluster meet
- · cluster replicate
- cluster flushslots
- cluster addslots
- cluster delslots
- cluster setslot
- · cluster saveconfig
- cluster forget
- cluster failover
- cluster bumpepoch
- cluster set-config-epoch
- · cluster reset

Redis 3.2.4 파라미터에 대한 자세한 내용은 Redis 3.2.4 파라미터 변경 사항 (p. 282) 단원을 참조하십시오.

Redis용 ElastiCache 버전 2.8.24(개선됨)

버전 2.8.23부터 추가된 Redis 개선 사항에는 버그 수정 및 불량 메모리 액세스 주소의 로깅이 포함됩니다. 자세한 내용은 Redis 2.8 출시 정보를 참조하십시오.

Redis용 ElastiCache 버전 2.8.23(개선됨)

버전 2.8.22부터 추가된 Redis 개선 사항에는 버그 수정이 포함됩니다. 자세한 내용은 Redis 2.8 출시 정보를 참조하십시오. 이 릴리스에는 새 파라미터 close-on-slave-write에 대한 지원도 포함됩니다. 이 파라미터가 활성화되면 읽기 전용 복제본에 쓰려고 시도하는 클라이언트를 연결 해제합니다.

Redis 2.8.23 파라미터에 대한 자세한 내용은 ElastiCache 사용 설명서의 Redis 2.8.23(개선 됨) 추가 파라미터 (p. 285) 단원을 참조하십시오.

Redis용 ElastiCache 버전 2.8.22(개선됨)

버전 2.8.21부터 추가된 Redis 개선 사항에는 다음이 포함됩니다.

- 백업 오버헤드에 대해 메모리를 적게 할당하고 애플리케이션에 많이 할당할 수 있는 forkless 백업 및 동기화에 대해 지원합니다. 자세한 내용은 동기화 및 백업 구현 방법 (p. 143) 단원을 참조하십시오. forkless 프로세스는 지연 시간과 처리량 모두에 영향을 줄 수 있습니다. 높은 쓰기 처리량의 경우 복제본이 다시 동기화되면, 동기화되는 전체 시간에 대해 접속 불가능하게 될 수 있습니다.
- 장애 조치가 발생한 경우, 가능하면 언제든지 복제본이 기본 항목과 전체 동기화가 아닌 부분적인 동기화를 수행하므로 이제 복제 그룹이 더 빠르게 복구됩니다. 또한, 기본 항목 및 복제본 모두 동기화 중 더 이상디스크를 사용하지 않으므로 속도가 향상됩니다.
- 두 개의 새 CloudWatch 지표 지원.
 - ReplicationBytes 읽기 전용 복제본으로 전송되는 복제 그룹 기본 클러스터의 바이트 수.
 - SaveInProgress 백그라운드 저장 프로세스가 실행 중인지 여부를 나타내는 이진 값.

자세한 내용은 CloudWatch 측정치를 통한 모니터링 사용 (p. 359) 단원을 참조하십시오.

- 복제 PSYNC 동작에서 중요한 버그 수정의 수. 자세한 내용은 Redis 2.8 출시 정보를 참조하십시오.
- 다중 AZ 복제 그룹에서 향상된 복제 성능을 유지하고 증가된 클러스터 안정성을 유지하기 위해 비 ElastiCache 복제본이 더 이상 지원되지 않습니다.
- 복제 그룹에서 기본 클러스터와 복제본 간의 데이터 일관성을 향상하기 위해 복제본에서는 기본 클러스터 와 별도로 더 이상 키를 제거하지 않습니다.
- Redis 구성 변수 appendonly 및 appendfsync는 Redis 버전 2.8.22 이상에서 지원되지 않습니다.
- 메모리가 부족한 상황에서 큰 출력 버퍼가 있는 클라이언트는 복제본 클러스터에서 연결이 해제될 수 있습니다. 연결이 해제되면 클라이언트가 다시 연결해야 합니다. 이러한 상황은 대부분 PUBSUB 클라이언트에 대해 발생합니다.

Redis용 ElastiCache 버전 2.8.21

버전 2.8.19부터 추가된 Redis 개선 사항에는 여러 가지 버그 수정이 포함됩니다. 자세한 내용은 Redis 2.8 출시 정보를 참조하십시오.

Redis용 ElastiCache 버전 2.8.19

버전 2.8.6부터 추가된 Redis 개선 사항에는 다음이 포함됩니다.

- HyperLogLog에 대해 지원합니다. 자세한 내용은 Redis 새 데이터 구조: HyperLogLog를 참조하십시오.
- 정렬된 세트 데이터 유형은 이제 ZRANGEBYLEX, ZLEXCOUNT 및 ZREMRANGEBYLEX의 새 명령을 통해 사전 순 범위 쿼리를 지원합니다.
- 기본 노드에서 복제본 노드로 부실 데이터가 전송되는 것을 방지하기 위해 백그라운드 저장(bgsave) 하위 프로세스가 중단될 경우 마스터 SYNC가 실패합니다.
- HyperLogLogBasedCommands CloudWatch 지표 지원. 자세한 내용은 Redis 지표 (p. 360) 단원을 참조하십시오.

Redis용 ElastiCache 버전 2.8.6

버전 2.6.13부터 추가된 Redis 개선 사항에는 다음이 포함됩니다.

- 읽기 전용 복제본에 대한 복원성 및 내결함성이 개선되었습니다.
- 부분적 재동기화를 지원합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 엔진 버전 업그레이드

- 항상 사용할 수 있어야 하는 읽기 전용 복제본의 사용자 정의 최소 숫자를 지원합니다.
- 게시/구독에 대한 전체 지원—서버에서의 이벤트를 클라이언트에 알리는 기능입니다.
- 기본 노드 장애의 자동 감지 및 기본 노드에서 보조 노드로 장애 조치

Redis용 ElastiCache 버전 2.6.13

Redis 버전 2.6.13은 Redis용 Amazon ElastiCache가 지원하는 초기 Redis 버전이었습니다. 자동 장애 조치가 포함된 다중 AZ는 Redis 2.6.13에서 지원되지 않습니다.

엔진 버전 업그레이드

사용자는 캐시 클러스터를 실행하는 프로토콜 표준 소프트웨어를 ElastiCache에서 제공하는 새 버전으로 업 그레이드할지 여부와 그 시기를 조정할 수 있습니다. 이 제어 수준을 사용하면 특정 버전과의 호환성을 유지 하고, 프로덕션에 배포하기 전에 애플리케이션으로 새 버전을 테스트하고, 원하는 조건과 일정에 맞춰 버전 업그레이드를 수행할 수 있습니다.

버전 업그레이드에는 약간의 호환성 위험이 있을 수 있으므로 업그레이드가 자동으로 이루어지지 않기 때문에 업그레이드는 사용자가 시작해야 합니다.

클러스터 또는 복제 그룹을 수정하고 새 엔진 버전을 지정하여 엔진 버전 업그레이드를 시작합니다. 자세한 내용은 다음 자료를 참조하십시오.

- ElastiCache 클러스터 수정 (p. 87)
- 복제 그룹 수정 (p. 165)

Important

- 최신 엔진 버전으로 업그레이드할 수 있지만 이전 엔진 버전으로 다운그레이드할 수 없습니다. 이전 엔진 버전을 사용하려면, 기존 클러스터를 삭제하고 이전 엔진 버전을 통해 새로 생성해야 합니다.
- 엔진 버전 관리는 패치 발생 방법을 최대한 제어할 수 있도록 설계되었습니다. 그러나 ElastiCache는 시스템 또는 캐시 소프트웨어에 심각한 보안 취약성이 발견되는 등 발생할 가능성 이 거의 없는 이벤트의 경우 사용자를 대신하여 클러스터에 패치를 적용할 수 있는 권한을 보유합 니다
- Redis 엔진 버전 5.0.5부터는 가동 중지 시간을 최소화하면서 클러스터 버전을 업그레이드할 수 있습니다. 전체 업그레이드 과정 중에도 클러스터를 읽을 수 있으며, 몇 초 정도 시간이 걸리는 장애 조치 작업 중인 경우를 제외하면 대부분 업그레이드 기간 중에 쓰기도 가능합니다.
- 5.0.5 이전 버전으로 ElastiCache 클러스터를 업그레이드할 수도 있습니다. 관련된 프로세스는 동일하지만 DNS 전파 중에 장애 조치 시간이 더 길어질 수 있습니다(30s-1m).
- Redis용 ElastiCache는 Redis(클러스터 모드 비활성화됨)와 Redis(클러스터 모드 활성화됨) 간의 전환을 지원하지 않습니다.
- Redis용 Amazon ElastiCache 엔진 업그레이드 프로세스는 기존 데이터를 최대한 유지할 수 있도록 진행되며, 성공적인 Redis 복제가 필요합니다.
- 엔진을 업그레이드할 때 Redis(클러스터 모드 비활성화됨)에서 Redis(클러스터 모드 활성화됨)로 직접 업그레이드할 수 없습니다. 다음 절차에서는 Redis(클러스터 모드 비활성화됨)에서 Redis(클러스터 모드 활성화됨)로 업그레이드하는 방법을 보여줍니다.

Redis(클러스터 모드 비활성화됨)에서 Redis(클러스터 모드 활성화됨)로 엔진 버전을 업 그레이드하려면

- 1. Redis(클러스터 모드 비활성화됨) 클러스터 또는 복제 그룹에 대한 백업을 만듭니다. 자세한 내용은 수동 백업 만들기 (p. 190) 단원을 참조하십시오.
- 2. 백업을 사용하여 샤드가 하나인 Redis(클러스터 모드 활성화됨) 클러스터(노드 그룹)를 만들고 시드합니다. 클러스터 또는 복제 그룹을 생성할 때 새 엔진 버전을 지정하고 클러스터 모

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 엔진 버전 업그레이드

드를 활성화합니다. 자세한 내용은 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211) 단원을 참조하십시오.

- 3. 이전 Redis(클러스터 모드 비활성화됨) 클러스터 또는 복제 그룹을 삭제합니다. 자세한 내용은 클러스터 삭제하기 (p. 102) 또는 복제 그룹 삭제 (p. 166) 단원을 참조하십시오.
- 4. 새 Redis(클러스터 모드 활성화됨) 클러스터 또는 복제 그룹을 필요한 샤드(노드 그룹) 수까지 확장합니다. 자세한 내용은 Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241) 단원을 참조하십시오.
- 단일 Redis 클러스터 및 다중 AZ가 비활성화된 클러스터의 경우 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)에 명시된 대로 Redis에 충분한 메모리를 사용할 수 있도록 하는 것이 좋습니다. 이러한 경우 업그레이드 프로세스 중에는 서비스 요청에 기본 항목을 사용할 수 없습니다.
- 다중 AZ가 활성화된 Redis 클러스터의 경우, 수신 쓰기 트래픽이 낮은 기간 동안 엔진 업그레이 드를 예약하는 것이 좋습니다. Redis 5.0.5 이상으로 업그레이드하면 업그레이드 프로세스 동안 기본 클러스터를 서비스 요청에 계속 사용할 수 있습니다. Redis 5.0.4 이하로 업그레이드하면 DNS 업데이트와 관련하여 몇 초 동안 간단한 중단이 발생할 수 있습니다.

샤드가 여러 개인 클러스터 및 복제 그룹은 다음과 같이 처리 및 패치가 적용됩니다.

- 모든 샤드는 병렬로 처리됩니다. 언제든지 하나의 샤드에서 오직 하나의 업그레이드 작업이 수행됩니다.
- 각 샤드에서 기본 복제본이 처리되기 전에 모든 복제본이 처리됩니다. 하나의 샤드에 복제본이 적게 있는 경우에는 다른 샤드의 복제본의 처리가 완료되기 전에 해당 샤드의 기본 복제본이 처 리됩니다.
- 모든 샤드에서 기본 노드가 연속하여 처리됩니다. 한번에 오직 하나의 기본 노드가 업그레이드 됩니다.
- 현재 클러스터 또는 복제 그룹에서 암호화가 활성화되어 있는 경우에는 암호화를 지원하지 않는 엔진 버전으로 업그레이드할 수 없습니다(예를 들면 3.2.6에서 3.2.10로 업그레이드 불가능).

엔진 버전 업그레이드 방법

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 통해 수정하고 최신 엔진 버전을 지정하여 클러스터 또 는 복제 그룹에 대한 버전 업그레이드를 시작합니다. 자세한 내용은 다음 항목을 참조하십시오.

클러스터 및 복제 그룹을 수정하는 방법		
AWS Management 콘솔 사용 (p. 87) AWS Management 콘솔 사용 (p. 166)		
AWS CLI 사용 (p. 88) AWS CLI 사용 (p. 166)		
ElastiCache API 사용 (p. 89) ElastiCache API 사용 (p. 166)		

차단된 Redis 엔진 업그레이드 해결

다음 표에 표시된 대로 대기 중인 스케일 업 작업이 있는 경우 Redis 엔진 업그레이드 작업이 차단됩니다.

대기 중 작업	차단된 작업
스케일 업	즉시 엔진 업그레이드
엔진 업그레이드	즉시 스케일 업
스케일 업 및 엔진 업그레이드	즉시 스케일 업
스케일 밥 및 엔진 밥그데이드	즉시 엔진 업그레이드

차단된 Redis 엔진 업그레이드를 해결하려면

- 다음 중 하나를 수행하십시오.
 - 즉시 적용 확인란을 선택 취소하여 다음 유지 관리 기간에 대해 Redis 엔진 업그레이드 작업을 예약합니다.

CLI의 경우, --no-apply-immediately를 사용합니다. API의 경우, ApplyImmediately=false를 사용합니다.

- Redis 엔진 업그레이드 작업을 수행하기 위해 다음 유지 관리 기간(또는 그 이후)까지 기다립니다.
- 즉시 적용 확인란을 선택한 채로 이 클러스터 수정 사항에 Redis 확장 작업을 추가합니다.

CLI의 경우, --apply-immediately를 사용합니다. API의 경우, ApplyImmediately=true를 사용합니다.

이러한 접근 방식에서는 이를 즉시 수행하여 다음 유지 관리 기간 동안 엔진 업그레이드를 효과적으로 취소합니다.

리전 및 가용 영역 선택

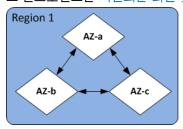
AWS 클라우드 컴퓨팅 리소스는 가용성이 매우 높은 데이터 설비에 있습니다. 추가 확장성 및 안정성을 제공하기 위해 이러한 데이터 센터 시설은 여러 물리적 위치에 배치됩니다. 이러한 위치는 지역 및 가용 영역으로 분류됩니다.

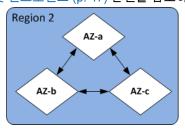
AWS 리전은 크고 광범위하게 별도의 지리적 위치에 분산되어 있습니다. 가용 영역은 다른 가용 영역에서 발생한 장애로부터 격리할 수 있도록 설계된 AWS 리전 내 개별적인 위치입니다. 가용 영역은 같은 AWS 리전에 있는 다른 가용 영역에 대해 저렴하고 지연 시간이 짧은 네트워크 연결을 제공합니다.

Important

각 리전은 완전히 독립적입니다. 시작하는 모든 ElastiCache 활동(예: 클러스터 생성)은 현재 기본 리전에서만 실행됩니다.

특정 리전에서 클러스터를 생성하거나 사용하려면 해당하는 리전 서비스 엔드포인트를 사용하십시오. 서비스 엔드포인트는 지원되는 리전 및 엔드포인트 (p. 47) 단원을 참조하십시오.





리전 및 가용 영역

주제

- 지원되는 리전 및 엔드포인트 (p. 47)
- 노드 찾기 (p. 50)

지원되는 리전 및 엔드포인트

Amazon ElastiCache는 여러 AWS 리전에서 사용할 수 있습니다. 따라서 요구 사항에 적합한 위치에서 ElastiCache 클러스터를 시작할 수 있습니다. 예를 들어, 고객과 가장 가까운 AWS 리전 또는 특정 법적 요구 사항을 준수하는 AWS 리전에서 시작할 수 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 지원되는 리전 및 엔드포인트

기본적으로 AWS SDK, AWS CLI, ElastiCache API 및 ElastiCache 콘솔은 미국 서부(오레곤) 리전을 참조합니다. ElastiCache를 새 리전에서 사용할 수 있게 됨에 따라 이러한 리전의 새 엔드포인트 또한 고유 HTTP 요청, AWS SDK, AWS CLI 및 콘솔에서 사용할 수 있습니다.

각 리전은 다른 리전에서 완전히 격리되도록 설계되었습니다. 각 리전 안에는 가용 영역(AZ)가 여러 개 있습니다. 서로 다른 가용 영역에서 노드를 시작하면 가능한 최고 수준의 내결함성을 갖출 수 있습니다. 리전 및 가용 영역에 대한 자세한 내용은 이 주제의 맨 위에 있는 리전 및 가용 영역 선택 (p. 47) 단원을 참조하십시오.

ElastiCache가 지원되는 리전

리전 이름/리전	엔드포인트	프로토콜	
미국 동부(오하이오) 리 전	elasticache.us- east-2.amazonaws.co	HTTPS m	
us-east-2			
미국 동부(버지니아 북 부) 지역	elasticache.us- east-1.amazonaws.co	HTTPS m	
us-east-1			
미국 서부(캘리포니아 북 부) 리전	elasticache.us- west-1.amazonaws.co	HTTPS m	
us-west-1			
미국 서부(오레곤) 지역	elasticache.us-	HTTPS	
us-west-2	west-2.amazonaws.co	m	
캐나다(중부) 리전	elasticache.ca-	HTTPS	
ca-central-1	central-1.amazonaws	.com	
아시아 태평양(뭄바이) 리전	elasticache.ap- south-1.amazonaws.c	HTTPS om	
ap-south-1			
아시아 태평양(도쿄) 리 전 ap-northeast-1	elasticache.ap- northeast-1.amazona	HTTPS ws.com	
아시아 태평양(서울) 리전	elasticache.ap- northeast-2.amazona	HTTPS ws.com	
ap-northeast-2			
아시아 태평양(오사카-로 컬) 리전 * ap-northeast-3	elasticache.ap- northeast-3.amazona	HTTPS ws.com	
아시아 태평양(싱가포르) 리전	elasticache.ap- southeast-1.amazona	HTTPS ws.com	
ap-southeast-1			

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 지원되는 리전 및 엔드포인트

-1-1 -1=			
리전 이름/리전	엔드포인트	프로토콜	
아시아 태평양(시드니) 리전	elasticache.ap- southeast-2.amazona	HTTPS ws.com	
ap-southeast-2			
유럽(프랑크푸르트) 리전	elasticache.eu- central-1.amazonaws	HTTPS .com	
eu-central-1			
유럽(아일랜드) 리전	elasticache.eu- west-1.amazonaws.co	HTTPS m	
eu-west-1			
유럽(런던) 리전	elasticache.eu- west-2.amazonaws.co	HTTPS m	
eu-west-2			
eu-west-3	elasticache.eu- west-3.amazonaws.co	HTTPS m	
남아메리카(상파울루) 리 전	elasticache.sa- east-1.amazonaws.co	HTTPS m	
sa-east-1			
중국(북경) 지역	elasticache.cn- north-1.amazonaws.c	HTTPS om.cn	
cn-north-1			
중국(닝샤) 리전	elasticache.cn- northwest-1.amazona	HTTPS ws.com.cn	
cn-northwest-1			
아시아 태평양(홍콩) 리 전	elasticache.ap- east-1.amazonaws.co	HTTPS m	
ap-east-1			
AWS GovCloud(미국 서 부)	elasticache.us- gov- west-1.amazonaws.co	HTTPS m	
us-gov-west-1			
	vCloud(US) 사용 방법에 대 네스: ElastiCache를 참조하		
참고:			
데 사용할 수 있는 로컬 리	걸) 리전은 액세스를 요청하 전입니다. 아시아 태평양(오 자에게 문의하십시오. 아시여 원합니다.	사카-로컬) 리전을 사용하	

일부 리전은 노드 유형의 하위 집합을 지원합니다. AWS 리전별로 지원되는 노드 유형을 정리한 표는 AWS 리전별로 지원되는 노드 유형 (p. 56) 단원을 참조하십시오.

리전별 AWS 제품 및 서비스 표는 리전별 제품 및 서비스에서 참조하십시오.

노드 찾기

Amazon ElastiCache는 AZ(가용 영역) 하나 또는 여러 개에서 클러스터의 모든 노드를 찾도록 지원합니다. 뿐만 아니라 여러 AZ에서 클러스터의 노드를 찾도록 선택하면(권장 사항) ElastiCache를 통해 각 노드의 AZ 를 선택하거나 ElastiCache에서 자동으로 선택하도록 할 수 있습니다.

여러 AZ에서 노드를 찾으면 AZ 하나에서 정전과 같은 장애가 발생할 경우 전체 시스템이 실패하는 경우가 없어집니다. AZ 하나에서 모든 노드를 찾거나 여러 AZ에 노드를 분산시키는 사이에 지연 시간 차이가 크게 발생하지 않는다는 사실이 테스트를 통해 드러났습니다.

기존 클러스터를 수정할 때 노드를 추가거나 클러스터를 생성할 때 각 노드에 대한 AZ를 지정할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.

- 클러스터 생성 (p. 65)
- ElastiCache 클러스터 수정 (p. 87)
- 클러스터에 노드 추가 (p. 91)

유지 관리 관리 중

모든 클러스터 및 복제 그룹에는 시스템 변경 내용이 적용되는 주 단위 유지 관리 기간이 있습니다. 클러스터 나 복제 그룹을 생성 또는 수정할 때 원하는 유지 관리 기간을 지정하지 않으면 ElastiCache가 임의로 선택한 요일에 리전의 유지 관리 기간 내에서 60분의 유지 관리 기간을 지정합니다.

리전별로 8시간 블록 시간 중에서 60분 유지 관리 시간이 임의로 선택됩니다. 다음 표는 기본 유지 관리 기간이 할당된 각 지역별 시간 블록 목록입니다. 리전의 유지 관리 기간 블록 외부에서 원하는 유지 관리 기간을 선택할 수 있습니다.

지역 코드	리전 이름	리전 유지 관리 기간
ap-northeast-1	아시아 태평양(도쿄) 리전	13:00-21:00 UTC
ap-northeast-2	아시아 태평양(서울) 리전	12:00–20:00 UTC
ap-northeast-3	아시아 태평양(오사카-로컬) 리전	12:00–20:00 UTC
ap-south-1	아시아 태평양(뭄바이) 리전	17:30–1:30 UTC
ap-southeast-1	아시아 태평양(싱가포르) 리전	14:00–22:00 UTC
ap-southeast-2	아시아 태평양(시드니) 리전	12:00-20:00 UTC
cn-north-1	중국(베이징) 리전	14:00-22:00 UTC
eu-central-1	유럽(프랑크푸르트) 리전	23:00-07:00 UTC
eu-west-1	유럽(아일랜드) 리전	22:00-06:00 UTC
eu-west-2	유럽(런던) 리전	23:00-07:00 UTC
sa-east-1	남아메리카(상파울루) 리전	01:00-09:00 UTC
us-east-1	미국 동부(버지니아 북부) 지역	03:00-11:00 UTC
us-east-2	미국 동부(오하이오) 리전	04:00-12:00 UTC
us-gov-west-1	AWS GovCloud (US) 리전	06:00-14:00 UTC
us-west-1	미국 서부(캘리포니아 북부) 리전	06:00-14:00 UTC

지역 코드	리전 이름	리전 유지 관리 기간	
us-west-2	미국 서부(오레곤) 지역	06:00-14:00 UTC	

클러스터 또는 복제 그룹의 유지 관리 기간 변경

유지 관리 기간은 사용률이 가장 낮은 시간에 할당되어야 하므로 수시로 수정되어야 할 수 있습니다. 클러스터나 복제 그룹을 수정하여 요청한 유지 관리 활동이 이루어지는 기간을 최대 24시간까지 지정할 수 있습니다. 이 시간 동안 사용자가 요청한 지연된 또는 대기 중인 클러스터 수정이 발생합니다.

추가 정보

유지 관리 기간 및 노드 대체에 대한 자세한 내용은 다음을 참조하십시오.

- 유지 관리 및 노드 교체에 대한 ElastiCache 유지 관리—FAQ
- 노드 교체 (p. 60)—노드 교체 관리
- ElastiCache 클러스터 수정 (p. 87)—클러스터의 유지 관리 기간 변경
- 복제 그룹 수정 (p. 165)—복제 그룹의 유지 관리 기간 변경

노드 관리

노드는 Amazon ElastiCache 배포의 가장 작은 구성 요소입니다. 안전한 네트워크 부착 RAM의 크기가 고 정된 청크입니다. 각 노드는 클러스터나 복제 그룹이 생성되거나 마지막으로 수정되었을 때 선택한 엔진을 실행합니다. 각 노드에는 고유한 DNS(Domain Name Service) 이름 및 포트가 있습니다. 여러 유형의 ElastiCache 노드가 지원되며, 연결된 메모리 양과 컴퓨팅 파워는 각각 다릅니다.

일반적으로 샤딩 지원으로 인해 Redis(클러스터 모드 활성화됨) 배포에는 더 작은 노드가 다수 있습니다. 반대로 Redis(클러스터 모드 비활성화됨) 배포에는 클러스터에 더 적은 수의 더 큰 노드가 있습니다. 사용할 노드 크기에 대한 자세한 내용은 노드 크기 선택 (p. 67) 단원을 참조하십시오.

주제

- Redis 노드 및 샤드 (p. 51)
- 노드에 연결 (p. 52)
- ElastiCache 예약 노드 (p. 54)
- 지원되는 노드 유형 (p. 54)
- 노드 교체 (p. 60)

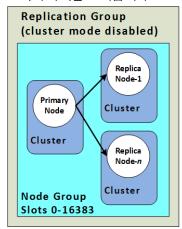
노드와 관련된 몇 가지 중요한 작업은 다음과 같습니다.

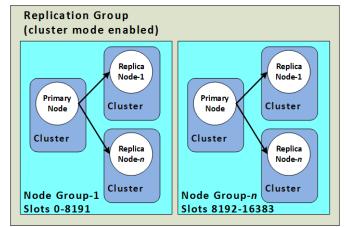
- 클러스터에 노드 추가 (p. 91)
- 클러스터에서 노드 제거 (p. 96)
- Redis용 ElastiCache 클러스터 조정 (p. 219)
- 연결 엔드포인트 찾기 (p. 180)

Redis 노드 및 샤드

샤드(API 및 CLI에서 노드 그룹)는 노드의 계층적 정렬이며 각각 클러스터에 래핑되어 있습니다. 샤드는 복제를 지원합니다. 샤드에서 노드 하나가 읽기/쓰기 기본 노드로 사용됩니다. 샤드에 있는 다른 모든 노드는 기본 노드의 읽기 전용 복제본 역할을 합니다. Redis 버전 3.2 이상에서는 클러스터 내에서 여러 샤드를 지원합니다(API 및 CLI에서 복제 그룹). 이 지원으로 인해 Redis(클러스터 모드 활성화됨) 클러스터에서 데이터를 파티셔닝할 수 있습니다.

다음 다이어그램에서는 Redis(클러스터 모드 비활성화됨) 클러스터와 Redis(클러스터 모드 활성화됨) 클러스터의 차이를 보여줍니다.





Redis(클러스터 모드 비활성화됨)와 Redis(클러스터 모드 활성화됨) 모두 샤드를 통한 복제를 지원합니다. API 작업 DescribeReplicationGroups(CLI: describe-replication-groups)는 기타 정보뿐 아니라 멤버 노드를 포함한 노드 그룹, 노드 그룹에서의 노드 역할을 나열합니다.

Redis 클러스터를 생성할 때 클러스터링을 활성화하여 클러스터를 생성할지 여부를 지정합니다. Redis(클러스터 모드 비활성화됨) 클러스터는 읽기 복제본 노드를 추가하거나(총 5개까지) 삭제하여 수평으로 조정할수 있는 1개 이내의 샤드를 포함합니다. 자세한 내용은 고가용성을 위한 복제 그룹 사용 (p. 127), Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본 추가 (p. 176) 또는 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본 삭제 (p. 178) 단원을 참조하십시오. Redis(클러스터 모드 비활성화됨) 클러스터 역시 노드 유형을 변경하여 수직으로 조정할 수 있습니다. 자세한 내용은 복제본 노드가있는 Redis(클러스터 모드 비활성화됨) 클러스터 조정 (p. 230) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨) 클러스터를 생성할 때 1 ~ 90개까지 샤드를 지정합니다.

Note

노드 또는 샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 한도 증가를 요청하는 방법에 대한 지침은 AWS 서비스 제한을 참조하고 한도 유형을 인스턴스 유형별 클러스터당 노드로 선택하십시오

Redis(클러스터 모드 활성화됨) 클러스터가 생성된 후에는 변경(스케일 인 또는 스케일 아웃)이 가능합니다. 자세한 내용은 Redis용 ElastiCache 클러스터 조정 (p. 219) 및 노드 교체 (p. 60) 단원을 참조하십시오.

새 클러스터를 생성할 때 빈 상태로 시작되지 않도록 이전 클러스터의 데이터를 시드할 수 있습니다. 이는 클러스터 그룹에 이전 클러스터와 동일한 수의 샤드가 있는 경우에만 사용할 수 있는 방식입니다. 이렇게 하면 노드 유형이나 엔진 버전을 변경해야 하는 경우 도움이 됩니다. 자세한 내용은 수동 백업 만들기 (p. 190) 및 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.

노드에 연결

Redis 클러스터의 노드에 연결하기 전에 노드의 엔드포인트가 있어야 합니다. 엔드포인트를 찾으려면 다음을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 181)
- Redis(클러스터 모드 활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 182)
- 엔드포인트 찾기 (AWS CLI) (p. 184)
- 엔드포인트 찾기 (ElastiCache API) (p. 186)

다음 예제에서는 redis-cli 유틸리티를 사용하여 Redis를 실행하는 클러스터에 연결합니다.

Note

Redis 및 사용 가능한 Redis 명령에 대한 자세한 내용은 http://redis.io/commands 웹페이지를 참조하십시오.

redis-cli를 사용하여 Redis 클러스터에 연결하려면

1. 선택한 연결 유틸리티를 사용하여 Amazon EC2 인스턴스에 연결합니다.

Note

Amazon EC2 인스턴스에 연결하는 방법에 대한 지침은 Amazon EC2 Getting Started Guide 단원을 참조하십시오.

2. redis-cli를 구축하려면, GNU 컴파일러 모음(gcc)을 다운로드하여 설치합니다. EC2 인스턴스의 명령 프롬프트에서 다음 명령을 입력하고 확인 프롬프트에서 y를 입력합니다.

```
sudo yum install gcc
```

다음과 같이 유사한 출력이 나타납니다.

```
Loaded plugins: priorities, security, update-motd, upgrade-helper
Setting up Install Process
Resolving Dependencies
--> Running transaction check
...(output omitted)...
Total download size: 27 M
Installed size: 53 M
Is this ok [y/N]: y
Downloading Packages:
(1/11): binutils-2.22.52.0.1-10.36.amzn1.x86_64.rpm
                                                         | 5.2 MB
                                                                       00:00
                                                         | 4.8 MB
(2/11): cpp46-4.6.3-2.67.amzn1.x86_64.rpm
                                                                       00:00
(3/11): gcc-4.6.3-3.10.amzn1.noarch.rpm
                                                          | 2.8 kB
                                                                       00:00
...(output omitted)...
Complete!
```

3. redis-cli 유틸리티를 다운로드하고 컴파일합니다. 이 유틸리티는 Redis 소프트웨어 배포에 포함되어 있습니다. EC2 인스터스의 명령 프롬프트에 다음 명령을 입력합니다.

Note

Ubuntu 시스템의 경우 make를 실행하기 전에 make distclean을 실행합니다.

```
wget http://download.redis.io/redis-stable.tar.gz
tar xvzf redis-stable.tar.gz
cd redis-stable
make distclean // ubuntu systems only
make
```

4. EC2 인스턴스의 명령 프롬프트에서 다음 명령을 입력하여 클러스터의 엔드포인트를 이 예제에 표시된 것으로 대체합니다.

연결할 클러스터의 노드마다 이 단계를 반복합니다.

```
src/redis-cli -c -h mycachecluster.eaogs8.0001.usw2.cache.amazonaws.com -p 6379
```

다음과 유사한 Redis 명령 프롬프트가 나타납니다.

```
redis mycachecluster.eaogs8.0001.usw2.cache.amazonaws.com 6379>
```

Redis 명령을 실행하여 연결을 테스트합니다.

이제 클러스터에 연결되어 Redis 명령을 실행할 수 있습니다. 다음은 Redis 응답이 있는 몇 가지 예제 명령입니다.

```
set a "hello"
                       // Set key "a" with a string value and no expiration
OK
get a
                       // Get value for key "a"
"hello"
get b
                       // Get value for key "b" results in miss
(nil)
set b "Good-bye" EX 5 // Set key "b" with a string value and a 5 second expiration
get b
"Good-bye"
                   // wait 5 seconds
get b
                       // key has expired, nothing returned
(nil)
quit
                       // Exit from redis-cli
```

Secure Sockets Layer(SSL) 암호화(전송 중 데이터 암호화 활성화)가 지원되는 노드 또는 클러스터를 연결하는 방법은 Redis용 ElastiCache 전송 중 데이터 암호화(TLS) (p. 300) 단원을 참조하십시오.

ElastiCache 예약 노드

하나 이상의 노드를 예약하여 비용을 줄일 수 있습니다. 노드 유형과 예약 기간—(1년 또는 3년)에 따라 예약 노드에 선결제 요금이 부과됩니다. 선결제 요금 외에도 시간당 사용 요금이 부과됩니다. 이 요금은 온디맨드 노드에서 발생하는 시간당 사용 요금보다 훨씬 낮습니다.

예약된 노드가 사용 사례에 대해 비용이 절감되는지 확인하려면 먼저 필요한 노드 수와 노드 크기를 결정합니다. 그런 다음 노드의 사용량을 예측하고 온디맨드 노드와 예약된 노드의 총 비용을 비교합니다. 클러스터에서 예약 노드와 온디맨드 노드를 함께 사용할 수 있습니다. 요금 정보는 Amazon ElastiCache 요금을 참조하십시오.

자세한 내용은 예약 노드를 통해 비용 관리 (p. 382) 단원을 참조하십시오.

지원되는 노드 유형

ElastiCache는 다음 노드 유형을 지원합니다. 일반적으로, 현재 세대 유형은 이전 세대의 동급 제품에 비해더 많은 메모리와 컴퓨팅 파워를 더 저렴하게 제공합니다.

- 범용:
 - 현재 세대:

M5 노드 유형: cache.m5.large, cache.m5.xlarge, cache.m5.2xlarge, cache.m5.4xlarge, cache.m5.12xlarge, cache.m5.24xlarge

M4 노드 유형: cache.m4.large, cache.m4.xlarge, cache.m4.2xlarge, cache.m4.4xlarge, cache.m4.10xlarge

T3 노드 유형: cache.t3.micro, cache.t3.small, cache.t3.medium

T2 노드 유형: cache.t2.micro, cache.t2.small, cache.t2.medium

• 이전 세대: (권장하지 않음)

T1 노드 유형: cache.t1.micro

M1 노드 유형: cache.m1.small, cache.m1.medium, cache.m1.large, cache.m1.xlarge

M3 노드 유형: cache.m3.medium, cache.m3.large, cache.m3.xlarge, cache.m3.2xlarge

- 컴퓨팅 최적화:
 - 이전 세대: (권장하지 않음)

C1 노드 유형: cache.c1.xlarge

- 메모리 최적화:
 - 현재 세대:

R5 노드 유형: cache.r5.large, cache.r5.xlarge, cache.r5.2xlarge, cache.r5.4xlarge, cache.r5.12xlarge, cache.r5.24xlarge

R4 노드 유형: cache.r4.large, cache.r4.xlarge, cache.r4.2xlarge, cache.r4.4xlarge, cache.r4.8xlarge, cache.r4.16xlarge

• 이전 세대: (권장하지 않음)

M2 노드 유형: cache.m2.xlarge, cache.m2.2xlarge, cache.m2.4xlarge

R3 노드 유형: cache.r3.large, cache.r3.xlarge, cache.r3.2xlarge, cache.r3.4xlarge, cache.r3.8xlarge

Amazon ElastiCache에서 범용 순간 확장 가능 T3 표준 캐시 노드를 실행할 수 있습니다. 이러한 노드는 기준 수준의 CPU 성능과 더불어 누적된 크레딧이 소진될 때까지 언제든 CPU 사용량을 순간 확장할 수 있는 기능을 제공합니다. CPU 크레딧은 1분 동안 CPU 코어의 전체 성능을 제공합니다.

Amazon ElastiCache의 T3 노드는 표준으로 구성되고 평균 CPU 사용률이 인스턴스의 기준 성능보다 일관되게 낮은 워크로드에 적합합니다. 기준 이상으로 순간 확장하려면 노드는 CPU 크레딧 밸런스에 누적한 크레딧을 사용합니다. 누적된 크레딧에서 노드가 부족한 경우, 성능이 점진적으로 기준 성능 수준으로 저하됩니다. 이렇게 점진적으로 저하되면 누적된 CPU 크레딧 밸런스가 고갈될 때 노드에 급격한 성능 저하가 발생하지 않습니다. 자세한 내용은 Amazon EC2 사용 설명서에서 성능 순간 확장 가능 인스턴스에 대한 CPU 크레딧및 기준 성능을 참조하십시오.

다음 표에는 성능 순간 확장 가능 노드 유형과 시간당 CPU 크레딧이 획득되는 속도가 나와 있습니다. 또한 노드가 누적할 수 있는 최대 획득 CPU 크레딧 개수와 노드당 vCPU 개수도 보여줍니다. 또한 기준 성능 수준 을 전체 코어 성능의 백분율로 제공합니다(단일 vCPU 사용).

시간당 획득된 CPU 크레딧	누적 가능 한 최대 획 득된 크레딧*	vCPUs	vCPU당 기준 성능	메모리(GiB)	네트워 크 성능
t32micro	288	2	10%	0.5	최대 5 기가비트
t3-small	576	2	20%	1.37	최대 5 기가비트
t34medium	576	2	20%	3.09	최대 5 기가비트

^{*} 누적될 수 있는 크레딧은 수는 24시간 동안 획득할 수 있는 크레딧의 수와 동일합니다.

** 테이블의 기준 성능은 vCPU 단위로 계산됩니다. vCPU가 1개 이상인 일부 노드 크기입니다. vCPU 백분율에 vCPU 개수를 곱하여 노드의 기준 CPU 사용률을 계산합니다.

다음 CPU 크레딧 지표는 성능 순간 확장 가능 인스턴스에 사용할 수 잇습니다.

- CPUCreditUsage
- CPUCreditBalance

이 지표에 대한 자세한 내용은 CPU 크레딧 지표 단원을 참조하십시오.

또한 다음 사항을 숙지해야 합니다.

- 모든 현재 세대 노드 유형은 기본적으로 Amazon VPC를 기반으로 Virtual Private Cloud(VPC)에서 생성됩니다.
- Redis AOF(append-only files)는 T1 또는 T2 인스턴스에서 지원되지 않습니다.
- 자동 장애 조치가 포함된 Redis 다중 AZ는 T1 인스턴스에서 지원되지 않습니다.
- Redis 구성 변수 appendonly 및 appendfsync는 Redis 버전 2.8.22 이상에서 지원되지 않습니다.

Note

지원되는 엔진 버전은 AWS 리전별로 다릅니다. 최신 엔진 버전은 모든 AWS 리전에서 지원됩니다. AWS 리전에 사용 가능한 엔진 버전을 확인하려면 지원되는 Redis용 ElastiCache 버전 (p. 37) 단원을 참조하십시오.

AWS 리전별로 지원되는 노드 유형

다음 표에는 각 AWS 리전에 지원되는 노드 유형이 나와 있습니다.

AWS 리전	ТЗ	T2	M4	M5	R4	R5
교s-east-2 국 동 부 (오 하 이 오)	예	예	예	예	예	예
교s-east-1 국 동 부 (버 지 니 아 북 부)	예	예	예	예	예	예
Qls-west-1 국 서 부	예	예	예	예	예	예

AWS 리전	Т3	T2	M4	M5	R4	R5
(캘 리 포 니 아 북 부 지 역)						
데s-west-2 국 서 부 (오 레 곤)	예	예	예	예	예	예
캐a-central-1 나 다 (중 부)	예	예	예	예	예	예
영p-south-1 시 아 태 평 양 (뭄 바 이)	예	예	예	예	예	예
에p-northeast-1 시 아 태 평 양 (도 교)	예	예	예	예	예	예
에p-northeast-2 시 아 태 평 양 (서 울)	예	예	예	예	예	예

AWS 리전	Т3	T2	M4	M5	R4	R5
@p-northeast-3 시 아 태 평 양 (오 사 카- 로 컬)*	예	예	예	예	예	
@p-southeast-1 시 아 태 평 양 (싱 가 포 르)	예	예	예	예	예	예
emp-southeast-2 시 아 태 평 양 (시 드 니)	예	예	예	예	예	예
ehp-east-1 시 아 태 평 양 (홍 콩)	예			예		예
용u-north-1 럽 (스 톡 흘 름)	예			예	아니요	예

AWS 리전	Т3	T2	M4	M5	R4	R5
Ru-central-1 럽 (프 랑 크 푸 르 트)	예	예	예	예	예	예
유u-west-1 럽 (아 일 랜 드)	예	예	예	예	예	예
유u-west-2 럽 (런 던)	예	예	예	예	예	예
듄네(짜 est-3 리)	예	예		예	예	예
남a-east-1 아 메 리 카 (상 파 울 루)	예	예	예	예	예	예
중n-north-1 국 (베 이 징)	예	예	예		예	예
중n-northwest-1 국 (닝 샤)	예	예	예		예	예
중e-south-1 동 (바 레 인)	예			예		예

노드 교체

AWS 리전	Т3	T2	M4	M5	R4	R5
ANA/Sgov-west-1 GovCloud (US- West)	예	예		예	예	예

* 아시아 태평양(오사카-로컬) 리전은 액세스를 요청하는 AWS 고객을 선택하는 데 사용할 수 있는 로컬 리전입니다. 아시아 태평양(오사카-로컬) 리전을 사용하려는 경우 해당 영업 담당 자에게 문의하십시오. 아시아 태평양(오사카-로컬) 리전은 단일 가용 영역을 지원합니다.

노드 유형 및 사양의 전체 목록은 다음을 참조하십시오.

- Amazon ElastiCache 제품 기능 및 세부 정보
- Redis 특정 파라미터 (p. 274)

노드 교체

Redis용 Amazon ElastiCache는 인스턴스에 원활하게 적용되는 패치 및 업그레이드를 통해 플릿을 주기적으로 업그레이드합니다. 하지만 기본 호스트에 필수 OS 업데이트를 적용하기 위해 Redis용 ElastiCache 노드를 다시 시작해야 하는 경우가 있습니다. 보안, 안정성 및 운영 성능을 강화하는 업그레이드 적용에 있어 이러한 교체가 필요합니다.

예정된 노드 교체 주기 이전에 언제든 이러한 교체를 직접 관리할 수 있는 옵션이 있습니다. 직접 대체를 관리할 때 노드를 다시 시작하면 인스턴스에서 OS 업데이트를 수신하고, 예정된 노드 대체는 취소됩니다. 노드 대체가 발생한다는 경고를 계속 수신할 수 있습니다. 이미 유지 관리의 필요성을 수동으로 완화한 경우 이 경고를 무시할 수 있습니다.

Note

Amazon ElastiCache에서 자동으로 생성된 교체 캐시 노드는 IP 주소가 다를 수 있습니다. 애플리케이션 구성을 검토하여 캐시 노드가 적절한 IP 주소와 연결되어 있는지 확인해야 합니다.

다음 목록은 ElastiCache에서 Redis 노드 하나의 대체를 예약할 경우 취할 수 있는 조치를 보여줍니다. 상황에 맞는 정보를 신속하게 찾으려면 다음 메뉴에서 선택하십시오.

- 아무 작업 안 함 (p. 61) Amazon ElastiCache에서 예정대로 노드를 대체합니다.
- 유지 관리 기간 변경 (p. 61) 더 적합한 시간으로 유지 관리 기간을 변경합니다.
- Redis(클러스터 모드 활성화됨) 구성
 - 모든 Redis 클러스터에서 노드만 대체 (p. 62) 백업 및 복원을 사용하여 Redis 클러스터의 노드를 대체하는 절차입니다.
 - Redis 클러스터에서 복제본 노드 대체 (p. 62) 클러스터 가동 중단 없이 복제본 수를 늘이거나 줄여 모든 Redis 클러스터에서 읽기 전용 복제본을 대체하는 절차입니다.
 - Redis(클러스터 모드 활성화됨)샤드에서 노드 대체 (p. 62) 클러스터 가동 중단 없이 확장 및 축소하여 Redis(클러스터 모드 활성화됨) 클러스터에서 노드를 대체하는 동적 절차입니다.
- Redis(클러스터 모드 비활성화됨) 구성
 - 모든 Redis 클러스터에서 노드만 대체 (p. 62) 백업 및 복원을 사용하여 Redis 클러스터의 모든 노드를 대체하는 절차입니다.
 - Redis 클러스터에서 복제본 노드 대체 (p. 62) 클러스터 가동 중단 없이 복제본 수를 늘이거나 줄여 모든 Redis 클러스터에서 읽기 전용 복제본을 대체하는 절차입니다.
 - Redis(클러스터 모드 비활성화됨)클러스터에서 노드 대체 (p. 62) 복제를 사용하여 Redis(클러스터 모드 비활성화됨) 클러스터의 노드를 대체하는 절차입니다.

- Redis(클러스터 모드 비활성화됨)읽기 전용 복제본 대체 (p. 63) Redis(클러스터 모드 비활성화됨) 복제 그룹의 읽기 전용 복제본을 수동으로 대체하는 절차입니다.
- Redis(클러스터 모드 비활성화됨)기본 노드 대체 (p. 63) Redis(클러스터 모드 비활성화됨) 복제 그 룹의 기본 노드를 수동으로 대체하는 절차입니다.

Redis 노드 대체 옵션

• 아무 작업 안 함 – 아무 작업도 하지 않으면 ElastiCache예약대로 노드를 대체합니다.

노드가 자동 장애 조치가 활성화된 노드의 구성원일 경우 Redis용 ElastiCache은 패치 중 개선된 가용성과 업데이트, 기타 유지 관리 관련 노드 교체를 제공합니다.

ElastiCache for Redis 클러스터 클라이언트를 사용하기 위해 설정되는 ElastiCache for Redis 클러스터 구성의 경우, 클러스터에서 수신 쓰기 요청을 처리하는 동안 교체가 완료됩니다.

자동 장애 조치가 활성화된 비클러스터 구성의 경우, 클러스터가 온라인 상태로 들어오는 쓰기 요청을 처리하는 동안 Redis 5.0.5의 클러스터가 교체를 완료합니다. Redis 5.0.4 이하에서 자동 장애 조치가 활성화된 클러스터의 경우, DNS 업데이트와 관련하여 최대 수 초 간의 짧은 쓰기 중단이 발생할 수 있습니다.

노드가 독립 실행형이면 먼저 Amazon ElastiCache에서 대체 노드를 시작한 후 기존 노드에서 동기화합니다. 그 동안은 서비스 요청에 기존 노드를 사용할 수 없습니다. 동기화가 완료되면 기존 노드가 종료되고 새 노드가 그 역할을 대신합니다. ElastiCache는 이 작업이 진행되는 동안 데이터를 보존하기 위해 노력합니다.

- 유지 관리 기간 변경 예약된 유지 관리 이벤트의 경우 ElastiCache에서 이메일 또는 알림 이벤트를 수신합니다. 이러한 경우 예약된 대체 시간 전에 유지 관리 기간을 변경하면 이제 노드가 새 시간에 대체됩니다. 자세한 내용은 다음 자료를 참조하십시오.
 - ElastiCache 클러스터 수정 (p. 87)
 - 복제 그룹 수정 (p. 165)

Note

유지 관리 기간을 이동해 교체 기간을 변경하는 기능은 ElastiCache 알림에 유지 관리 기간이 포함된 경우에만 사용할 수 있습니다. 알림에 유지 관리 기간이 포함되어 있지 않으면 교체 기간을 변경할 수 없습니다.

예를 들어 11월 9일 목요일 15:00, 다음 유지 관리 기간은 11월 10일 금요일 17:00라고 가정해 보겠습니다. 다음을 이러한 가정의 결과를 보여주는 3가지 시나리오입니다.

- 유지 관리 기간을 현재 날짜/시간 이후 및 예약된 다음 유지 관리 기간 이전인 금요일 16:00으로 변경합니다. 11월 10일 금요일 16:00에 노드가 대체됩니다.
- 유지 관리 기간을 현재 날짜/시간 이후 및 예약된 다음 유지 관리 기간 이전인 토요일 16:00으로 변경합니다. 11월 11일 토요일 16:00에 노드가 대체됩니다.
- 유지 관리 기간을 현재 날짜/시간보다 일주일 빠른 수요일 오후 4시로 변경합니다. 11월 15일 수요일 16:00에 노드가 대체됩니다.

지침은 유지 관리 관리 중 (p. 50) 단원을 참조하십시오.

노드 교체

• 모든 Redis 클러스터에서 노드만 대체 – 클러스터에 읽기 전용 복제본이 없는 경우 다음 절차를 사용하여 노드를 대체할 수 있습니다.

백업 및 복원을 사용하여 노드만을 대체하려면 다음을 수행합니다.

- 1. 노드 클러스터의 스냅샷을 생성합니다. 지침은 수동 백업 만들기 (p. 190) 단원을 참조하십시오.
- 2. 스냅샷에서 시드하여 새 클러스터를 생성합니다. 지침은 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.
- 3. 대체 예약한 노드가 포함된 클러스터를 삭제합니다. 지침은 클러스터 삭제하기 (p. 102) 단원을 참조하십시오.
- 4. 애플리케이션에서 이전 노드의 엔드포인트를 새 노드의 엔드포인트로 대체합니다.
- Redis 클러스터에서 복제본 노드 대체- 복제본 클러스터를 대체하려면, 복제본 개수를 늘립니다. 이렇게 하려면 복제본을 추가한 다음 대체할 복제본을 제거하여 복제본 수를 줄입니다. 이 프로세스는 동적이며 클러스터 중단 시간이 없습니다.

Note

샤드 또는 복제 그룹에 이미 five 복제본이 있는 경우 1단계와 2단계를 반대로 합니다.

모든 Redis 클러스터에서 복제본을 대체하려면 다음을 수행합니다.

- 1. 샤드 또는 복제 그룹에 복제본을 추가하여 복제본 수를 늘립니다. 자세한 내용은 샤드의 복제본 수 늘리기 (p. 168) 단원을 참조하십시오.
- 2. 대체하려는 복제본을 삭제합니다. 자세한 내용은 샤드의 복제본 수 줄이기 (p. 172) 단원을 참조하십시오.
- 3. 애플리케이션에서 엔드포인트를 업데이트합니다.
- Redis(클러스터 모드 활성화됨) 샤드에서 노드 대체 중단 시간 없이 클러스터에서 노드를 대체하려면 온 라인 리샤딩을 사용하십시오. 먼저 확장하여 샤드를 추가한 다음 축소하여 대체할 노드로 샤드를 삭제합니다.

Redis(클러스터 모드 활성화됨) 클러스터에서 모든 노드를 대체하려면 다음을 수행합니다.

- 1. 확장: 대체할 노드가 포함된 기존 샤드와 동일한 구성의 추가 샤드를 추가합니다. 자세한 내용은 온라인 리샤딩을 사용하여 샤드 추가 (p. 244) 단원을 참조하십시오.
- 2. 축소: 대체할 노드가 포함된 샤드를 삭제합니다. 자세한 내용은 온라인 리샤딩을 사용하여 샤드 제거 (p. 247) 단원을 참조하십시오.
- 3. 애플리케이션에서 엔드포인트를 업데이트합니다.
- Redis(클러스터 모드 비활성화됨) 클러스터에서 노드 대체 클러스터가 읽기 전용 복제본이 없는 Redis(클러스터 모드 비활성화됨) 클러스터인 경우 다음 절차를 사용하여 노드를 대체할 수 있습니다.

복제를 사용하여 노드 대체하려면(클러스터 모드 비활성화 전용)

- 1. 기본으로 대체하도록 예약한 노드가 있는 클러스터에 복제를 추가합니다. 이 클러스터에서 다중 AZ를 활성화하지 마십시오. 지침은 샤드 없이 Redis 클러스터에 복제를 추가하려면 (p. 92) 단원을 참조하십시오.
- 2. 클러스터에 읽기 전용 복제본을 추가합니다. 지침은 클러스터 (console)에 노드를 추가하려면 (p. 92) 단원을 참조하십시오API 버전 2015-02-02

- 3. 읽기 전용 복제본을 기본으로 승격합니다. 지침은 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본을 기본으로 승격 (p. 178) 단원을 참조하십시오.
- 4. 대체 예약한 노드를 삭제합니다. 지침은 클러스터에서 노드 제거 (p. 96) 단원을 참조하십시오.
- 5. 애플리케이션에서 이전 노드의 엔드포인트를 새 노드의 엔드포인트로 대체합니다.
- Redis(클러스터 모드 비활성화됨) 읽기 전용 복제본 대체 노드가 읽기 전용 복제본이면 노드를 대체하십시오.

클러스터에 복제본 노드가 한 개뿐이고 다중 AZ가 활성화되어 있으면 다중 AZ를 비활성화해야 복제본을 삭제할 수 있습니다. 지침은 복제 그룹 수정 (p. 165) 단원을 참조하십시오.

Redis(클러스터 모드 비활성화됨) 읽기 전용 복제본을 대체하려면

- 1. 대체 예약된 복제본을 삭제합니다. 지침은 다음을 참조하십시오.
 - 샤드의 복제본 수 줄이기 (p. 172)
 - 클러스터에서 노드 제거 (p. 96)
- 2. 대체 예약된 복제본을 대체할 새 복제본을 추가합니다. 삭제한 복제본과 같은 이름을 사용하는 경우 3 단계를 건너뛸 수 있습니다. 지침은 다음을 참조하십시오.
 - 샤드의 복제본 수 늘리기 (p. 168)
 - Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본 추가 (p. 176)
- 3. 애플리케이션에서 이전 복제본의 엔드포인트를 새 복제본의 엔드포인트로 대체합니다.
- 4. 시작할 때 다중 AZ를 비활성화한 경우 다시 활성화합니다. 지침은 자동 장애 조치가 있는 다중 AZ 활성화 (p. 136) 단원을 참조하십시오.
- Redis(클러스터 모드 비활성화됨) 기본 노드 대체 노드가 기본 노드이면 읽기 전용 복제본을 기본으로 승격하십시오. 그런 다음 기본 노드였던 복제본을 삭제합니다.

클러스터에 복제본이 한 개뿐이고 다중 AZ가 활성화되어 있으면 2단계에서 다중 AZ를 비활성화해야 복제 본을 삭제할 수 있습니다. 지침은 복제 그룹 수정 (p. 165) 단원을 참조하십시오.

Redis(클러스터 모드 비활성화됨) 기본 노드를 대체하려면

- 1. 읽기 전용 복제본을 기본으로 승격합니다. 지침은 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본을 기본으로 승격 (p. 178) 단원을 참조하십시오.
- 2. 대체 예약된 노드(이전의 기본)를 삭제합니다. 지침은 클러스터에서 노드 제거 (p. 96) 단원을 참조하십시오.
- 3. 대체 예약된 복제본을 대체할 새 복제본을 추가합니다. 삭제한 노드와 같은 이름을 사용하는 경우 애 플리케이션에서 엔드포인트 변경을 건너뛸 수 있습니다.

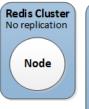
지침은 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본 추가 (p. 176) 단원을 참조하십시오.

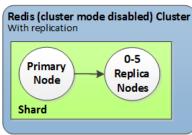
- 4. 애플리케이션에서 이전 노드의 엔드포인트를 새 노드의 엔드포인트로 대체합니다.
- 5. 시작할 때 다중 AZ를 비활성화한 경우 다시 활성화합니다. 지침은 자동 장애 조치가 있는 다중 AZ 활성화 (p. 136) 단원을 참조하십시오.

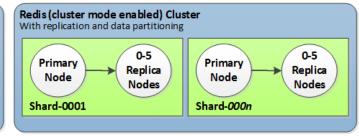
ElastiCache 클러스터 관리

클러스터는 하나 이상의 캐시 노드 모음으로서, 이 모든 노드는 Redis 캐시 엔진 소프트웨어의 인스턴스 하나를 실행합니다. 클러스터를 만들 때 모든 노드에서 사용할 엔진과 버전을 지정합니다.

다음 다이어그램은 일반적인 Redis 클러스터를 나타낸 것입니다. Redis 클러스터는 샤드(API/CLI: 노드 그룹)에 노드가 1개에서 최대 six개까지 포함될 수 있으며, 단일 노드 Redis(클러스터 모드 비활성화됨) 클러스터에는 샤드가 없고 다중 노드 Redis(클러스터 모드 비활성화됨) 클러스터에는 샤드가 1개 있습니다. Redis(클러스터 모드 활성화됨) 클러스터는 샤드에서 데이터가 분할된 최대 90개의 샤드를 포함할 수 있습니다. 샤드에 여러 노드가 있으면 노드 중 하나는 읽기/쓰기 기본 노드가 됩니다. 샤드의 나머지 노드는 모두 읽기 전용 복제본입니다.







일반적인 Redis 클러스터

클러스터 수준에서 대부분의 ElastiCache 작업이 수행됩니다. 특정 수의 노드 및 각 노드에 대한 속성을 제어하는 파라미터 그룹을 사용하여 클러스터를 설정할 수 있습니다. 클러스터 하나에 속한 모든 노드는 노드 유형, 파라미터 및 보안 그룹 설정이 동일합니다.

클러스터마다 클러스터 식별자가 있습니다. 클러스터 식별자는 고객이 제공하는 클러스터 이름입니다. ElastiCache API 및 AWS CLI 명령과 상호 작용할 때 이 식별자가 특정한 클러스터를 지정합니다. 클러스터 식별자는 한 AWS 리전 내의 해당 고객에 대해 고유해야 합니다.

ElastiCache는 여러 가지 엔진 버전을 지원합니다. 특별한 이유가 없으면 항상 엔진의 최신 버전을 사용하는 것이 좋습니다.

ElastiCache 클러스터는 Amazon EC2 인스턴스를 사용하여 액세스하도록 설계되었습니다. Amazon VPC 서비스 기반의 VPC에서 클러스터를 시작하는 경우 AWS 밖에서 액세스할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.

- 2단계: 액세스 자격 부여 (p. 23)
- AWS 외부에서 ElastiCache 리소스 액세스 (p. 107)

지원되는 Redis 버전

- Redis용 ElastiCache 버전 5.0.0(개선됨) (p. 40)
- Redis용 ElastiCache 버전 4.0.10(개선됨) (p. 41)
- Redis용 ElastiCache 버전 3.2.10(개선됨) (p. 41)
- Redis용 ElastiCache 버전 3.2.6(개선됨) (p. 42)
- Redis용 ElastiCache 버전 3.2.4(개선됨) (p. 42)
- Redis용 ElastiCache 버전 2.8.23(개선됨) (p. 43)
- Redis용 ElastiCache 버전 2.8.22(개선됨) (p. 44)
- Redis용 ElastiCache 버전 2.8.19 (p. 44)
- Redis용 ElastiCache 버전 2.8.6 (p. 44)

• Redis용 ElastiCache 버전 2.6.13 (p. 45)

기타 ElastiCache 클러스터 작업

클러스터와 관련된 추가 작업:

- 연결 엔드포인트 찾기 (p. 180)
- AWS 외부에서 ElastiCache 리소스 액세스 (p. 107)

클러스터 생성

이 단원에는 ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하는 클러스터 생성에 관한 지침이 나와 있습니다.

ElastiCache 클러스터는 AWS CloudFormation을 사용하여 생성할 수도 있습니다. 자세한 내용은 해당 접근 방식을 구현하는 방법에 대한 지침이 포함되어 있는 AWS Cloud Formation 사용 설명서에서 AWS::ElastiCache::CacheCluster를 참조하십시오.

클러스터 또는 복제 그룹을 생성할 때마다 준비 작업을 미리 하면 즉시 업그레이드하거나 변경할 필요가 없 어 좋습니다.

주제

- 요구 사항 결정 (p. 65)
- 노드 크기 선택 (p. 67)
- Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69)
- Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 72)
- 클러스터 생성 (AWS CLI) (p. 76)
- 클러스터 생성 (ElastiCache API) (p. 77)

요구 사항 결정

주제

- 메모리 및 프로세서 요구 사항 (p. 66)
- Redis 클러스터 구성 (p. 66)
- 조정 요구 사항 (p. 67)
- 액세스 요구 사항 (p. 67)
- 리전 및 가용 영역 요구 사항 (p. 67)

준비

시작하기 전에 다음 질문에 대한 답을 알아야 클러스터를 생성할 수 있습니다.

• 필요한 노드 인스턴스 유형은 무엇입니까?

인스턴스 노드 유형 선택에 도움이 필요한 경우 노드 크기 선택 (p. 67)를 참조하십시오.

• VPC와 Amazon VPC 중 어디서 클러스터를 시작하시겠습니까?

Important

Amazon VPC에서 클러스터를 시작하는 경우 클러스터를 생성하기 전에 동일한 VPC에서 서브넷 그룹을 생성해야 합니다. 자세한 내용은 서브넷 및 서브넷 그룹 (p. 329) 단원을 참조하십시오.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 클러스터 생성

ElastiCache는 Amazon EC2를 사용하여 AWS 내에서 액세스하도록 고안되었습니다. 하지만 Amazon VPC 기반의 VPC에서 시작하고 클러스터가 VPC에 있는 경우 AWS 밖에서 액세스 권한을 제공할 수 있습니다. 자세한 내용은 AWS 외부에서 ElastiCache 리소스 액세스 (p. 107) 단원을 참조하십시오.

• 파라미터 값을 사용자 지정해야 합니까?

그렇다면 사용자 지정 파라미터 그룹을 생성해야 합니다. 자세한 내용은 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

Redis를 실행하는 경우 적어도 reserved-memory 또는 reserved-memory-percent를 설정해 보십시오. 자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

• 자체 보안 그룹 또는 VPC 보안 그룹을 생성해야 합니까?

자세한 내용은 보안 그룹: EC2-Classic (p. 334) 및 VPC의 보안을 참조하십시오.

• 어떤 방법으로 내결함성을 구현하시겠습니까?

자세한 내용은 장애 완화 (p. 395) 단원을 참조하십시오.

주제

- 메모리 및 프로세서 요구 사항 (p. 66)
- Redis 클러스터 구성 (p. 66)
- 조정 요구 사항 (p. 67)
- 액세스 요구 사항 (p. 67)
- 리전 및 가용 영역 요구 사항 (p. 67)

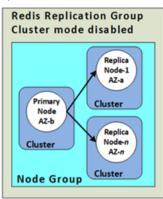
메모리 및 프로세서 요구 사항

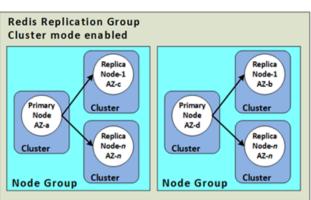
Amazon ElastiCache의 기본 빌딩 블록은 노드입니다. 노드는 개별적으로 또는 그룹으로 구성되어 클러스터 를 형성합니다. 클러스터에 사용할 노드 유형을 결정할 때 클러스터의 노드 구성과 저장해야하는 데이터의 양을 고려합니다.

Redis 클러스터 구성

Redis용 ElastiCache 클러스터는 0개에서 90개의 샤드(노드 그룹이라고도 함)로 구성됩니다. Redis 클러스터에 있는 데이터는 클러스터의 여러 샤드에 두루 분할됩니다. 애플리케이션은 엔드포인트라는 네트워크 주소를 사용하여 Redis 클러스터에 연결됩니다. Redis 샤드에 있는 노드는 읽기/쓰기 기본 노드 하나 및 기타노드 읽기 전용 보조(읽기 전용 복제본이라고도 함) 두 가지 역할 중 하나를 이행합니다. 노드 엔드포인트 외에도 Redis 클러스터는 구성 엔드포인트라는 엔드포인트를 가지고 있습니다. 애플리케이션에서는 이 엔드포인트를 사용하여 클러스터에서 읽거나 쓸 수 있으며, Redis용 ElastiCache에서 읽을 노드 또는 쓸 노드에 대한 결정을 남깁니다.







자세한 내용은 ElastiCache 클러스터 관리 (p. 64) 단원을 참조하십시오.

조정 요구 사항

더 크고 새로운 노드 유형으로 새 클러스터를 생성하여 모든 클러스터를 확장할 수 있습니다. Redis 클러스터를 조정하면 백업에서 볼 수 있고 새 클러스터가 비워지는 것을 방지할 수 있습니다.

자세한 내용은 이 가이드의 Redis용 ElastiCache 클러스터 조정 (p. 219)을 참조하십시오.

액세스 요구 사항

설계에 따라 Amazon ElastiCache 클러스터는 Amazon EC2 인스턴스에서 액세스합니다. ElastiCache 클러스터에 대한 네트워크 액세스는 클러스터를 생성한 사용자 계정에 대해 제한됩니다. 따라서 Amazon EC2 인스턴스에서 클러스터에 액세스하려면 먼저 Amazon EC2 인스턴스가 클러스터에 액세스하도록 승인해야 합니다. 이를 수행하는 단계는 EC2-VPC로 시작했는지, EC2-Classic으로 시작했는지에 따라 다릅니다.

클러스터를 EC2-VPC로 시작한 경우 클러스터에 대한 네트워크 진입을 허용해야 합니다. 클러스터를 EC2-Classic으로 시작한 경우 Amazon Elastic Compute Cloud 보안 그룹에 액세스하여 인스턴스와 연결된 ElastiCache 보안 그룹을 허용해야 합니다. 자세한 지침은 이 가이드의 2단계: 액세스 자격 부여 (p. 23)를 참조하십시오.

리전 및 가용 영역 요구 사항

Amazon ElastiCache은 모든 AWS 리전을 지원합니다. 애플리케이션과 가까운 리전에 ElastiCache 클러스터를 배치하면 지연 시간을 줄일 수 있습니다. 클러스터에 다중 노드가 있는 경우 다른 가용 영역에 노드를 배치하면 클러스터에 장애가 미치는 영향을 줄일 수 있습니다.

자세한 내용은 다음 자료를 참조하십시오.

- 리전 및 가용 영역 선택 (p. 47)
- 장애 완화 (p. 395)

노드 크기 선택

클러스터에 대해 선택하는 노드 크기는 비용, 성능, 내결함성에 영향을 미칩니다.

노드 크기 선택

다음 질문의 답을 생각해 보면 Redis 구현에 필요한 최소한의 노드 유형을 결정하는 데 도움이 될 수 있습니다.

• 데이터에 필요한 총 메모리는 얼마나 됩니까?

일반적인 에상치를 알아보려면 캐시할 항목의 크기를 선택합니다. 이 크기를 동시에 캐시에 보관할 항목 수로 곱합니다. 적당한 항목 크기 예상치를 구하고 캐시 항목을 직렬화한 후 문자 수를 계산합니다. 그런 다음 클러스터에 있는 샤드의 수에 대해 이를 나눕니다.

• 실행 중인 Redis 버전은 무엇입니까?

2.8.22 이전의 Redis 버전에서는 장애 조치, 스냅샷, 동기화, 기본 작업으로 복제본 승격을 위해 더 많은 메모리를 확보해야 합니다. 프로세스 중에 발생하는 모든 쓰기에 충분한 메모리를 사용할 수 있어야 하기 때문입니다.

Redis 버전 2.8.22 이상에서는 이전 프로세스에 비해 사용 가능한 메모리가 더 적게 필요한 forkless 저장 프로세스가 사용됩니다.

자세한 내용은 다음 자료를 참조하십시오.

- 동기화 및 백업 구현 방법 (p. 143)
- 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)
- 애플리케이션의 쓰기 작업이 얼마나 많습니까?

쓰기 작업이 많은 애플리케이션에서는 스냅샷을 만들거나 장애 조치를 할 때 데이터에 사용되지 않으며 사용할 수 있는 메모리가 훨씬 더 많이 필요할 수 있습니다. BGSAVE 프로세스가 수행될 때마다 BGSAVE 프로세스 중에 발생하는 모든 쓰기를 수용할 수 있도록 데이터가 사용하지 않는 메모리가 충분해야 합니다. 예로는 스냅샷을 만들 때, 클러스터의 복제본과 기본 클러스터를 동기화할 때, AOF(append-only file) 기능을 활성화할 때 등이 있습니다. 또 다른 예로는 복제본을 기본으로 승격하는 경우입니다(자동 장애 조치가 활성화된 다중 AZ가 있는 경우). 최악의 경우는 프로세스 중에 모든 데이터가 다시 작성되는 경우입니다. 이 경우 데이터에만 필요한 메모리의 두 배가 되는 노드 인스턴스 크기가 필요합니다.

더 자세한 내용은 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404) 단원을 참조하십시오.

• 구현 애플리케이션이 Redis(클러스터 모드 비활성화됨) 클러스터입니까, 아니면 샤드가 여러 개인 Redis(클러스터 모드 활성화됨) 클러스터입니까?

Redis(클러스터 모드 비활성화됨) 클러스터

Redis(클러스터 모드 비활성화됨) 클러스터를 구현하는 경우 노드 유형은 위에서 설명한 대로 필요한 오버 헤드와 모든 데이터를 수용할 수 있어야 합니다.

예를 들어, 모든 항목의 총 크기를 12GB로 추정합니다. 이 경우, 메모리가 13.3GB인 cache.m3.xlarge 노드 또는 메모리가 13.5GB인 cache.r3.large 노드를 사용할 수 있습니다. 하지만 BGSAVE 작업에는 메모리가 더 필요할 수 있습니다. 애플리케이션이 쓰기 작업이 많은 경우 메모리 요구 사항을 최소 24GB로 두 배로 늘립니다. 따라서 27.9GB의 메모리가 있는 cache.m3.2xlarge 또는 30.5GB의 메모리가 있는 cache.r3.xlarge를 사용합니다.

샤드가 여러 개인 Redis(클러스터 모드 활성화됨)

샤드가 여러 개인 Redis(클러스터 모드 활성화됨) 클러스터를 구현하는 경우 노드 유형은 bytes-for-data-and-overhead / number-of-shards 바이트의 데이터를 수용할 수 있어야 합니다.

예를 들어, 모든 항목의 총 크기를 12GB로 추정하고 샤드가 2개입니다. 이 경우, 메모리가 6.05GB인 cache.m3.large 노드를 사용할 수 있습니다(12GB/2). 하지만 BGSAVE 작업에는 메모리가 더 필요할 수 있습니다. 애플리케이션이 쓰기 작업이 많은 경우 메모리 요구 사항을 최소 샤드당 12GB로 두 배로 늘립니다. 따라서 13.3GB의 메모리가 있는 cache.m3.xlarge 또는 13.5GB의 메모리가 있는 cache.r3.large를 사용합니다.

클러스터가 실행 중이면 CloudWatch에 게시된 메모리 사용량, 프로세서 사용률, 캐시 적중률 및 캐시 누락을 모니터링할 수 있습니다. 클러스터의 적중률이 기대에 미치지 못하거나 키가 너무 자주 제거되고 있음을 알수 있습니다. 이러한 경우 CPU 및 메모리 사양이 더 큰 다른 노드 크기를 선택할 수 있습니다.

CPU 사용량을 모니터링할 때 Redis는 단일 스레드입니다. 따라서 보고된 CPU 사용량을 CPU 코어 수에 곱 하면 실제 사용량을 확인할 수 있습니다. 예를 들어, 사용률 20%로 보고된 4 코어 CPU는 실제로 Redis가 사용하는 코어 1개가 80%로 실행되는 것입니다.

Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔)

ElastiCache는 Redis 엔진을 사용할 때 복제를 지원합니다. Redis 읽기/쓰기 기본 클러스터에 데이터가 쓰여지는 시간과 데이터가 읽기 전용 보조 클러스터에 전파되는 시간 사이의 지연 시간을 모니터링하기 위해 ElastiCache가 특수 키 ElastiCacheMasterReplicationTimestamp를 클러스터에 추가합니다. 이 키는 현재 UCT(협정 세계시)입니다. 나중에 Redis 클러스터가 복제 그룹에 추가될 수 있으므로 처음에는 복제 그룹의 멤버가 아니어도 모든 Redis 클러스터에 이 키가 포함됩니다. 복제 그룹에 대한 자세한 내용은 고가용성을 위한 복제 그룹 사용 (p. 127) 단원을 참조하십시오.

독립 실행형 Redis(클러스터 모드 비활성화됨) 클러스터를 생성하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에 있는 Amazon ElastiCache 콘솔을 여십시오.
- 상단 오른쪽 모서리의 목록에서 이 클러스터를 시작할 AWS 리전을 선택합니다.
- 3. 탐색 창에서 [Redis]를 선택합니다.
- 4. Create를 선택합니다.
- 5. [Cluster engine]에 대해 [Redis]를 선택한 후 [Cluster Mode enabled (Scale Out)] 확인란의 선택을 취소 합니다.
- 6. [Redis settings] 단원을 완료합니다.
 - a. [Name]에 클러스터 이름을 입력합니다.

클러스터 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.
- b. [Description] 상자에 이 클러스터의 설명을 입력합니다.
- c. Engine version compatibility(엔진 버전 호환성)에서 이 클러스터에서 실행할 Redis용 ElastiCache 엔진 버전을 선택합니다. 이전 버전을 실행할 특별한 이유가 없는 한 최신 버전을 선택 하는 것이 좋습니다.

Important

새로운 엔진 버전으로 업그레이드할 수 있습니다. 이에 대한 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오. 하지만 기존의 클러스터를 삭제하고 새로 만들지 않는 한 이전 엔진 버전으로 다운그레이드할 수 없습니다.

새로운 Redis 버전은 보다 향상되고 안정적인 사용자 환경을 제공하므로 ElastiCache 콘솔을 사용할 경우 Redis 버전 2.6.13, 2.8.6 및 2.8.19는 더 이상 지원되지 않습니다. 이러한 Redis 버전은 사용하지 않을 것을 권장합니다. 이러한 버전 중 하나를 사용해야 하는 경우 AWS CLI 또는 ElastiCache API에서 작업하십시오.

자세한 정보는 다음 항목을 참조하십시오.

	AWS CLI	ElastiCache API
클러스터 생성	클러스터 생성 (AWS CLI) (p. 76)	클러스터 생성 (ElastiCache API) (p. 77)
	클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.	클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.
클러스터 수정	AWS CLI 사용 (p. 88) 클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.	ElastiCache API 사용 (p. 89) 클러스터 모드가 활성화된 상태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니다.
복제 그룹 생성	Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 149) Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154)	Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성 (p. 151) Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158)
복제 그룹 수정	AWS CLI 사용 (p. 166)	ElastiCache API 사용 (p. 166)

- d. 전송 중 데이터를 암호화하려면 [Encryption]에서 [Yes]를 선택합니다.
- e. [Encryption]에서 [Yes]를 선택하면 Redis 명령을 실행할 때 사용자에게 암호를 입력하도록 할 수 있습니다. 명령을 실행할 때 암호를 요구하도록 하려면 다음과 같이 수행하십시오.
 - i. [AUTH] 목록에서 [Yes]를 선택합니다.
 - ii. [AUTH token] 상자에 암호를 입력합니다.

ElastiCache에서 사용 시 AUTH 토큰 제약 조건

- 암호는 최소 16자, 최대 128자의 인쇄 가능한 문자여야 합니다.
- 허용되는 유일한 인쇄 가능한 특수 문자는 !, &, #, \$, ^, <, > 및 입니다. 다른 인쇄 가능한 특수 문자는 AUTH token에 사용할 수 없습니다.
- AUTH는 전송 중 암호화가 활성화된 클러스터를 생성하는 경우에만 활성화할 수 있습니다.
- 클러스터 생성 시 설정한 암호는 변경할 수 없습니다.

다음과 같이 보다 엄격한 정책을 따르는 것이 좋습니다.

- 다음 문자 유형 중 3개 이상을 포함하는 문자 조합을 포함해야 합니다.
 - 대문자
 - 소문자
 - 숫자
 - 영숫자 이외의 문자(!, &, #, \$, ^, <, >, -)
- 사전에 나오는 단어를 그대로 또는 약간 수정한 형태로 포함하면 안 됩니다.

- 이미 사용한 비밀번호와 같거나 비슷하면 안 됩니다.
- [Port]에서 기본 포트 6379를 허용합니다. 다른 포트를 사용하려면 포트 번호를 입력합니다.
- g. [Parameter group]에 대해 이 클러스터에 사용할 파라미터 그룹을 선택하거나 [Create new]를 선택 하여 이 클러스터에 사용할 새 파라미터 그룹을 생성합니다.

파라미터 그룹은 클러스터의 런타임 파라미터를 제어합니다. 파라미터 그룹에 대한 자세한 내용은 Redis 특정 파라미터 (p. 274) 및 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

h. 노드 유형에서 아래쪽 화살표(▼)를 클릭합니다. [Change node type] 대화 상자에서 원하는 노드 유 형의 인스턴스 패밀리를 선택하고 이 클러스터에 사용할 노드 유형을 선택한 다음 [Save]를 선택합 니다.

자세한 내용은 노드 크기 선택 (p. 67) 단원을 참조하십시오.

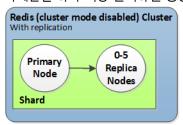
[Number of replicas]에 대해 이 클러스터의 읽기 전용 복제본 개수를 선택합니다.

[None]을 선택하면 [description] 및 [Multi-AZ with Auto-Failover] 필드가 사라지고 생성된 클러스터 가 다음과 같이 표시됩니다.



복제본 노드 없이 생성된 Redis(클러스터 모드 비활성화됨) 클러스터

복제본을 하나 이상 선택하면 생성된 클러스터가 다음과 같이 표시됩니다.



복제본 노드 없이 생성된 Redis(클러스터 모드 비활성화됨) 클러스터

- 7. [Advanced Redis settings]를 선택하고 단원을 완료합니다.
 - a. 복제본을 하나 이상 포함하기로 선택한 경우 [Multi-AZ with Auto-Failover] 확인란을 사용할 수 있습 니다. 다중 AZ 및 자동 장애 조치를 사용하는 것이 좋습니다. 자세한 내용은 Redis 실행 시 장애 완 화 (p. 395) 단원을 참조하십시오.
 - b. [Subnet group]에 대해 이 클러스터에 적용할 서브넷을 선택합니다.

자세한 내용은 서브넷 및 서브넷 그룹 (p. 329) 단원을 참조하십시오.

- c. [Availability zone(s)]에 대해 다음 두 가지 옵션이 있습니다.
 - 기본 설정 없음 ElastiCache에서 클러스터의 노드에 사용할 가용 영역을 선택합니다.
 - Specify availability zones(가용 영역 지정) 노드 목록이 나타나고 각 노드 이름 오른쪽의 목록에 서 가용 영역을 선택하여 클러스터에 있는 각 노드의 가용 영역을 지정할 수 있습니다.

자세한 내용은 리전 및 가용 영역 선택 (p. 47) 단원을 참조하십시오.

d. [Security groups]에 대해 이 클러스터에 사용할 보안 그룹을 선택합니다.

<u>자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316)</u> 단원을 참조하십시오. API 버전 2015-02-02

e. .RDB 파일의 데이터로 클러스터를 시드하려면 Seed RDB file S3 location(RDB 파일 S3 위치 지정) 상자에 .RDB 파일의 Amazon S3 위치를 입력합니다.

자세한 내용은 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211) 단원을 참조하십시오.

f. 정기적인 자동 백업을 예약하려면 [Enable automatic backups]를 선택한 다음 자동으로 삭제되기 전에 자동 백업을 보존할 기간(일)을 입력합니다. 정기적인 자동 백업을 예약하지 않으려면 [Enable automatic backups] 확인란의 선택을 취소합니다. 어떤 경우든 수동으로 삭제해야 하는 수동 백업 을 항상 생성할 수 있습니다.

Redis 백업 및 복원에 대한 자세한 내용은 Redis용 ElastiCache 백업 및 복원 (p. 187) 단원을 참조하십시오.

g. 유지 관리 기간은 ElastiCache가 클러스터의 시스템 유지 관리를 예약하는 시간이며 일반적으로 매주 한 시간입니다. ElastiCache에서 유지 관리 기간의 요일과 시간을 선택하도록 허용하거나(기본 설정 없음) 요일 시간 및 기간을 직접 선택할 수 있습니다(Specify maintenance window(유지관리 기간 설정)). 유지 관리 기간 지정을 선택한 경우 유지 관리 기간의 [Start day], [Start time] 및 [Duration](시간)을 선택합니다. 모든 시간은 UCT 시간입니다.

자세한 내용은 유지 관리 관리 중 (p. 50) 단원을 참조하십시오.

- h. [Notifications]에 대해 기존의 Amazon Simple Notification Service(Amazon SNS) 항목을 선택하거 나 수동 ARN 입력을 선택하고 Amazon 리소스 이름(ARN) 항목을 입력합니다. Amazon SNS를 통해 인터넷에 연결된 스마트 디바이스에 알림을 푸시할 수 있습니다. 기본적으로 알림이 비활성화됩니다. 자세한 내용은 https://aws.amazon.com/sns/를 참조하십시오.
- 8. 입력 및 선택한 내용을 모두 검토한 다음 돌아가 필요한 내용을 수정합니다. 준비가 되면 [Create]를 선택해 클러스터를 시작합니다.

클러스터 상태가 가용이 되면 Amazon EC2 액세스 권한을 부여하고 클러스터에 연결하며 사용할 수 있습니다. 자세한 내용은 2단계: 액세스 자격 부여 (p. 23) 및 3단계: 클러스터 노드에 연결 (p. 23) 단원을 참조하십시오.

Important

클러스터를 사용할 수 있게 되면 클러스터를 적극 사용하지 않더라도 클러스터가 활성화되어 있는 매 시간 또는 60분 미만 단위로 비용이 청구됩니다. 이 클러스터의 요금 발생을 중지하려면 클러스터를 삭제해야 합니다. 클러스터 삭제하기 (p. 102) 단원을 참조하십시오.

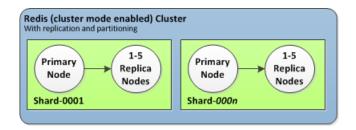
Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔)

Redis 3.2.4 이상을 실행 중이면 Redis(클러스터 모드 활성화됨) 클러스터를 생성할 수 있습니다. Redis(클러스터 모드 활성화됨) 클러스터는 현재 일부 제한이 있지만 1개부터 90개의 샤드(API/CLI: 노드 그룹)에 데이터를 분할하도록 지원합니다. Redis(클러스터 모드 비활성화됨)와 Redis(클러스터 모드 활성화됨)의 비교 내용은 지원되는 Redis용 ElastiCache 버전 (p. 37) 단원을 참조하십시오.

ElastiCache 관리 콘솔, ElastiCache용 AWS CLI 및 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터(API/CLI: 복제 그룹)를 생성할 수 있습니다.

ElastiCache 콘솔을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터를 만들려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에 있는 Amazon ElastiCache 콘솔을 여십시오.
- 2. 상단 오른쪽 모서리의 목록에서 이 클러스터를 시작할 AWS 리전을 선택합니다.
- 3. 탐색 창에서 [Redis]를 선택합니다.
- 4. Create를 선택합니다.
- 5. [Cluster engine]에 대해 [Redis]를 선택한 후 [Cluster Mode enabled (Scale Out)]를 선택합니다. 그러면 다음과 같이 표시되는 Redis(클러스터 모드 활성화됨) 클러스터가 생성됩니다.



복제 및 데이터 분할을 통해 생성된 Redis(클러스터 모드 활성화됨) 클러스터

- 6. Redis(클러스터 모드 활성화됨) 설정 단원을 완료합니다.
 - a. [Name] 상자에 클러스터 이름을 입력합니다.

클러스터 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.
- b. [Description] 상자에 이 클러스터의 설명을 입력합니다.
- c. 클러스터에 대해 전송 중 데이터 암호화를 활성화하려면 [In-transit encryption]을 선택합니다.

[In-transit encryption]을 선택하면 [Redis auth token]과 토큰(암호) 값을 입력할 수 있는 상자, 이렇게 두 가지 추가 옵션이 나타납니다.

- d. 클러스터에 대해 미사용 데이터 암호화를 활성화하려면 [At-rest encryption]을 선택합니다.
- e. 이 클러스터에서 작업을 수행하는 데 암호를 요구하도록 하려면:
 - i. [Redis auth token]을 선택합니다.
 - ii. [Redis auth token] 상자에 이 클러스터에서 작업을 수행하는 경우 사용해야 할 토큰(암호)을 입력합니다.

ElastiCache에서 사용 시 AUTH 토큰 제약 조건

- 암호는 최소 16자, 최대 128자의 인쇄 가능한 문자여야 합니다.
- 허용되는 유일한 인쇄 가능한 특수 문자는 !, &, #, \$, ^, <, > 및 입니다. 다른 인쇄 가능한 특수 문자는 AUTH token에 사용할 수 없습니다.
- AUTH는 전송 중 암호화가 활성화된 클러스터를 생성하는 경우에만 활성화할 수 있습니다.
- 클러스터 생성 시 설정한 암호는 변경할 수 없습니다.

다음과 같이 보다 엄격한 정책을 따르는 것이 좋습니다.

- 다음 문자 유형 중 3개 이상을 포함하는 문자 조합을 포함해야 합니다.
 - 대문자
 - 소문자
 - 숫자
 - 영숫자 이외의 문자(!, &, #, \$, ^, <, >, -)
- 사전에 나오는 단어를 그대로 또는 약간 수정한 형태로 포함하면 안 됩니다.
- 이미 사용한 비밀번호와 같거나 비슷하면 안 됩니다.
- f. Engine version compatibility(엔진 버전 호환성)에서 이 클러스터에서 실행할 Redis용

 ElastiCache 엔진 버전을 선택합니다. 있정 방전을 실행할 특별한 이유가 없는 한 최신 버전을 선택하는 것이 좋습니다.
 73

- g. 전송 중 데이터를 암호화하려면 [Encryption]에서 [Yes]를 선택합니다.
- h. [Encryption]에서 [Yes]를 선택하면 Redis 명령을 실행할 때 사용자에게 암호를 입력하도록 할 수 있 습니다. 명령을 실행할 때 암호를 요구하도록 하려면 다음과 같이 수행하십시오.
 - i. [AUTH] 목록에서 [Yes]를 선택합니다.
 - ii. [AUTH token] 상자에 암호를 입력합니다.

ElastiCache에서 사용 시 AUTH 토큰 제약 조건

- 암호는 최소 16자, 최대 128자의 인쇄 가능한 문자여야 합니다.
- 허용되는 유일한 인쇄 가능한 특수 문자는 !, &, #, \$, ^, <, > 및 입니다. 다른 인쇄 가능한 특수 문자는 AUTH token에 사용할 수 없습니다.
- AUTH는 전송 중 암호화가 활성화된 클러스터를 생성하는 경우에만 활성화할 수 있습니다.
- 클러스터 생성 시 설정한 암호는 변경할 수 없습니다.

다음과 같이 보다 엄격한 정책을 따르는 것이 좋습니다.

- 다음 문자 유형 중 3개 이상을 포함하는 문자 조합을 포함해야 합니다.
 - 대문자
 - 소문자
 - 숫자
 - 영숫자 이외의 문자(!, &, #, \$, ^, <, >, -)
- 사전에 나오는 단어를 그대로 또는 약간 수정한 형태로 포함하면 안 됩니다.
- 이미 사용한 비밀번호와 같거나 비슷하면 안 됩니다.
- i. [Port] 상자에서 기본 포트 6379를 허용합니다. 다른 포트를 사용하려면 포트 번호를 입력합니다.
- [Parameter group]에 대해 이 클러스터에 사용할 파라미터 그룹을 선택하거나 [Create new]를 선택 하여 이 클러스터에 사용할 새 파라미터 그룹을 생성합니다.

파라미터 그룹은 클러스터의 런타임 파라미터를 제어합니다. 파라미터 그룹에 대한 자세한 내용은 Redis 특정 파라미터 (p. 274) 및 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

k. 노드 유형에서 아래쪽 화살표(▼)를 선택합니다. [Change node type] 대화 상자에서 원하는 노드 유 형의 인스턴스 패밀리를 선택하고 이 클러스터에 사용할 노드 유형을 선택한 다음 [Save]를 선택합 니다.

자세한 내용은 노드 크기 선택 (p. 67) 단원을 참조하십시오.

I. 샤드 수에서 이 Redis(클러스터 모드 활성화됨) 클러스터에 사용할 샤드(파티션/노드 그룹) 수를 선 택합니다.

Redis(클러스터 모드 활성화됨)에서는 클러스터에서 실행 중인 Redis의 버전에 따라 클러스터 내 샤드 수를 동적으로 변경할 수 있습니다.

- Redis 3.2.10—클러스터가 Redis 3.2.10을 실행 중인 경우에는 클러스터 내 샤드 수를 동적 으로 변경할 수 있습니다. 자세한 내용은 Redis(클러스터 모드 활성화됨)에서 클러스터 조 정 (p. 241) 단원을 참조하십시오.
- 기타 버전의 Redis—클러스터가 3.2.10 이외의 Redis 버전을 실행 중인 경우에 클러스터 내에서 샤드 수를 변경하려면 샤드 수를 바꿔서 클러스터를 새로 생성해야 합니다. 자세한 내용은 선택적 으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.
- m. [Replicas per shard]에 대해 각 샤드에 포함할 읽기 전용 복제본 노드 수를 선택합니다.

Redis(클러스터 모드 활성화됨)에는 다음과 같은 제한 사항이 있습니다.

• 콘솔을 사용하여 클러스터를 생성할 때 샤드마다 복제본 수가 동일합니다.

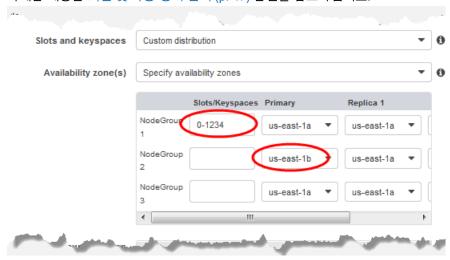
- 샤드당 읽기 전용 복제본 수가 고정되어 변경할 수 없습니다. 샤드(API/CLI: 노드 그룹)당 복제본 수를 늘리거나 줄이려면 새로운 복제본 수로 새 클러스터를 생성해야 합니다. 자세한 내용은 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211) 단원을 참조하십시오.
- n. [Subnet group]에 대해 이 클러스터에 적용할 서브넷을 선택합니다.

자세한 내용은 서브넷 및 서브넷 그룹 (p. 329) 단원을 참조하십시오.

- 7. [Advanced Redis settings]를 클릭하고 단원을 완료합니다.
 - a. [Slots and keyspaces]에 대해 샤드(파티션)에 키를 배포할 방법을 선택합니다. 배포할 키가 16,384 개 있습니다(0에서 16383개까지 번호 지정).
 - Equal distribution(균등 배포) ElastiCache는 가능한 한 균등하게 샤드에 키스페이스를 배포합니다.
 - Custom distribution(사용자 지정 배포) 가용 영역 아래의 표에서 각 샤드에 대한 키 범위를 지정합니다.
 - b. [Availability zone(s)]에 대해 다음 두 가지 옵션이 있습니다.
 - 기본 설정 없음 ElastiCache에서 가용 영역을 선택합니다.
 - Specify availability zones(가용 영역 설정) 각 클러스터의 가용 영역을 지정합니다.

가용 영역을 지정하도록 선택한 경우 샤드에 있는 각 클러스터에 대해 목록에서 가용 영역을 선택 합니다.

자세한 내용은 리전 및 가용 영역 선택 (p. 47) 단원을 참조하십시오.



키스페이스 및 가용 영역 지정

c. [Security groups]에 대해 이 클러스터에 사용할 보안 그룹을 선택합니다.

자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

d. .RDB 파일의 데이터로 클러스터를 시드하면 [Seed RDB file S3 location] 상자에 .RDB 파일의 S3 위치를 입력합니다.

자세한 내용은 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨)에서는 노드 그룹마다 별도의 .RDB 파일이 있어야 합니다.

e. 정기적인 자동 백업을 예약하려면 [Enable automatic backups]를 선택한 후 자동으로 삭제되기 전에 각 자동 백업을 보존할 기간(일)을 입력합니다. 정기적인 자동 백업을 예약하지 않으려면 [Enable automatic backups] 확인란의 선택을 취소합니다. 어떤 경우든 수동 백업을 항상 생성할 수 있습니다.

Redis 백업 및 복원에 대한 자세한 내용은 Redis용 ElastiCache 백업 및 복원 (p. 187) 단원을 참조하십시오.

f. 유지 관리 기간은 ElastiCache가 클러스터의 시스템 유지 관리를 예약하는 시간이며 일반적으로 매주 한 시간입니다. ElastiCache에서 유지 관리 기간의 요일과 시간을 선택하도록 허용하거나(기본 설정 없음) 요일 시간 및 기간을 직접 선택할 수 있습니다(Specify maintenance window(유지 관리 기간 설정)). [Specify maintenance window]를 선택할 경우 목록에서 유지 관리 기간의 [Start day], [Start time] 및 [Duration](시간)을 선택합니다. 모든 시간은 UCT 시간입니다.

자세한 내용은 유지 관리 관리 중 (p. 50) 단원을 참조하십시오.

- g. 알림에 대해 기존의 Amazon Simple Notification Service (Amazon SNS) 항목을 선택하거나 수동 ARN 입력을 선택하고 Amazon 리소스 이름(ARN) 항목을 입력합니다. Amazon SNS를 통해 인터넷에 연결된 스마트 디바이스에 알림을 푸시할 수 있습니다. 기본적으로 알림이 비활성화됩니다. 자세한 내용은 https://aws.amazon.com/sns/를 참조하십시오.
- 8. 입력 및 선택한 내용을 모두 검토한 다음 돌아가 필요한 내용을 수정합니다. 준비가 되면 [Create cluster]를 선택하여 클러스터를 시작하거나 [Cancel]을 선택해 작업을 취소합니다.

ElastiCache 콘솔 대신 ElastiCache API 또는 AWS CLI를 사용하여 동일한 결과를 얻으려면 다음을 참조하십시오.

- API: CreateReplicationGroup
- CLI: create-replication-group

클러스터 상태가 [available]이 되면 클러스터에 EC2 액세스 권한을 부여하고 클러스터에 연결하며 사용할수 있습니다. 자세한 내용은 2단계: 액세스 자격 부여 (p. 23) 및 3단계: 클러스터 노드에 연결 (p. 23) 단원을 참조하십시오.

Important

클러스터를 사용할 수 있게 되면 클러스터를 적극 사용하지 않더라도 클러스터가 활성화되어 있는 매 시간 또는 60분 미만 단위로 비용이 청구됩니다. 이 클러스터의 요금 발생을 중지하려면 클러스터를 삭제해야 합니다. 클러스터 삭제하기 (p. 102) 단원을 참조하십시오.

클러스터 생성 (AWS CLI)

AWS CLI를 사용하여 클러스터를 생성하려면 create-cache-cluster 명령을 사용하십시오.

Important

클러스터를 사용할 수 있게 되면 클러스터를 적극 사용하지 않더라도 클러스터가 활성화되어 있는 매 시간 또는 60분 미만 단위로 비용이 청구됩니다. 이 클러스터의 요금 발생을 중지하려면 클러스터를 삭제해야 합니다. 클러스터 삭제하기 (p. 102) 단원을 참조하십시오.

주제

- Redis(클러스터 모드 비활성화됨) 캐시 클러스터 생성 (AWS CLI) (p. 77)
- Redis(클러스터 모드 활성화됨) 클러스터 생성 (AWS CLI) (p. 77)

Redis(클러스터 모드 비활성화됨) 캐시 클러스터 생성 (AWS CLI)

Example – 읽기 전용 복제본이 없는 Redis(클러스터 모드 비활성화됨) 클러스터 다음 CLI 코드는 복제본 없이 Redis(클러스터 모드 비활성화됨) 캐시 클러스터를 생성합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-cache-cluster \
--cache-cluster-id my-cluster \
--cache-node-type cache.r4.large \
--engine redis \
--engine-version 3.2.4 \
--num-cache-nodes 1 \
--cache-parameter-group default.redis3.2 \
--snapshot-arns arn:aws:s3:myS3Bucket/snap.rdb
```

Windows의 경우:

```
aws elasticache create-cache-cluster ^
--cache-cluster-id my-cluster ^
--cache-node-type cache.r4.large ^
--engine redis ^
--engine-version 3.2.4 ^
--num-cache-nodes 1 ^
--cache-parameter-group default.redis3.2 ^
--snapshot-arns arn:aws:s3:myS3Bucket/snap.rdb
```

Redis(클러스터 모드 활성화됨) 클러스터 생성 (AWS CLI)

create-cache-cluster 작업을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터(API/CLI: 복제 그룹)를 생성할 수 없습니다. Redis(클러스터 모드 활성화됨) 클러스터(API/CLI: 복제 그룹)를 생성하려면 Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154) 단원을 참조하십시오.

자세한 내용은 AWS CLI에서 ElastiCache 참조 항목 create-replication-group을 참조하십시오.

클러스터 생성 (ElastiCache API)

ElastiCache API를 사용하여 클러스터를 생성하려면 CreateCacheCluster 작업을 사용하십시오. Important

클러스터를 사용할 수 있게 되면 클러스터를 사용하지 않더라도 클러스터가 활성화되어 있는 매 시간 또는 60분 미만 단위로 비용이 청구됩니다. 이 클러스터의 요금 발생을 중지하려면 클러스터를 삭제해야 합니다. 클러스터 삭제하기 (p. 102) 단원을 참조하십시오.

주제

- Redis(클러스터 모드 비활성화됨) 캐시 클러스터 생성 (ElastiCache API) (p. 77)
- Redis(클러스터 모드 활성화됨)에서 캐시 클러스터 생성 (ElastiCache API) (p. 78)

Redis(클러스터 모드 비활성화됨) 캐시 클러스터 생성 (ElastiCache API)

다음 코드는 Redis(클러스터 모드 비활성화됨) 캐시 클러스터 (ElastiCache API)를 생성합니다.

읽기 쉽도록 줄바꿈이 추가됩니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=CreateCacheCluster
```

&CacheClusterId=my-cluster &CacheNodeType=cache.r4.large &CacheParameterGroup=default.redis3.2 &Engine=redis &EngineVersion=3.2.4 &NumCacheNodes=1 &SignatureVersion=4 &SignatureMethod=HmacSHA256 &SnapshotArns.member.1=arn%3Aaws%3As3%3A%3AmyS3Bucket%2Fdump.rdb &Timestamp=20150508T220302Z &Version=2015-02-02 &X-Amz-Algorithm=AWS4-HMAC-SHA256 &X-Amz-Credential=<credential> &X-Amz-Date=20150508T220302Z &X-Amz-Expires=20150508T220302Z &X-Amz-SignedHeaders=Host &X-Amz-Signature=<signature>

Redis(클러스터 모드 활성화됨)에서 캐시 클러스터 생성 (ElastiCache API)

CreateCacheCluster 작업을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터(API/CLI: 복제 그룹)를 생성할 수 없습니다. Redis(클러스터 모드 활성화됨) 클러스터(API/CLI: 복제 그룹)를 생성하려면 Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158) 단원을 참조하십시오.

자세한 내용은 ElastiCache API 참조 항목 CreateReplicationGroup을 참조하십시오.

클러스터 세부 정보 보기

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 클러스터 하나 이상에 대한 세부 정보를 볼 수 있습니다.

주제

- Redis(클러스터 모드 비활성화됨) 클러스터 세부 정보 보기 (콘솔) (p. 78)
- Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (콘솔) (p. 79)
- 클러스터 세부 정보 보기 (AWS CLI) (p. 80)
- 클러스터 세부 정보 보기 (ElastiCache API) (p. 86)

Redis(클러스터 모드 비활성화됨) 클러스터 세부 정보 보기 (콘솔)

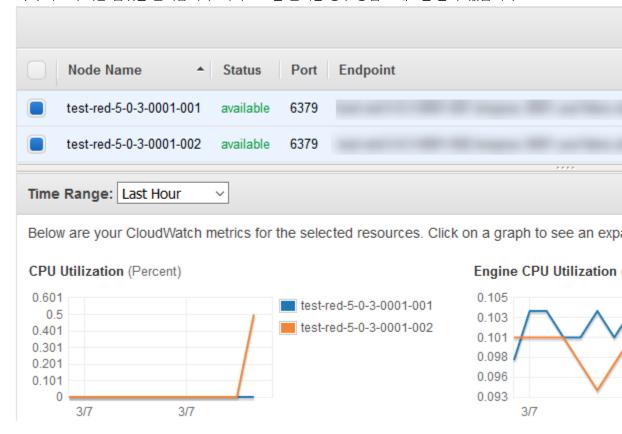
ElastiCache 콘솔, ElastiCache용 AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 비활성 화됨) 클러스터의 세부 정보를 볼 수 있습니다.

다음 절차에서는 ElastiCache 콘솔을 사용하여 Redis(클러스터 모드 비활성화됨) 클러스터의 세부 정보를 보 는 방법을 자세히 설명합니다.

Redis(클러스터 모드 비활성화됨) 클러스터의 세부 정보를 보려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에 있는 Amazon ElastiCache 콘솔을 여십시오.
- 2. ElastiCache 콘솔 대시보드에서 Redis를 선택하여 Redis 버전을 실행하는 모든 클러스터의 목록을 표시합니다.
- 3. 클러스터의 세부 정보를 보려면 클러스터 이름의 왼쪽에 있는 확인란을 선택합니다. Clustered Redis가 아니라 Redis 엔진을 실행하는 클러스터를 선택해야 합니다. 그러면 클러스터의 기본 엔드포인트를 포함하여 클러스터에 대한 세부 정보가 표시됩니다.
- 4. 노드 정보를 보려면

- a. 클러스터의 이름을 선택합니다.
- b. [Nodes] 탭을 선택합니다. 그러면 클러스터에서 읽으려면 사용해야 하는 노드의 엔드포인트를 포함 하여 각 노드의 세부 정보가 표시됩니다.
- c. 하나 이상의 노드에서 지표를 보려면 노드 ID의 왼쪽에 있는 상자를 선택한 다음 [Time range] 목록 에서 지표의 시간 범위를 선택합니다. 여러 노드를 선택할 경우 중첩 그래프를 볼 수 있습니다.



Redis 노드 2개에 대한 지난 1시간의 측정치

Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (콘솔)

ElastiCache 콘솔, ElastiCache용 AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터의 세부 정보를 볼 수 있습니다.

다음 절차에서는 ElastiCache 콘솔을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터의 세부 정보를 보는 방법을 자세히 설명합니다.

Redis(클러스터 모드 활성화됨) 클러스터의 세부 정보를 보려면

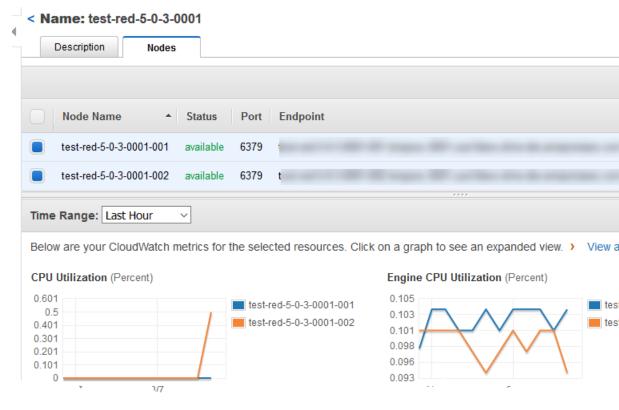
- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에 있는 Amazon ElastiCache 콘솔을 여십시오.
- 2. 상단 오른쪽 모서리의 목록에서 원하는 AWS 리전을 선택합니다.
- 3. ElastiCache 콘솔 대시보드에서 Redis를 선택하여 Redis 버전을 실행하는 모든 클러스터의 목록을 표시합니다.
- 4. Redis(클러스터 모드 활성화됨) 클러스터의 세부 정보를 보려면 클러스터 이름의 왼쪽에 있는 상자를 선택합니다. Redis뿐 아니라 Clustered Redis 엔진을 실행하는 클러스터를 선택해야 합니다.

화면에서 아래의 클러스터를 확대하여 클러스터 구성 엔드포인트를 포함해 클러스터에 대한 세부 정보를 표시합니다.

- 5. 클러스터의 샤드와 각 샤드의 노드 수 목록을 보려면 클러스터 이름을 선택합니다.
- 6. 노드의 특정 정보를 보려면
 - a. 샤드 ID를 선택합니다.
 - b. [Nodes] 탭을 선택합니다.

그러면 클러스터에서 데이터를 읽는 데 사용할 각 노드의 엔드포인트를 포함하여 각 노드에 대한 정 보가 표시됩니다.

c. 노드 1개 이상의 지표를 보려면 노드 ID 왼쪽의 상자를 선택한 후 [Time range] 목록에서 지표의 시간 범위를 선택합니다. 여러 노드를 선택하면 중첩 그래프가 생성됩니다.



Redis 노드 2개에 대한 지난 1시간의 지표

클러스터 세부 정보 보기 (AWS CLI)

AWS CLI describe-cache-clusters 명령을 사용하여 클러스터의 세부 정보를 볼 수 있습니다. --cache-cluster-id 파라미터가 생략되면 여러 클러스터(최대 --max-items개)의 세부 정보가 반환됩니다. --cache-cluster-id 파라미터가 포함되면 지정한 클러스터의 세부 정보가 반환됩니다. --max-items 파라미터를 사용하여 반환되는 레코드 수를 제한할 수 있습니다.

다음 코드는 my-cluster의 세부 정보를 나열합니다.

```
aws elasticache describe-cache-clusters --cache-cluster-id my-cluster
```

다음 코드는 클러스터 최대 25개의 세부 정보를 나열합니다.

```
aws elasticache describe-cache-clusters --max-items 25
```

Example

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-cache-clusters \
    --cache-cluster-id my-cluster \
    --show-cache-node-info
```

Windows의 경우:

```
aws elasticache describe-cache-clusters ^
--cache-cluster-id my-cluster ^
--show-cache-node-info
```

이 작업은 다음과 비슷한 출력을 생성합니다(JSON 형식).

다음 JSON 출력은 세 가지 Redis 클러스터에 대한 것입니다.

- my-cluster1 노드가 1개인 Redis(클러스터 모드 비활성화됨) 클러스터.
- my-cluster2 노드가 2개인 Redis(클러스터 모드 비활성화됨) 클러스터(기본 노드 1개, 읽기 전용 복제 본 노드 1개).
- my-cluster22 -샤드가 2개인 Redis(클러스터 모드 활성화됨) 클러스터(각 샤드에는 기본 노드 1개와 읽기 전용 복제본 노드 2개가 포함). 이 클러스터의 암호화는 활성화되었습니다.

```
"CacheClusters": [
        {
            "CacheClusterStatus": "available",
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "wed:12:00-wed:13:00",
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "08:30-09:30",
            "TransitEncryptionEnabled": false,
            "AtRestEncryptionEnabled": false,
            "CacheClusterId": "my-cluster1",
            "CacheClusterCreateTime": "2018-02-26T21:06:43.420Z",
            "PreferredAvailabilityZone": "us-west-2c",
            "AuthTokenEnabled": false,
            "PendingModifiedValues": {}
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            "SnapshotRetentionLimit": 0,
            "AutoMinorVersionUpgrade": true,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
```

```
"NumCacheNodes": 1
```

```
"CacheClusters": [
            "SecurityGroups": [
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": false,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": false,
            "PreferredAvailabilityZone": "us-west-2a",
            "TransitEncryptionEnabled": false,
            "ReplicationGroupId": "my-cluster2",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
            "CacheClusterId": "my-cluster2-001",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": false,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": false,
            "PreferredAvailabilityZone": "us-west-2b",
            "TransitEncryptionEnabled": false,
            "ReplicationGroupId": "my-cluster2",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
            "CacheClusterId": "my-cluster2-002",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
```

```
"CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
            "SecurityGroups": [
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": false,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": false,
            "PreferredAvailabilityZone": "us-west-2c",
            "TransitEncryptionEnabled": false,
            "ReplicationGroupId": "my-cluster2",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
            "CacheClusterId": "my-cluster2-003",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
        }
```

```
"CacheClusters": [
        {
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": true,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": true,
            "PreferredAvailabilityZone": "us-west-2a",
            "TransitEncryptionEnabled": true,
```

```
"ReplicationGroupId": "my-cluster3",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
            "CacheClusterId": "my-cluster3-0001-001",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2.cluster.on"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.6",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
        },
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": true,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": true,
            "PreferredAvailabilityZone": "us-west-2b",
            "TransitEncryptionEnabled": true,
            "ReplicationGroupId": "my-cluster3",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
            "CacheClusterId": "my-cluster3-0001-002",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2.cluster.on"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.6",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
        },
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            ٦,
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": true,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": true,
```

```
"PreferredAvailabilityZone": "us-west-2c",
            "TransitEncryptionEnabled": true,
            "ReplicationGroupId": "my-cluster3",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
            "CacheClusterId": "my-cluster3-0001-003",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2.cluster.on"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.6",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
       },
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
                }
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": true,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": true,
            "PreferredAvailabilityZone": "us-west-2b",
            "TransitEncryptionEnabled": true,
            "ReplicationGroupId": "my-cluster3",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
            "CacheClusterId": "my-cluster3-0002-001",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2.cluster.on"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.6",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
       },
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": true,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
```

```
"CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": true,
            "PreferredAvailabilityZone": "us-west-2c",
            "TransitEncryptionEnabled": true,
            "ReplicationGroupId": "my-cluster3",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
            "CacheClusterId": "my-cluster3-0002-002",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2.cluster.on"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.6",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
            "SecurityGroups": [
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": true,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": true,
            "PreferredAvailabilityZone": "us-west-2a",
            "TransitEncryptionEnabled": true,
            "ReplicationGroupId": "my-cluster3",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
            "CacheClusterId": "my-cluster3-0002-003",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2.cluster.on"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.6",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
        }
    ]
```

자세한 내용은 AWS CLI의 ElastiCache 항목 describe-cache-clusters를 참조하십시오.

클러스터 세부 정보 보기 (ElastiCache API)

ElastiCache API DescribeCacheClusters 작업을 사용하여 클러스터의 세부 정보를 볼 수 있습니다. CacheClusterId 파라미터가 포함되면 지정한 클러스터의 세부 정보가 반환됩니다. CacheClusterId 파

라미터가 생략되면 클러스터 최대 MaxRecords개(기본값 100)의 세부 정보가 반환됩니다. MaxRecords의 값은 20 이상 또는 100 이하여야 합니다.

다음 코드는 my-cluster의 세부 정보를 나열합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheClusters
&CacheClusterId=my-cluster
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

다음 코드는 클러스터 최대 25개의 세부 정보를 나열합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheClusters
&MaxRecords=25
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 ElastiCache API 참조 항목 DescribeCacheClusters을 참조하십시오.

ElastiCache 클러스터 수정

클러스터에서 노드를 추가하거나 제거하는 것 외에도 보안 그룹을 추가하거나 유지 관리 기간 또는 파라미터 그룹을 변경하는 등 기존의 클러스터를 변경해야 할 경우도 있습니다.

유지 관리 기간을 사용률이 가장 낮은 시간으로 낮추는 것이 유익하므로 수정해야 할 때도 있습니다.

클러스터의 파라미터를 변경하면 변경 사항은 또는 클러스터가 재시작된 즉시 또는 그 이후에 클러스터에 적용됩니다. 이는 클러스터의 파라미터 그룹 자체에서 변경하든 파라미터 값을 클러스터의 파라미터 그룹 내에서 변경하든 마찬가지입니다. 특정 파라미터의 변경 사항이 적용되는 시점을 확인하려면 Redis 특정 파라미터 (p. 274)에 대한 표에서 변경 사항 적용 열을 참조하십시오. 클러스터 재부팅에 관한 자세한 내용은 클러스터 재부팅 (p. 89) 단원을 참조하십시오.

AWS Management 콘솔 사용

클러스터를 삭제하려면 (console)

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 상단 오른쪽 모서리의 목록에서 수정하려는 클러스터가 있는 AWS 리전을 선택합니다.
- 3. 탐색 창에서 수정하려는 클러스터에서 실행 중인 엔진을 선택합니다.

선택한 엔진의 클러스터 목록이 나타납니다.

- 4. 클러스터 목록에서 수정하고자 하는 클러스터의 이름 왼쪽에 있는 상자가 아닌 클러스터의 이름을 선택 합니다.
- 5. [Modify]를 선택합니다.

[Modify Cluster] 창이 나타납니다.

6. [Modify Cluster] 창에서 원하는 내용을 수정합니다.

Important

새로운 엔진 버전으로 업그레이드할 수 있습니다. 이에 대한 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오. 하지만 기존의 클러스터를 삭제하고 새로 만들지 않는 한이전 엔진 버전으로 다운그레이드할 수 없습니다.

새로운 Redis 버전은 보다 향상되고 안정적인 사용자 환경을 제공하므로 ElastiCache 콘솔을 사용할 경우 Redis 버전 2.6.13, 2.8.6 및 2.8.19는 더 이상 지원되지 않습니다. 이러한 Redis 버전은 사용하지 않을 것을 권장합니다. 이러한 버전 중 하나를 사용해야 하는 경우 AWS CLI 또는 ElastiCache API에서 작업하십시오.

자세한 정보는 다음 항목을 참조하십시오.

	AWS CLI	ElastiCache API
클러스터 생성	클러스터 생성 (AWS CLI) (p. 76)	클러스터 생성 (ElastiCache API) (p. 77)
	클러스터 모드가 활성화된 상태 에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니다.	클러스터 모드가 활성화된 상태 에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니다.
클러스터 수정	AWS CLI 사용 (p. 88)	ElastiCache API 사용 (p. 89)
	클러스터 모드가 활성화된 상태 에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니다.	클러스터 모드가 활성화된 상태 에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니다.
복제 그룹 생성	Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 149) Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154)	Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성 (p. 151)
		Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158)
복제 그룹 수정	AWS CLI 사용 (p. 166)	ElastiCache API 사용 (p. 166)

[Apply Immediately] 상자는 엔진 버전 수정에만 적용됩니다. 변경 사항을 즉시 적용하려면 Apply Immediately(즉시 적용) 확인란을 선택합니다. 이 상자를 선택하지 않으면 다음 번 유지 관리 기간에 노드 유형 및 엔진 버전 수정이 적용됩니다. 유지 관리 기간 변경과 같은 다른 수정은 즉시 적용됩니다.

7. [Modify]를 선택합니다.

AWS CLI 사용

AWS CLI modify-cache-cluster 작업을 사용하여 기존의 클러스터를 수정할 수 있습니다. 클러스터의 구성 값을 수정하려면 클러스터 ID, 변경할 파라미터 및 파라미터의 새 값을 지정합니다. 다음 예제에서는 my-cluster라는 클러스터의 유지 관리 기간을 변경하고 변경 사항을 즉시 적용합니다.

Important

새로운 엔진 버전으로 업그레이드할 수 있습니다. 이에 대한 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오. 하지만 기존의 클러스터 또는 복제 그룹을 삭제하고 새로 만들지 않는 한 이전 엔진 버전으로 다운그레이드할 수 없습니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-cluster \
--cache-cluster-id my-cluster \
--preferred-maintenance-window sun:23:00-mon:02:00
```

Windows의 경우:

```
aws elasticache modify-cache-cluster ^
--cache-cluster-id my-cluster ^
--preferred-maintenance-window sun:23:00-mon:02:00
```

--apply-immediately 파라미터는 노드 유형, 엔진 버전의 수정 및 클러스터의 노드 수 변경에만 적용됩니다. 이 변경 사항을 즉시 적용하려면 --apply-immediately 파라미터를 사용하십시오. 다음 번 유지 관리 기간으로 이 변경을 연기하려면 --no-apply-immediately 파라미터를 사용하십시오. 유지 관리 기간 변경과 같은 다른 수정은 즉시 적용됩니다.

자세한 내용은 AWS CLI의 ElastiCache 항목 modify-cache-cluster를 참조하십시오.

ElastiCache API 사용

ElastiCache API ModifyCacheCluster 작업을 사용하여 기존의 클러스터를 수정할 수 있습니다. 클러스터의 구성 값을 수정하려면 클러스터 ID, 변경할 파라미터 및 파라미터의 새 값을 지정합니다. 다음 예제에서는 my-cluster라는 클러스터의 유지 관리 기간을 변경하고 변경 사항을 즉시 적용합니다.

Important

새로운 엔진 버전으로 업그레이드할 수 있습니다. 이에 대한 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오. 하지만 기존의 클러스터 또는 복제 그룹을 삭제하고 새로 만들지 않는 한 이전 엔진 버전으로 다운그레이드할 수 없습니다.

읽기 쉽도록 줄바꿈이 추가됩니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=ModifyCacheCluster
    &CacheClusterId=my-cluster
    &PreferredMaintenanceWindow=sun:23:00-mon:02:00
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150901T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20150202T220302Z
    &X-Amz-DignedHeaders=Host
    &X-Amz-Expires=20150901T220302Z
    &X-Amz-Expires=20150901T220302Z
    &X-Amz-Credential=<credential>
    &X-Amz-Signature=<signature>
```

ApplyImmediately 파라미터는 노드 유형, 엔진 버전의 수정 및 클러스터의 노드 수 변경에만 적용됩니다. 이 변경 사항을 즉시 적용하려면 ApplyImmediately 파라미터를 true로 설정하십시오. 다음 번 유지 관리 기간으로 이 변경을 연기하려면 ApplyImmediately 파라미터를 false로 설정하십시오. 유지 관리 기간 변경과 같은 다른 수정은 즉시 적용됩니다.

자세한 내용은 ElastiCache API 참조 항목 ModifyCacheCluster을 참조하십시오.

클러스터 재부팅

일부 변경 사항은 클러스터를 재부팅해야 적용됩니다. 예를 들어, 일부 파라미터는 파라미터 그룹의 파라미터 값을 변경할 경우 재부팅해야 변경 사항이 적용됩니다.

클러스터를 재부팅할 때 클러스터가 모든 데이터를 플러시하고 해당 엔진을 다시 시작합니다. 이 프로세스 중에는 클러스터에 액세스할 수 없습니다. 클러스터가 해당 데이터를 모두 플러시하기 때문에 클러스터가 다 시 사용 가능한 상태가 되면 빈 클러스터로 시작됩니다.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 클러스터를 재부팅할 수 있습니다. ElastiCache 콘솔, AWS CLI 또는 ElastiCache API 중 어느 것을 사용하건 단일 클러스터 재부팅만 시작할 수 있습니다. 여러 클러스터를 재부팅하려면 프로세스나 작업에서 반복해야 합니다.

Redis(클러스터 모드 활성화됨) 및 재부팅

Redis(클러스터 모드 활성화됨) 클러스터 재부팅해야 변경 사항이 적용되는 파라미터를 변경할 경우 다음 단계를 따르십시오.

- 1. 클러스터의 수동 백업을 만듭니다. 수동 백업 만들기 (p. 190) 단원을 참조하십시오.
- 2. Redis(클러스터 모드 활성화됨) 클러스터를 삭제하십시오. 클러스터 삭제하기 (p. 102) 단원을 참조하십시오.
- 3. 변경한 파라미터 그룹 및 백업을 사용해 클러스터를 복원하여 새로운 클러스터를 시드합니다. 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.

AWS Management 콘솔 사용

ElastiCache 콘솔을 사용하여 클러스터를 재부팅할 수 있습니다.

클러스터를 재부팅하려면 (console)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 상단 오른쪽 모서리의 목록에서 원하는 AWS 리전을 선택합니다.
- 3. 재부팅하려는 클러스터에서 실행 중인 엔진을 탐색 창에서 선택합니다.

선택한 엔진을 실행하는 모든 클러스터의 목록이 표시됩니다.

4. 클러스터 이름 왼쪽의 상자를 선택하여 재부팅할 클러스터를 선택합니다.

[Reboot] 버튼이 활성화됩니다.

클러스터를 2개 이상 선택하면 [Reboot] 버튼이 비활성화됩니다.

5. [Reboot]를 선택합니다.

클러스터 재부팅 확인 화면이 나타납니다.

6. 클러스터를 재부팅하려면 [Reboot]를 선택합니다. 클러스터 상태가 [rebooting cluster nodes]로 변경됩니다.

클러스터를 재부팅하지 않으려면 [Cancel]을 선택합니다.

여러 클러스터를 재부팅하려면 재부팅할 클러스터마다 2단계에서 5단계까지 반복합니다. 한 클러스터의 재 부팅이 완료되기를 기다렸다가 그다음 클러스터를 재부팅할 필요는 없습니다.

AWS CLI 사용

클러스터 (AWS CLI)를 재부팅하려면 reboot-cache-cluster CLI 작업을 사용하십시오.

클러스터의 특정 노드를 재부팅하려면 --cache-node-ids-to-reboot를 사용하여 재부팅할 특정 클러 스터를 나열하십시오. 다음 명령을 통해 my-cluster의 노드 0001, 0002 및 0004가 재부팅됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache reboot-cache-cluster \
--cache-cluster-id my-cluster \
--cache-node-ids-to-reboot 0001 0002 0004
```

Windows의 경우:

```
aws elasticache reboot-cache-cluster ^
--cache-cluster-id my-cluster ^
--cache-node-ids-to-reboot 0001 0002 0004
```

클러스터의 모든 노드를 재부팅하려면 --cache-node-ids-to-reboot 파라미터를 사용하고 클러스터의 모든 노드 ID를 나열하십시오. 자세한 내용은 reboot-cache-cluster를 참조하십시오.

ElastiCache API 사용

ElastiCache API를 사용하여 클러스터를 재부팅하려면 RebootCacheCluster 작업을 사용하십시오.

클러스터의 특정 노드를 재부팅하려면 CacheNodeIdsToReboot를 사용하여 재부팅할 특정 클러스터를 나열하십시오. 다음 명령을 통해 my-cluster의 노드 0001, 0002 및 0004가 재부팅됩니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=RebootCacheCluster
    &CacheClusterId=my-cluster
    &CacheNodeIdsToReboot.member.1=0001
    &CacheNodeIdsToReboot.member.2=0002
    &CacheNodeIdsToReboot.member.3=0004
    &Version=2015-02-02
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T192317Z
    &X-Amz-Credential=<credential>
```

클러스터의 모든 노드를 재부팅하려면 CacheNodeIdsToReboot 파라미터를 사용하고 클러스터의 모든 노드 ID를 나열하십시오. 자세한 내용은 RebootCacheCluster를 참조하십시오.

클러스터에 노드 추가

Redis(클러스터 모드 활성화됨) 클러스터를 재구성하려면 Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241) 단원을 참조하십시오.

ElastiCache 관리 콘솔 AWS CLI 또는 ElastiCache API를 사용하여 클러스터에 노드를 추가할 수 있습니다.

AWS Management 콘솔 사용

단일 노드 Redis(클러스터 모드 비활성화됨) 클러스터(복제가 활성화되지 않은 클러스터)에 노드를 추가하려면 복제를 추가한 후 복제본 노드를 추가하는 2단계 프로세스를 진행합니다.

항목

- 샤드 없이 Redis 클러스터에 복제를 추가하려면 (p. 92)
- 클러스터 (console)에 노드를 추가하려면 (p. 92)

다음 절차에서는 복제가 활성화되지 않은 단일 노드 Redis에 복제를 추가합니다. 복제를 추가할 때 기존의 노드는 복제가 활성화된 클러스터의 기본 노드가 됩니다. 복제를 추가한 후에는 최대 5개의 복제본 노드를 클러스터에 추가할 수 있습니다.

샤드 없이 Redis 클러스터에 복제를 추가하려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.

Redis 엔진을 실행하는 클러스터 목록이 표시됩니다.

3. 노드를 추가하고자 하는 클러스터의 이름 왼쪽에 있는 상자가 아닌 클러스터의 이름을 선택합니다.

복제가 활성화되지 않은 Redis 클러스터에 다음 내용이 적용됩니다.

- Clustered Redis가 아니라 Redis를 실행합니다.
- 제로 샤드를 포함합니다.

클러스터에 샤드가 있으면 복제가 이미 활성화되어 있으며 클러스터 (console)에 노드를 추가하려면 (p. 92)에서 계속할 수 있습니다.

- 4. [Add replication]을 선택합니다.
- 5. 복제가 활성화된 이 클러스터의 설명을 [Add Replication]에 입력합니다.
- 6. [추가]를 선택합니다.

클러스터 상태가 [available]로 돌아오면 다음 절차에서 계속 진행하고 복제본을 클러스터에 추가할 수 있습니다.

클러스터 (console)에 노드를 추가하려면

다음 절차에 따라 클러스터에 노드를 추가할 수 있습니다.

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 노드를 추가하려는 클러스터에서 실행 중인 엔진을 탐색 창에서 선택합니다.

선택한 엔진을 실행하는 클러스터 목록이 표시됩니다.

3. 클러스터 목록에서 노드를 추가하고자 하는 클러스터의 이름 왼쪽에 있는 상자가 아닌 클러스터의 이름 을 선택합니다.

클러스터가 Redis(클러스터 모드 활성화됨) 클러스터인 경우에는 Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241) 단원을 참조하십시오.

클러스터가 제로 샤드 Redis(클러스터 모드 비활성화됨) 클러스터인 경우에는 샤드 없이 Redis 클러스터에 복제를 추가하려면 (p. 92)의 단계들을 먼저 완료하십시오.

- 4. [Add node]를 선택합니다.
- 5. Add Node(노드 추가) 대화 상자에 요청을 받은 정보를 입력합니다.
- 6. 이 노드를 즉시 추가하려면 [Apply Immediately Yes] 버튼을 선택하고, 클러스터의 다음 유지 관리 기간 중에 이 노드를 추가하려면 [No]를 선택합니다.

신규 추가 및 제거 요청이 대기 중 요청에 미치는 영향

시나리오	대기 중 작업	신규 요청	Results
시나리오 1	삭제	삭제	신규 삭제 요청(대기 중 또는 즉시)은 대기 중 삭제 요청 을 대체합니다.
			예를 들어 노드 0001, 0003 및 0007에서 삭제 요청이 대기 중일 때 노드 0002 및 0004를 삭제하는 신규 요청

시나리오	대기 중 작업	신규 요청	Results
			이 생성될 경우 노드 0002 및 0004만 삭제됩니다. 노드 0001, 0003 및 0007은 삭제되지 않습니다.
시나리오 2	삭제	생성	신규 생성 요청(대기 중 또는 즉시)은 대기 중 삭제 요청 을 대체합니다.
			예를 들어 노드 0001, 0003 및 0007에서 삭제 요청이 대기 중일 때 노드를 생성하는 신규 요청이 생성될 경우 새노드가 생성되고 노드 0001, 0003 및 0007은 삭제되지 않습니다.
시나리오 3	생성	삭제	신규 삭제 요청(대기 중 또는 즉시)은 대기 중 생성 요청 을 대체합니다.
			예를 들어 노드 2개를 생성하는 요청이 대기 중일 때 노드 0003을 삭제하는 요청이 생성될 경우 새 노드는 생성되 지 않고 노드 0003이 삭제됩니다.
시나리오 4	생성	생성	신규 생성 요청은 대기 중 생성 요청에 추가됩니다.
			예를 들어 노드 2개를 생성하는 요청이 대기 중일 때 노드 3개를 생성하는 신규 요청이 생성될 경우 신규 요청이 대 기 중 요청에 추가되어 노드 5개가 생성됩니다.
			Important
			신규 생성 요청이 Apply Immediately - Yes(즉 시 적용 - 예)로 설정된 경우 모든 생성 요청 이 즉시 실행됩니다. 신규 생성 요청이 Apply Immediately - No(즉시 적용 - 아니요)로 설정된 경우 모든 생성 요청은 대기 중 작업입니다.

어떤 작업이 대기 중인지 알아보려면 설명 탭을 선택하여 대기 중 생성 또는 삭제 작업이 몇 개인지 확인 합니다. 대기 중 생성 작업과 대기 중 삭제 작업이 동시에 있을 수는 없습니다.



7. [Add] 버튼을 선택합니다.

잠시 후 [creating] 상태로 새로운 노드가 노드 목록에 나타납니다. 노드가 표시되지 않으면 브라우저 페이지를 새로 고치십시오. 노드가 사용 가능 상태가 되면 새로운 노드를 사용할 수 있습니다.

AWS CLI 사용

복제가 활성화되지 않은 기존 Redis(클러스터 모드 비활성화됨) 클러스터에 노드를 추가하려면 기존 클러스터를 기본 클러스터로 지정하여 먼저 복제 그룹을 만들어야 합니다. 자세한 내용은 사용 가능한 Redis 캐시클러스터(AWS CLI)를 사용하여 복제 그룹 생성 (p. 144) 단원을 참조하십시오. 복제 그룹이 [available] 상태가 되면 다음 프로세스로 계속 진행할 수 있습니다.

AWS CLI를 사용하여 클러스터에 노드를 추가하려면 다음 파라미터와 함께 AWS CLI 작업 modify-cache-cluster를 사용하십시오.

- --cache-cluster-id 노드를 추가할 캐시 클러스터의 ID입니다.
- --num-cache-nodes --num-cache-nodes 파라미터는 수정이 적용된 후 이 클러스터에 포함할 노드 수를 지정합니다. 이 클러스터에 노드를 추가하려면 --num-cache-nodes가 이 클러스터의 현재 노드 수보다 커야 합니다. 이 값이 현재 노드 수보다 작으면 ElastiCache가 파라미터 cache-node-ids-toremove 및 클러스터에서 제거할 노드 목록을 필요로 합니다. 자세한 내용은 AWS CLI 사용 (p. 97) 단 원을 참조하십시오.
- --apply-immediately 또는 --no-apply-immediately 이 노드를 즉시 추가할지 아니면 다음 번 유지 관리 기간에 추가할지 지정합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-cluster \
    --cache-cluster-id my-cluster \
    --num-cache-nodes 5 \
    --apply-immediately
```

Windows의 경우:

```
aws elasticache modify-cache-cluster ^
--cache-cluster-id my-cluster ^
--num-cache-nodes 5 ^
--apply-immediately
```

이 작업은 다음과 비슷한 출력을 생성합니다(JSON 형식).

```
"CacheCluster": {
    "Engine": "memcached",
    "CacheParameterGroup": {
        "CacheParameterGroupName": "default.memcached1.4",
        "ParameterApplyStatus": "in-sync"
},
    "CacheClusterId": "my-cluster",
    "PreferredAvailabilityZone": "us-west-2b",
    "ConfigurationEndpoint": {
        "Port": 11211,
        "Address": "rlh-mem000.7alc7bf-example.cfg.usw2.cache.amazonaws.com"
},
    "CacheSecurityGroups": [],
    "CacheClusterCreateTime": "2016-09-21T16:28:28.973Z",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterStatus": "modifying",
```

자세한 내용은 AWS CLI 항목 modify-cache-cluster를 참조하십시오.

ElastiCache API 사용

복제가 활성화되지 않은 기존 Redis(클러스터 모드 비활성화됨) 클러스터에 노드를 추가하려면 기존 클러스터를 기본 클러스터로 지정하여 먼저 복제 그룹을 만들어야 합니다. 자세한 내용은 독립형 Redis(클러스터모드 비활성화됨) 클러스터에 복제본 추가 (ElastiCache API) (p. 146) 단원을 참조하십시오. 복제 그룹이 [available] 상태가 되면 다음 프로세스로 계속 진행할 수 있습니다.

클러스터 (ElastiCache API)에 노드를 추가하려면

- 다음 파라미터와 함께 ModifyCacheCluster API 작업을 호출합니다.
 - CacheClusterId 노드를 추가할 클러스터의 ID입니다.
 - NumCacheNodes NumCachNodes 파라미터는 수정이 적용된 후 이 클러스터에 포함할 노드 수를 지 정합니다. 이 클러스터에 노드를 추가하려면 NumCacheNodes가 이 클러스터의 현재 노드 수보다 커 야 합니다. 이 값이 현재 노드 수보다 작으면 ElastiCache가 클러스터에서 제거할 노드 목록과 함께 파 라미터 CacheNodeIdsToRemove를 필요로 합니다(ElastiCache API 사용 (p. 100) 참조).
 - ApplyImmediately 이 노드를 즉시 추가할지 아니면 다음 번 유지 관리 기간에 추가할지 지정합니다.
 - Region 노드를 추가할 클러스터의 AWS 리전을 지정합니다.

다음 예제에서는 클러스터에 노드를 추가하기 위한 호출을 보여줍니다.

Example

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=ModifyCacheCluster
    &ApplyImmediately=true
    &NumCacheNodes=5
    &CacheClusterId=my-cluster
&Region=us-east-2
    &Version=2014-12-01
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20141201T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20141201T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Credential=<credential>
```

&X-Amz-Signature=<signature>

자세한 내용은 ElastiCache API 항목 ModifyCacheCluster를 참조하십시오.

클러스터에서 노드 제거

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 클러스터에서 노드를 삭제할 수 있습니다.

주제

- AWS Management 콘솔 사용 (p. 96)
- AWS CLI 사용 (p. 97)
- ElastiCache API 사용 (p. 100)

AWS Management 콘솔 사용

클러스터 (console)에서 노드를 제거하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 상단 오른쪽 모서리의 목록에서 노드를 제거할 클러스터의 AWS 리전을 선택합니다.
- 3. 노드를 제거하려는 클러스터에서 실행 중인 엔진을 탐색 창에서 선택합니다.

선택한 엔진을 실행하는 클러스터 목록이 표시됩니다.

4. 클러스터 목록에서 노드를 제거할 클러스터 이름을 선택합니다.

클러스터의 노드 목록이 나타납니다.

5. 제거할 노드의 노드 ID 왼쪽에 있는 상자를 선택합니다. ElastiCache 콘솔을 사용하면 노드를 한 번에 하나만 삭제할 수 있으므로 노드를 여러 개 선택하면 노드 삭제 버튼이 비활성화됩니다.

[Delete Node] 대화 상자가 나타납니다.

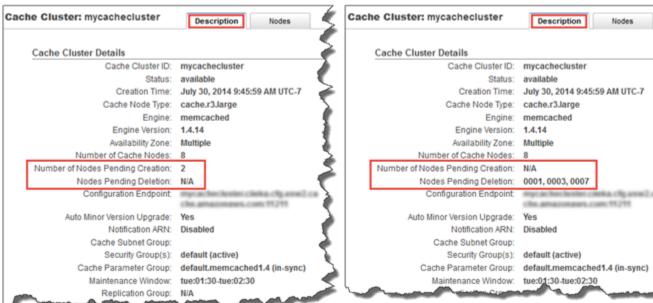
6. 노드를 삭제하려면 [Delete Node] 대화 상자를 완료하고 [Delete Node]를 선택합니다. 노드를 삭제하지 않으려면 [Cancel]을 선택합니다.

신규 추가 및 제거 요청이 대기 중 요청에 미치는 영향

시나리오	대기 중 작업	신규 요청	Results
시나리오 1	삭제	삭제	신규 삭제 요청(대기 중 또는 즉시)은 대기 중 삭제 요청을 대체합니다.
			예를 들어 노드 0001, 0003 및 0007에서 삭제 요청이 대기 중일 때 노드 0002 및 0004를 삭제하는 신규 요청이 생성될 경우 노드 0002 및 0004만 삭제됩니다. 노드 0001, 0003 및 0007은 삭제되지 않습니다.
시나리오 2	삭제	생성	신규 생성 요청(대기 중 또는 즉시)은 대기 중 삭제 요청을 대체합니다.
			예를 들어 노드 0001, 0003 및 0007에서 삭제 요청이 대기 중일 때 노드를 생성하는 신규 요청이 생성될 경우 새 노드 가 생성되고 노드 0001, 0003 및 0007은 삭제되지 않습니 다.

시나리오	대기 중 작업	신규 요청	Results
시나리오 3	생성	삭제	신규 삭제 요청(대기 중 또는 즉시)은 대기 중 생성 요청을 대체합니다.
			예를 들어 노드 2개를 생성하는 요청이 대기 중일 때 노드 0003을 삭제하는 요청이 생성될 경우 새 노드는 생성되지 않고 노드 0003이 삭제됩니다.
시나리오 4	생성	생성	신규 생성 요청은 대기 중 생성 요청에 추가됩니다.
			예를 들어 노드 2개를 생성하는 요청이 대기 중일 때 노드 3 개를 생성하는 신규 요청이 생성될 경우 신규 요청이 대기 중 요청에 추가되어 노드 5개가 생성됩니다.
			Important
			신규 생성 요청이 Apply Immediately - Yes(즉시 적용 - 예)로 설정된 경우 모든 생성 요청이 즉시 실행됩니다. 신규 생성 요청이 Apply Immediately - No(즉시 적용 - 아니요)로 설정된 경우 모든 생성요청은 대기 중 작업입니다.

어떤 작업이 대기 중인지 알아보려면 설명 탭을 선택하여 대기 중 생성 또는 삭제 작업이 몇 개인지 확인합니 다. 대기 중 생성 작업과 대기 중 삭제 작업이 동시에 있을 수는 없습니다.



AWS CLI 사용

- 1. 제거할 노드의 ID를 확인합니다. 자세한 내용은 클러스터 세부 정보 보기 (p. 78) 단원을 참조하십시오.
- 2. 다음 예제와 같이 제거할 노드 목록과 함께 modify-cache-cluster CLI 작업을 사용하십시오.

명령줄 인터페이스를 사용하여 클러스터에서 노드를 제거하려면 다음 파라미터와 함께 modify-cache-cluster 명령을 사용하십시오.

- --cache-cluster-id 노드를 제거할 캐시 클러스터의 ID입니다.
- --num-cache-nodes --num-cache-nodes 파라미터는 수정이 적용된 후 이 클러스터에 포함할 노드 수를 지정합니다.
- --cache-node-ids-to-remove 이 클러스터에서 제거할 노드 ID 목록입니다.
- --apply-immediately 또는 --no-apply-immediately 이 노드를 즉시 제거할지 아니면 다음 번 유지 관리 기간에 제거할지 지정합니다.
- --region 노드를 제거할 클러스터의 리전을 지정합니다.

다음 예제에서는 my-cluster 클러스터에서 노드 0001을 즉시 제거합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-cluster \
    --cache-cluster-id my-cluster \
    --num-cache-nodes 2 \
    --cache-node-ids-to-remove 0001 \
    --region us-east-2 \
    --apply-immediately
```

Windows의 경우:

이 작업은 다음과 비슷한 출력을 생성합니다(JSON 형식).

```
{
    "CacheClusters": [
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": false,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": false,
            "PreferredAvailabilityZone": "us-west-2a",
            "TransitEncryptionEnabled": false,
            "ReplicationGroupId": "my-cluster2",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
            "CacheClusterId": "my-cluster2-001",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
```

```
"ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
                }
            ٦,
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": false,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": false,
            "PreferredAvailabilityZone": "us-west-2b",
            "TransitEncryptionEnabled": false,
            "ReplicationGroupId": "my-cluster2",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
            "CacheClusterId": "my-cluster2-002",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            },
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
                }
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "AuthTokenEnabled": false,
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "12:30-13:30",
            "AutoMinorVersionUpgrade": true,
            "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
            "CacheClusterStatus": "available",
            "AtRestEncryptionEnabled": false,
            "PreferredAvailabilityZone": "us-west-2c",
            "TransitEncryptionEnabled": false,
            "ReplicationGroupId": "my-cluster2",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
            "CacheClusterId": "my-cluster2-003",
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
```

```
"CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            "SnapshotRetentionLimit": 0,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
        },
            "CacheClusterStatus": "available",
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
                }
            ٦,
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "Engine": "redis",
            "PreferredMaintenanceWindow": "wed:12:00-wed:13:00",
            "CacheSubnetGroupName": "default",
            "SnapshotWindow": "08:30-09:30",
            "TransitEncryptionEnabled": false,
            "AtRestEncryptionEnabled": false,
            "CacheClusterId": "my-cluster1",
            "CacheClusterCreateTime": "2018-02-26T21:06:43.420Z",
            "PreferredAvailabilityZone": "us-west-2c",
            "AuthTokenEnabled": false,
            "PendingModifiedValues": {},
            "CacheNodeType": "cache.r4.large",
            "CacheParameterGroup": {
                "CacheNodeIdsToReboot": [],
                "ParameterApplyStatus": "in-sync",
                "CacheParameterGroupName": "default.redis3.2"
            },
            "SnapshotRetentionLimit": 0,
            "AutoMinorVersionUpgrade": true,
            "EngineVersion": "3.2.10",
            "CacheSecurityGroups": [],
            "NumCacheNodes": 1
        }
    ]
}
```

자세한 내용은 AWS CLI 항목 describe-cache-cluster 및 modify-cache-cluster를 참조하십시오.

ElastiCache API 사용

ElastiCache API를 사용하여 노드를 제거하려면 다음과 같이 캐시 클러스터 ID 및 제거할 노드 목록과 함께 ModifyCacheCluster API 작업을 호출하십시오.

- CacheClusterId 노드를 제거할 캐시 클러스터의 ID입니다.
- NumCacheNodes NumCacheNodes 파라미터는 수정이 적용된 후 이 클러스터에 포함할 노드 수를 지정합니다.
- CacheNodeIdsToRemove.member.n 클러스터에서 제거할 노드 ID 목록입니다.
 - CacheNodeIdsToRemove.member.1=0004
 - CacheNodeIdsToRemove.member.1=0005
- ApplyImmediately 이 노드를 즉시 제거할지 아니면 다음 번 유지 관리 기간에 제거할지 지정합니다.

• Region 노드를 제거할 클러스터의 리전을 지정합니다.

다음 예제에서는 my-cluster 클러스터에서 노드 0004 및 0005를 즉시 제거합니다.

https://elasticache.us-west-2.amazonaws.com/ ?Action=ModifyCacheCluster &CacheClusterId=my-cluster &ApplyImmediately=true &CacheNodeIdsToRemove.member.1=0004 &CacheNodeIdsToRemove.member.2=0005 &NumCacheNodes=3 &Region us-east-2 &Version=2014-12-01 &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20141201T220302Z &X-Amz-Algorithm=AWS4-HMAC-SHA256 &X-Amz-Date=20141201T220302Z &X-Amz-SignedHeaders=Host &X-Amz-Expires=20141201T220302Z &X-Amz-Credential = < credential > &X-Amz-Signature=<signature>

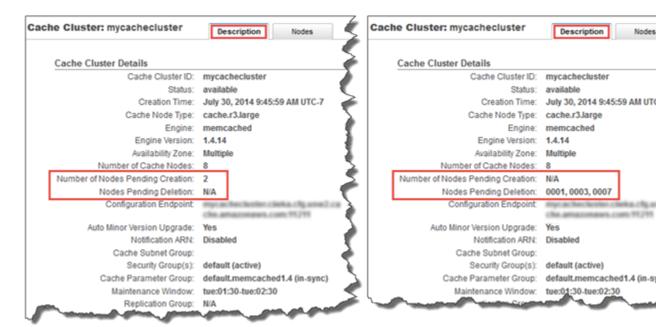
자세한 내용은 ElastiCache API 항목 ModifyCacheCluster를 참조하십시오.

대기 중인 노드 추가 또는 삭제 작업 취소

변경 사항을 즉시 적용하지 않도록 선택하면 다음 번 유지 관리 기간에 수행할 때까지 작업이 [pending] 상태 가 됩니다. 대기 중인 작업을 취소할 수 있습니다.

대기 중인 작업을 취소하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 상단 오른쪽 모서리의 목록에서 대기 중인 노드 추가 또는 삭제 작업을 취소할 AWS 리전을 선택합니다.
- 3. 취소하려는 대기 중 작업이 있는 클러스터에서 실행 중인 엔진을 탐색 창에서 선택합니다. 선택한 엔진을 실행하는 모든 클러스터의 목록이 표시됩니다.
- 4. 클러스터 목록에서 보류 중인 작업을 취소하고자 하는 클러스터의 이름 왼쪽에 있는 상자가 아닌 클러스 터의 이름을 선택합니다.
- 5. 어떤 작업이 대기 중인지 알아보려면 설명 탭을 선택하여 대기 중 생성 또는 삭제 작업이 몇 개인지 확인 합니다. 대기 중 생성 작업과 대기 중 삭제 작업이 동시에 있을 수는 없습니다.



- 6. [Nodes] 탭을 선택합니다.
- 7. 대기 중인 작업을 모두 취소하려면 [Cancel Pending]을 클릭합니다. [Cancel Pending] 대화 상자가 나타 납니다
- 8. [Cancel Pending] 버튼을 선택하여 대기 중인 작업을 모두 취소하도록 확인하거나 [Cancel]을 선택하여 작업을 유지합니다.

클러스터 삭제하기

클러스터가 available 상태면 클러스터를 적극 사용하고 있는지 여부에 관계없이 요금이 부과됩니다. 요금 발생을 중지하려면 클러스터를 삭제하십시오.

Warning

Redis용 ElastiCache 클러스터를 삭제하는 경우 수동 스냅샷은 보존됩니다. 클러스터를 삭제하기 전에 최종 스냅샷을 생성할 수 있는 옵션도 있습니다. 자동 캐시 스냅샷은 보존되지 않습니다.

AWS Management 콘솔 사용

다음은 배포에서 클러스터 하나를 삭제하는 절차입니다. 클러스터를 여러 개 삭제하려면 삭제할 클러스터마다 절차를 반복하십시오. 클러스터 하나를 다 삭제한 후 다른 클러스터 삭제 절차가 시작될 때까지 기다릴 필요는 없습니다.

클러스터를 삭제하려면

- 1. AWS Management 콘솔에 로그인한 후 https://console.aws.amazon.com/elasticache/에서 Amazon ElastiCache 콘솔을 엽니다.
- 2. ElastiCache 콘솔 대시보드에서 삭제하려는 클러스터가 실행 중인 엔진을 선택합니다.

선택한 엔진을 실행 중인 모든 클러스터의 목록이 표시됩니다.

삭제할 클러스터를 선택하려면 클러스터 목록에서 클러스터의 이름을 선택합니다.

Important

ElastiCache 콘솔에서는 클러스터를 한 번에 하나씩만 삭제할 수 있습니다. 여러 클러스터를 선택하면 삭제 작업이 비활성화됩니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 클러스터 삭제하기

- 4. [Actions] 버튼을 선택하고 작업 목록에서 [Delete]를 선택합니다.
- 5. 클러스터 삭제 확인 화면에서 삭제를 선택하여 클러스터를 삭제하거나 취소를 선택하여 클러스터를 유 지합니다.

[Delete]를 선택한 경우 클러스터 상태가 [deleting]으로 바뀝니다.

클러스터가 클러스터 목록에서 제거되는 즉시 요금 부과가 중단됩니다.

AWS CLI 사용

다음 코드는 캐시 클러스터 my-cluster를 삭제합니다.

```
aws elasticache delete-cache-cluster --cache-cluster-id my-cluster
```

delete-cache-cluster CLI 작업은 캐시 클러스터를 하나만 삭제합니다. 캐시 클러스터를 여러 개 삭제 하려면 삭제할 캐시 클러스터마다 delete-cache-cluster를 호출하십시오. 캐시 클러스터 하나를 다 삭 제한 후 다른 캐시 클러스터를 삭제할 때까지 기다릴 필요는 없습니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache delete-cache-cluster \
    --cache-cluster-id my-cluster \
    --region us-east-2
```

Windows의 경우:

```
aws elasticache delete-cache-cluster ^
--cache-cluster-id my-cluster ^
--region us-east-2
```

자세한 내용은 AWS CLI의 ElastiCache 항목 delete-cache-cluster를 참조하십시오.

ElastiCache API 사용

다음 코드는 클러스터 my-cluster를 삭제합니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=DeleteCacheCluster
    &CacheClusterId=my-cluster
    &Region us-east-2
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &XZ-Amz-Date=20150202T220302Z
    &X-Amz-Date=20150202T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20150202T220302Z
    &X-Amz-Credential=<credential>
    &X-Amz-Signature=<signature>
```

DeleteCacheCluster API 작업은 캐시 클러스터를 하나만 삭제합니다. 캐시 클러스터를 여러 개 삭제하 려면 삭제할 캐시 클러스터마다 DeleteCacheCluster를 호출하십시오. 캐시 클러스터 하나를 다 삭제한 후 다른 캐시 클러스터를 삭제할 때까지 기다릴 필요는 없습니다.

자세한 내용은 ElastiCache API 참조 항목 DeleteCacheCluster을 참조하십시오.

클러스터 또는 복제 그룹에 액세스

Amazon ElastiCache 인스턴스는 Amazon EC2 인스턴스를 통해서만 액세스하도록 설계되었습니다.

Amazon Virtual Private Cloud(Amazon VPC)에서 ElastiCache 인스턴스를 시작하면 동일한 Amazon VPC의 Amazon EC2 인스턴스에서 ElastiCache 인스턴스에 액세스할 수 있습니다. 또는 VPC 피어링을 사용하여 다른 Amazon VPC의 Amazon EC2에서 ElastiCache 인스턴스에 액세스할 수 있습니다.

EC2 Classic에서 ElastiCache 인스턴스를 시작하면, 인스턴스와 연결된 Amazon EC2 보안 그룹의 캐시 보안 그룹에 대한 액세스를 허용하여 EC2 인스턴스가 클러스터에 액세스하는 것을 허용합니다. 기본적으로 클러스터에 대한 액세스는 클러스터를 시작한 계정으로 제한됩니다.

주제

- 클러스터의 플랫폼 확인 (p. 104)
- 클러스터 또는 복제 그룹에 액세스 권한 부여 (p. 106)

클러스터의 플랫폼 확인

계속하기 전에 EC2-VPC로 시작했는지, EC2-Classic으로 시작했는지 확인합니다.

자세한 정보는 Detecting Your Supported Platforms and Whether You Have a Default VPC를 참조하십시오.

ElastiCache 콘솔을 사용하여 클러스터 플랫폼 확인

다음 절차에서는 ElastiCache 콘솔을 사용하여 클러스터를 EC2-VPC로 시작했는지, EC2-Classic으로 시작했는지 확인합니다.

ElastiCache 콘솔을 사용하여 클러스터 플랫폼을 확인하려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. Redis 엔진을 실행 중인 클러스터의 목록을 보려면 왼쪽 탐색 창에서 Redis를 선택합니다.
- 클러스터 목록에서 클러스터 이름 왼쪽의 상자를 선택하여 액세스 권한을 부여할 클러스터를 확장합니다.
- 4. [Subnet group:]을 찾습니다.



- [Subnet group]에 이름이 있는 경우에는 여기 나와 있듯이 EC2-VPC에서 클러스터를 시작한 것이므로 클러스터를 EC2-VPC로 시작한 경우 (p. 106)의 지침을 따라야 합니다.
- [Subnet group] 이름 대신 대시(-)가 있는 경우에는 EC2-Classic에서 클러스터를 시작한 것이므로 EC2-Classic에서 실행 중인 클러스터를 시작한 경우 (p. 106)의 지침을 따라야 합니다.

자세한 정보는 Detecting Your Supported Platforms and Whether You Have a Default VPC를 참조하십시오.

AWS CLI를 사용하여 클러스터 플랫폼 확인

다음 절차에서는 AWS CLI을 사용하여 클러스터를 EC2-VPC로 시작했는지, EC2-Classic으로 시작했는지 확인합니다.

AWS CLI를 사용하여 클러스터 플랫폼을 확인하려면

- 1. 명령 창을 엽니다.
- 2. 명령 프롬프트에서 다음 명령을 실행합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-cache-clusters \
--show-cache-cluster-details \
--cache-cluster-id my-cluster
```

Windows의 경우:

```
aws elasticache describe-cache-clusters ^
--show-cache-cluster-details ^
--cache-cluster-id my-cluster
```

이 명령의 JSON 출력은 다음과 같습니다. 공간 절약을 위해 일부 출력이 생략됩니다.

```
"CacheClusters": [
    {
        "Engine": "redis",
        "AuthTokenEnabled": false,
        "CacheParameterGroup": {
            "CacheNodeIdsToReboot": [],
            "CacheParameterGroupName": "default.redis3.2",
            "ParameterApplyStatus": "in-sync"
        "CacheClusterId": "my-cluster-001",
        "CacheSecurityGroups": [],
        "NumCacheNodes": 1,
        "AtRestEncryptionEnabled": false,
        "CacheClusterCreateTime": "2018-01-16T20:09:34.449Z",
        "ReplicationGroupId": "my-cluster",
        "AutoMinorVersionUpgrade": true,
        "CacheClusterStatus": "available",
        "PreferredAvailabilityZone": "us-east-2a",
        "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "SecurityGroups": [
                "Status": "active",
                "SecurityGroupId": "sg-e8c03081"
            }
        ٦,
        "TransitEncryptionEnabled": false,
        "CacheSubnetGroupName": "default",
        "EngineVersion": "3.2.10",
        "PendingModifiedValues": {},
        "PreferredMaintenanceWindow": "sat:05:30-sat:06:30",
        "CacheNodeType": "cache.t2.medium"
    }
]
```

}

- CacheSubnetGroupName 값이 있는 경우에는 EC2-VPC에서 클러스터를 시작한 것이므로 클러스터를 EC2-VPC로 시작한 경우 (p. 106)의 지침을 따라야 합니다.
- CacheSubnetGroupName 값이 없는 경우에는 EC2-Classic에서 클러스터를 시작한 것이므로 EC2-Classic에서 실행 중인 클러스터를 시작한 경우 (p. 106)의 지침을 따라야 합니다.

클러스터 또는 복제 그룹에 액세스 권한 부여

클러스터를 EC2-VPC로 시작한 경우

Amazon Virtual Private Cloud(Amazon VPC)로 클러스터를 시작한 경우 동일한 Amazon VPC에서 실행 중인 Amazon EC2 인스턴스에서만 ElastiCache 클러스터에 연결할 수 있습니다. 이 경우 클러스터에 네트워크 진입을 허용해야 합니다.

Amazon VPC 보안 그룹에서 클러스터로의 네트워크 진입을 허용하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/ec2/에서 Amazon EC2 콘솔을 엽니다.
- 2. 탐색 창의 [Network & Security] 아래에서 [Security Groups]를 선택합니다.
- 3. 보안 그룹 목록에서 Amazon VPC를 위한 보안 그룹을 선택합니다. ElastiCache 사용을 위한 보안 그룹을 생성하지 않는 한 이 보안 그룹의 이름은 default로 지정됩니다.
- 4. [Inbound] 탭을 선택하고 다음을 수행합니다.
 - a. [Edit]를 선택합니다.
 - b. [Add another rule]을 선택합니다.
 - c. [Type] 열에서 [Custom TCP rule]을 선택합니다.
 - d. [Port range] 상자에 클러스터 노드의 포트 번호를 입력합니다. 이 번호는 클러스터를 시작할 때 지정한 번호와 동일해야 합니다. 이며 Redis의 기본 포트는 **6379**입니다.
 - e. 소스 상자에서 포트 범위(0.0.0.0/0)를 가진 위치 무관을 선택하면 Amazon VPC 내에서 시작한 Amazon EC2 인스턴스를 ElastiCache 노드에 연결할 수 있습니다.

Important

ElastiCache 클러스터를 0.0.0.0/0(4.e.단계)으로 열면 공용 IP 주소가 없기 때문에 클러스터가 인터넷에 노출되지 않으므로 VPC 외부에서 액세스할 수 없습니다. 그러나 기본 보안 그룹이 고객 계정의 다른 Amazon EC2 인스턴스에 적용될 수 있으며 이러한 인스턴스는 공용 IP 주소를 가질 수 있습니다. 포트 6379에서 무언가를 실행하면 비의도적으로 해당서비스가 노출될 수 있습니다. 따라서 ElastiCache가 독점적으로 사용하는 VPC 보안 그룹을 생성하는 것이 좋습니다. 자세한 정보는 사용자 지정 보안 그룹을 참조하십시오.

f. Save를 선택합니다.

Amazon EC2 인스턴스를 Amazon VPC로 시작하면 해당 인스턴스를 ElastiCache 클러스터에 연결할 수 있습니다.

EC2-Classic에서 실행 중인 클러스터를 시작한 경우

클러스터를 EC2-Classic으로 시작한 경우 Amazon EC2인스턴스가 클러스터에 액세스할 수 있게 하려면 인 스턴스와 연결 된 Amazon EC2 보안 그룹을 허용하여 캐시 보안 그룹에 액세스합니다.

클러스터에 Amazon EC2 보안 그룹의 액세스를 허용하려면

1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.

2. 보안 그룹 목록을 표시하려면 왼쪽 탐색 창의 [Security Groups]를 선택합니다.

Important

[Security Groups]이 탐색 창에 표시되지 않은 경우에는 EC2-Classic이 아니라 EC2-VPC에서 클러스터를 시작한 것이므로 클러스터를 EC2-VPC로 시작한 경우 (p. 106)의 지침을 따라야 합니다.

- 3. [default] 보안 그룹의 왼쪽에 있는 확인란을 선택합니다.
- 4. 화면 아래쪽에 있는 목록에서 승인하려는 [EC2 Security Group Name]을 선택합니다.
- 5. 액세스를 허가하려면 [Add]를 선택합니다.

보안 그룹과 연결된 Amazon EC2 인스턴스를 ElastiCache 클러스터에 연결하도록 승인했습니다.

보안 그룹의 액세스를 취소하려면 승인한 보안 그룹 목록에서 보안 그룹을 찾은 다음 [Remove]를 선택합니다.

ElastiCache 보안 그룹에 대한 자세한 내용은 보안 그룹: EC2-Classic (p. 334) 단원을 참조하십시오.

AWS 외부에서 ElastiCache 리소스 액세스

Elastiache는 VPC에 내부적으로 사용하도록 설계된 서비스입니다. 인터넷 트래픽의 지연 시간 및 보안 문제로 인해 외부 액세스는 권장되지 않습니다. 그러나 테스트 또는 개발 목적으로 Elastiache에 대한 외부 액세스가 필요한 경우 VPN을 통해 수행할 수 있습니다.

AWS 클라이언트 VPN을 사용하면 다음과 같은 이점을 통해 Elastiache 노드에 대한 외부 액세스를 허용합니다.

- 승인된 사용자 또는 인증 키에 대한 제한된 액세스
- VPN 클라이언트와 AWS VPN 엔드포인트 간의 암호화된 트래픽
- 특정 서브넷 또는 노드에 대한 제한된 액세스
- 사용자 또는 인증 키로부터의 액세스를 쉽게 취소
- 감사 연결

다음 절차에서는 다음 작업을 수행하는 방법을 보여줍니다.

주제

- 인증 기관 생성 (p. 107)
- AWS Client VPN 구성 요소 구성 (p. 108)
- VPN 클라이언트 구성 (p. 110)

인증 기관 생성

다양한 기술이나 도구를 사용하여 CA(인증 기관)를 생성할 수 있습니다. OpenVPN 프로젝트에서 제공하는 easy-rsa 유틸리티를 사용하는 것이 좋습니다. 선택한 옵션에 관계없이 키를 안전하게 유지해야 합니다. 다음 절차에서는 easy-rsa 스크립트를 다운로드하고 인증 기관과 첫 번째 VPN 클라이언트를 인증하는 키를 생성합니다.

- 초기 인증서를 생성하려면 터미널을 열고 다음 작업을 수행하십시오.
 - git clone https://github.com/OpenVPN/easy-rs
 - cd easy-rsa
 - ./easyrsa3/easyrsa init-pki
 - ./easyrsa3/easyrsa build-ca nopass
 - ./easyrsa3/easyrsa build-server-full server nopass

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 클러스터 또는 복제 그룹에 액세스 권한 부여

· ./easyrsa3/easyrsa build-client-full client1.domain.tld nopass

인증서를 포함하는 pki 하위 디렉터리는 easy-rsa 아래에 생성됩니다.

- 다음과 같이 AWS Certificate Manager(ACM)에 서버 인증서를 제출합니다.
 - ACM 콘솔에서 인증서 관리자를 선택합니다.
 - 인증서 가져오기를 선택합니다.
 - easy-rsa/pki/issued/server.crt 파일에서 사용할 수 있는 퍼블릭 키 인증서를 인증서 본문 필드에 입력합니다.
 - easy-rsa/pki/private/server.key에서 사용할 수 있는 프라이빗 키를 인증서 프라이빗 키 필드에 붙여 넣습니다. BEGIN AND END PRIVATE KEY 사이의 모든 선(BEGIN 및 END 선 포함)을 선택해야 합니다
 - easy-rsa/pki/ca.crt 파일에서 사용할 수 있는 CA 퍼블릭 키를 인증서 체인 필드에 붙여넣습니다.
 - 검토 및 가져오기를 선택합니다.
 - 가져오기를 선택합니다.

AWS CLI를 사용하여 서버의 인증서를 ACM에 제출하려면 다음 명령을 실행하십시오. aws acm import-certificate --certificate file://easy-rsa/pki/issued/server.crt --private-key file://easy-rsa/pki/private/server.key --certificate-chain file://easy-rsa/pki/ca.crt --region region

나중에 사용할 수 있도록 인증서 ARN을 기록해 둡니다.

AWS Client VPN 구성 요소 구성

AWS 콘솔 사용

AWS 콘솔에서 서비스, VPC 순으로 선택합니다.

Virtual Private Network(가상 프라이빗 네트워크)에서 Client VPN Endpoints(클라이언트 VPN 엔드포인트)를 선택하고 다음을 수행합니다.

AWS Client VPN 구성 요소 구성

- Create Client VPN Endpoint(클라이언트 VPN 엔드포인트 생성)를 선택합니다.
- 다음과 같은 옵션을 지정할 수 있습니다.
 - Client IPv4 CIDR(클라이언트 IPv4 CIDR): 넷마스크가 최소 /22 범위에 있는 프라이빗 네트워크를 사용합니다. 선택한 서브넷이 VPC 네트워크의 주소와 충돌하지 않는지 확인합니다. 예: 10.0.0.0/22.
 - Server certificate ARN(서버 인증서 ARN)에서 앞서 가져온 인증서의 ARN을 선택합니다.
 - Use mutual authentication(상호 인증 사용)을 선택합니다.
 - Client certificate ARN(클라이언트 인증서 ARN)에서 앞서 가져온 인증서의 ARN을 선택합니다.
 - Create Client VPN Endpoint(클라이언트 VPN 엔드포인트 생성)를 선택합니다.

AWS CLI 사용

다음 명령을 실행합니다.

aws ec2 create-client-vpn-endpoint --client-cidr-block "10.0.0.0/22" --server-certificate-arn arn:aws:acm:us-east-1:012345678912:certificate/0123abcd-ab12-01a0-123a-123456abcdef --authentication-options Type=certificate-authentication,,MutualAuthentication={ClientRootCertificateChainArn=arn:aws:acm:us-east-1:012345678912:certificate/123abcd-ab12-01a0-123a-123456abcdef} --connection-log-options Enabled=false

출력 예:

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 클러스터 또는 복제 그룹에 액세스 권한 부여

"ClientVpnEndpointId": "cvpn-endpoint-0123456789abcdefg",
"Status": { "Code": "pending-associate" }, "DnsName": "cvpnendpoint-0123456789abcdefg.prod.clientvpn.us-east-1.amazonaws.com" }

대상 네트워크를 VPN 엔드포인트에 연결

- 새 VPN 엔드포인트를 선택한 다음 연결 탭을 선택합니다.
- 연결을 선택하고 다음 옵션을 지정합니다.
 - VPC: Elasticache 클러스터의 VPC를 선택합니다.
 - Elastiache 클러스터의 네트워크 중 하나를 선택합니다. 의심스러운 경우 Elastiache 대시보드의 서브넷 그룹에 있는 네트워크를 검토하십시오.
 - 연결을 선택합니다. 필요한 경우 나머지 네트워크에 대해 이 단계를 반복합니다.

AWS CLI 사용

다음 명령을 실행합니다.

aws ec2 associate-client-vpn-target-network --client-vpn-endpoint-id cvpn-endpoint-0123456789abcdefg --subnet-id subnet-0123456789abdcdef

출력 예:

```
"Status": { "Code": "associating" }, "AssociationId": "cvpn-assoc-0123456789abdcdef" }
```

VPN 보안 그룹 검토

VPN 엔드포인트는 VPC의 기본 보안 그룹을 자동으로 채택합니다. 인바운드 및 아웃바운드 규칙을 점검하고 보안 그룹이 VPN 네트워크(VPN 엔드포인트 설정에 정의됨)에서 서비스 포트의 Elastiache 네트워크로의 트래픽을 허용하는지 확인합니다(기본적으로 Redis의 경우 6379, Memcached의 경우 11211).

VPN 엔드포인트에 할당된 보안 그룹을 변경해야 하는 경우 다음과 같이 진행합니다.

- 현재 보안 그룹을 선택합니다.
- Apply Security Group(보안 그룹 적용)을 선택합니다.
- 보안 그룹을 선택합니다.

AWS CLI 사용

다음 명령을 실행합니다.

aws ec2 apply-security-groups-to-client-vpn-target-network --client-vpn-endpoint-id cvpn-endpoint-0123456789abcdefga --vpc-id vpc-0123456789abdcdef --security-group-ids sg-0123456789abdcdef

출력 예:

```
"SecurityGroupIds": [ "sg-0123456789abdcdef" ] }
```

Note

ElastiCache 보안 그룹은 VPN 클라이언트에서 오는 트래픽을 허용해야 합니다. 클라이언트의 주소는 VPC 네트워크에 따라 VPN 엔드포인트 주소로 마스킹 처리됩니다. 따라서 Elastiache 보안 그룹에서 인바운드 규칙을 생성할 때 VPC 네트워크(VPN 클라이언트 네트워크가 아님)를 고려하십시으

대상 네트워크에 대한 VPN 액세스 승인

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 클러스터 또는 복제 그룹에 액세스 권한 부여

Authorization(권한 부여) 탭에서 Authorize Ingress(권한 부여 승인)를 선택하고 다음과 같이 지정합니다.

- 액세스를 활성화할 대상 네트워크: 0.0.0.0/0을 사용하여 모든 네트워크(인터넷 포함)에 대한 액세스를 허용하거나 Elastiache 네트워크/호스트를 제한합니다.
- Grant access to:(다음에 대한 액세스 권한 부여:)에서 Allow access to all users(모든 사용자에게 액세스 허용)를 선택합니다.
- Add Authorization Rules(권한 부여 규칙 추가)를 선택합니다.

AWS CLI 사용

다음 명령을 실행합니다.

aws ec2 authorize-client-vpn-ingress --client-vpn-endpoint-id cvpn-endpoint-0123456789abcdefg --target-network-cidr 0.0.0.0/0 --authorize-all-groups

출력 예:

```
{ "Status": { "Code": "authorizing" } }
```

VPN 클라이언트에서 인터넷에 액세스하도록 허용

VPN을 통해 인터넷을 검색해야 하는 경우 추가 경로를 만들어야 합니다. 라우팅 테이블 탭을 선택한 다음 Create Route(라우팅 생성)를 선택합니다.

- 라우팅 대상 주소: 0.0.0.0/0
- Target VPC Subnet ID(대상 VPC 서브넷 ID): 인터넷 액세스 권한이 있는 연결된 서브넷 중 하나를 선택합니다.
- Create Route(라우팅 생성)를 선택합니다.

AWS CLI 사용

다음 명령을 실행합니다.

aws ec2 create-client-vpn-route --client-vpn-endpoint-id cvpn-endpoint-0123456789abcdefg --destination-cidr-block 0.0.0.0/0 --target-vpc-subnet-id subnet-0123456789abdcdef

출력 예:

```
{ "Status": { "Code": "creating" } }
```

VPN 클라이언트 구성

AWS Client VPN 대시보드에서 최근에 생성된 VPN 엔드포인트를 선택하고 Download Client Configuration(클라이언트 구성 다운로드)를 선택합니다. 구성 파일과 easy-rsa/pki/private/client1.domain.tld.key 및 easy-rsa/pki/issued/client1.domain.tld.crt 파일을 복사합니다. 구성 파일을 편집하고 다음 파라미터를 변경하거나 추가합니다.

- cert: client1.domain.tld.crt 파일을 가리키는 파라미터 인증서가 있는 새 줄을 추가합니다. 파일의 전체 경로를 사용합니다. 예: cert /home/user/.cert/client1.domain.tld.crt
- cert: key: client1.domain.tld.key 파일을 가리키는 파라미터 키가 있는 새 줄을 추가합니다. 파일의 전체 경로를 사용합니다. 예: key /home/user/.cert/client1.domain.tld.key

다음 명령을 사용하여 VPN 연결을 설정합니다. sudo openvpn --config downloaded-client-config.ovpn

액세스 취소

특정 클라이언트 키의 액세스를 무효화해야 하는 경우 CA에서 키를 취소해야 합니다. 그런 다음 취소 목록을 AWS Client VPN에 제출합니다.

easy-rsa로 키 취소:

- cd easy-rsa
- ./easyrsa3/easyrsa revoke client1.domain.tld
- 계속하려면 "예"를 입력하고 중단하려면 다른 값을 입력합니다.

Continue with revocation: `yes` ... * `./easyrsa3/easyrsa gen-crl

• 업데이트된 CRL이 생성되었습니다. CRL 파일: /home/user/easy-rsa/pki/crl.pem

취소 목록을 AWS Client VPN으로 가져오기:

- AWS Management 콘솔에서 서비스, VPC 순으로 선택합니다.
- Client VPN Endpoints(클라이언트 VPN 엔드포인트)를 선택합니다.
- 클라이언트 VPN 엔드포인트를 선택한 다음 작업 -> Import Client Certificate CRL(클라이언트 인증서 CRL 가져오기)을 선택합니다.
- crl.pem 파일의 내용을 붙여넣습니다.

AWS CLI 사용

다음 명령을 실행합니다.

aws ec2 import-client-vpn-client-certificate-revocation-list --certificate-revocation-list file://./easy-rsa/pki/crl.pem --client-vpn-endpoint-id cvpn-endpoint-0123456789abcdefg

출력 예:

```
Example output: { "Return": true }
```

샤드 작업

샤드(API/CLI: 노드 그룹)는 1~six개의 Redis 노드 모음입니다. Redis(클러스터 모드 비활성화됨) 클러스터는 둘 이상의 샤드를 가질 수 없습니다. Redis(클러스터 모드 활성화됨) 클러스터에는 1~90개의 샤드가 있을 수 있습니다. 하나의 클러스터당 최대 90개의 노드로 구성된 더 많은 수의 샤드와 더 적은 수의 복제본을 가진 클러스터를 생성할 수 있습니다. 이 클러스터 구성은 90개의 샤드 및 0개의 복제본부터 15개의 샤드 및 5개의 복제본까지 해당될 수 있으며, 이는 허용되는 최대 복제본 수입니다. 클러스터의 데이터는 클러스터의샤드로 분할됩니다. 샤드에 둘 이상의 노드가 있는 경우 샤드는 한 노드가 읽기/쓰기 기본 노드가 되고 다른노드가 읽기 전용 복제본 노드인 복제를 구현합니다.

Note

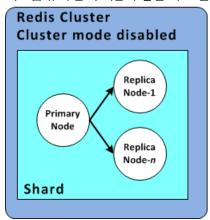
노드 또는 샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 한도 증가를 요청하는 방법에 대한 지침은 AWS 서비스 제한을 참조하고 한도 유형을 '인스턴스 유형별 클러스터당 노드'로 선택하십시오.

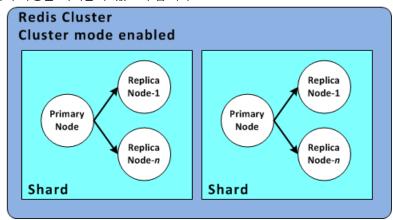
ElastiCache 콘솔을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터를 생성할 때 클러스터의 샤드수와 샤드의 노드 수를 지정합니다. 자세한 내용은 Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 72) 단원을 참조하십시오. ElastiCache API 또는 AWS CLI를 사용하여 클러스터(API/CLI에서는 복

제 그룹이라고 함)를 생성하는 경우 샤드(API/CLI: 노드 그룹)의 노드 수를 독립적으로 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.

- · API: CreateReplicationGroup
- CLI: create-replication-group

샤드의 각 노드는 컴퓨팅, 스토리지 및 메모리 사양이 동일합니다. ElastiCache API는 노드 수, 보안 설정 및 시스템 유지 관리 기간과 같은 샤드 전체의 속성을 제어할 수 있도록 합니다.





Redis 샤드 구성

샤드 ID를 검색하는 방법

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 샤드 ID를 검색할 수 있습니다.

주제

- AWS Management 콘솔 사용 (p. 112)
- AWS CLI 사용 (p. 113)
- ElastiCache API 사용 (p. 114)

AWS Management 콘솔 사용

주제

- Redis(클러스터 모드 비활성화됨)의 경우 (p. 112)
- Redis(클러스터 모드 활성화됨)의 경우 (p. 112)

Redis(클러스터 모드 비활성화됨)의 경우

Redis(클러스터 모드 비활성화됨) 복제 그룹의 샤드 ID는 항상 0001입니다.

Redis(클러스터 모드 활성화됨)의 경우

다음 절차에서는 AWS Management 콘솔을 사용하여 Redis(클러스터 모드 활성화됨) 복제 그룹의 샤드 ID 를 검색합니다.

Redis(클러스터 모드 활성화됨) 복제 그룹에서 샤드 ID를 찾으려면 다음을 수행합니다.

1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 샤드 ID를 검색하는 방법

- 2. 탐색 창에서 Redis를 선택한 후 샤드 ID를 검색할 Redis(클러스터 모드 활성화됨) 복제 그룹의 이름을 선택합니다.
- 3. 샤드 이름 열에서 샤드 ID는 샤드 이름의 마지막 네 자리 숫자입니다.

AWS CLI 사용

Redis(클러스터 모드 비활성화됨) 또는 Redis(클러스터 모드 활성화됨) 복제 그룹의 샤드(노드 그룹) ID를 검색하려면 다음 선택적 파라미터와 함께 AWS CLI 작업 describe-replication-groups를 사용합니다.

• --replication-group-id—사용되면 지정된 복제 그룹의 세부 정보 출력을 제한하는 선택적 파라미터입니다. 이 파라미터가 생략되면 최대 100개의 복제 그룹의 세부 정보가 반환됩니다.

Example

이 명령은 sample-repl-group의 세부 정보를 반환합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-replication-groups \
--replication-group-id sample-repl-group
```

Windows의 경우:

```
aws elasticache describe-replication-groups ^
--replication-group-id sample-repl-group
```

이 명령의 출력은 다음과 같습니다. 샤드(노드 그룹) ID는 더 쉽게 찾을 수 있도록 여기에 ## ##됩니다.

```
{
    "ReplicationGroups": [
       {
            "Status": "available",
            "Description": "2 shards, 2 nodes (1 + 1 replica)",
            "NodeGroups": [
                    "Status": "available",
                    "Slots": "0-8191",
                    "NodeGroupId": "0001",
                    "NodeGroupMembers": [
                        {
                             "PreferredAvailabilityZone": "us-west-2c",
                             "CacheNodeId": "0001",
                             "CacheClusterId": "sample-repl-group-0001-001"
                            "PreferredAvailabilityZone": "us-west-2a",
                            "CacheNodeId": "0001",
                             "CacheClusterId": "sample-repl-group-0001-002"
                    ]
                },
                    "Status": "available",
                    "Slots": "8192-16383",
                    "NodeGroupId": "0002",
                    "NodeGroupMembers": [
                             "PreferredAvailabilityZone": "us-west-2b",
                             "CacheNodeId": "0001",
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 글로벌 데이터 스토어를 사용한 AWS 리전 간 복제

```
"CacheClusterId": "sample-repl-group-0002-001"
                        },
                        {
                             "PreferredAvailabilityZone": "us-west-2a",
                             "CacheNodeId": "0001",
                             "CacheClusterId": "sample-repl-group-0002-002"
                        }
                    1
                }
            ٦,
            "ConfigurationEndpoint": {
                "Port": 6379.
                "Address": "sample-repl-group.9dcv5r.clustercfg.usw2.cache.amazonaws.com"
            "ClusterEnabled": true,
            "ReplicationGroupId": "sample-repl-group",
            "SnapshotRetentionLimit": 1,
            "AutomaticFailover": "enabled",
            "SnapshotWindow": "13:00-14:00",
            "MemberClusters": [
                "sample-repl-group-0001-001",
                "sample-repl-group-0001-002",
                "sample-repl-group-0002-001",
                "sample-repl-group-0002-002"
            "CacheNodeType": "cache.m3.medium",
            "PendingModifiedValues": {}
        }
    ]
}
```

ElastiCache API 사용

Redis(클러스터 모드 비활성화됨) 또는 Redis(클러스터 모드 활성화됨) 복제 그룹의 샤드(노드 그룹) ID를 검색하려면 다음 선택적 파라미터와 함께 AWS CLI 작업 describe-replication-groups를 사용합니다.

• ReplicationGroupId—사용되면 지정된 복제 그룹의 세부 정보 출력을 제한하는 선택적 파라미터입니다. 이 파라미터가 생략되면 최대 xxx개의 복제 그룹의 세부 정보가 반환됩니다.

Example

이 명령은 sample-repl-group의 세부 정보를 반환합니다.

Linux, macOS, or Unix의 경우:

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReplicationGroup
&ReplicationGroupId=sample-repl-group
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

글로벌 데이터 스토어를 사용한 AWS 리전 간 복제

Redisdyd 글로벌 데이터 스토어 기능을 사용하면 AWS 리전 간에 빠르고 안정적이며 안전한 완전 관리형 복제를 수행할 수 있습니다. 이 기능을 사용하면 Redis용 ElastiCache에 대한 리전 간 읽기 전용 복제본 클러스터를 생성하여 AWS 리전 간에 지연 시간이 짧은 읽기 및 재해 복구를 수행할 수 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 개요

다음 섹션에서는 글로벌 데이터 스토어로 작업하는 방법에 대한 설명을 찾을 수 있습니다.

주제

- 개요 (p. 115)
- 사전 조건 및 제한 사항 (p. 116)
- 글로벌 데이터 스토리지 사용(콘솔) (p. 117)
- 글로벌 데이터 저장소(CLI) 사용 (p. 125)

개요

각 글로벌 데이터 스토어는 서로 복제하는 하나 이상의 클러스터 모음입니다.

글로벌 데이터 스토어는 다음과 같이 구성됩니다.

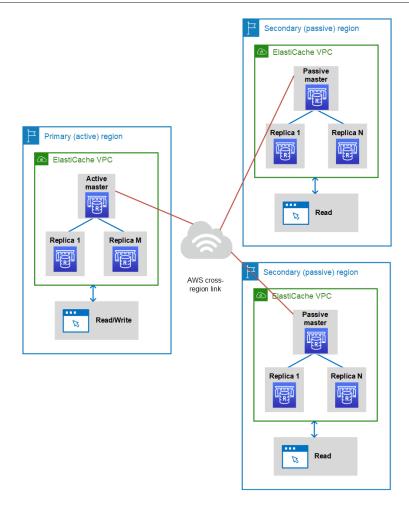
- 기본(활성) 클러스터 기본 클러스터는 글로벌 데이터 스토어 내의 모든 클러스터에 복제되는 쓰기를 허용합니다. 기본 클러스터는 읽기 요청도 허용합니다.
- 보조(수동) 클러스터 보조 클러스터는 읽기 요청만 허용하고 기본 클러스터에서 데이터 업데이트를 복제합니다. 보조 클러스터는 기본 클러스터와 다른 AWS 리전에 있어야 합니다.

ElastiCache에서 글로벌 데이터 스토어를 생성하면 Redis용 ElastiCache에서 기본 클러스터에서 보조 클러스터로 데이터를 자동으로 복제합니다. Redis 데이터를 복제할 AWS 리전을 선택한 다음 해당 AWS 리전에 보조 클러스터를 생성합니다. 그런 다음 ElastiCache는 두 클러스터 간의 자동 비동기식 데이터 복제를 설정하고 관리합니다.

Redis용 글로벌 데이터 스토어를 사용하면 다음과 같은 이점을 얻을 수 있습니다.

- 지리적 성능 추가 AWS 리전에 원격 복제본 클러스터를 설정하고 이들 간에 데이터를 동기화하면 해당 AWS 리전에서 데이터 액세스 지연 시간을 줄일 수 있습니다. 글로벌 데이터 스토어는 AWS 리전 간에 지연 시간이 짧은 지리적 읽기를 제공하여 애플리케이션의 응답성을 높일 수 있습니다.
- 재해 복구 글로벌 데이터 스토어의 기본 클러스터에서 성능 저하가 발생하는 경우 보조 클러스터를 새 기본 클러스터로 승격할 수 있습니다. 보조 클러스터가 포함된 AWS 리전에 연결하여 이를 수행할 수 있습 니다.

다음 다이어그램은 글로벌 데이터 스토어가 작동하는 방식을 보여줍니다.



사전 조건 및 제한 사항

글로벌 데이터 스토어를 시작하기 전에 다음 사항에 유의하십시오.

- 글로벌 데이터 스토어는 ap-northeast-2;, ap-southeast-1, ap-southeast-2, ap-northeast-1, eu-central-1, eu-west-2, eu-west-1, us-east-1, us-west-1, us-west-2 등의 AWS 리전에서 지원됩니다.
- 글로벌 데이터 스토어를 사용하려면 Redis 엔진 버전 5.0.6 이상과 R5 또는 M5 노드 유형을 사용합니다.
- 글로벌 데이터 스토어의 모든 —기본 및 보조— 클러스터에는 동일한 수의 마스터 노드, 노드 유형, 엔진 버전 및 샤드 수(클러스터 모드가 활성화된 경우)가 있어야 합니다. 글로벌 데이터 스토어의 각 클러스터에 는 해당 클러스터에 대한 로컬 읽기 트래픽을 수용하기 위해 다른 수의 읽기 복제본이 있을 수 있습니다.

기존 단일 노드 클러스터를 사용하려는 경우 복제를 활성화해야 합니다.

- 한 AWS 리전의 기본 클러스터에서 최대 두 개의 다른 AWS 리전의 보조 클러스터로 복제를 설정할 수 있습니다.
- VPC 클러스터에서만 글로벌 데이터 스토어로 작업할 수 있습니다. 자세한 내용은 Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴 (p. 320) 단원을 참조하십시오. EC2-Classic을 사용하는 경우 글로벌 데이터 스토어가 지원되지 않습니다.
- ElastiCache는 한 AWS 리전에서 다른 AWS 리전으로의 자동 장애 조치를 지원하지 않습니다. 필요한 경우 보조 클러스터를 수동으로 승격할 수 있습니다. 문제 해결 예는 보조 클러스터를 기본 클러스터로 승격 (p. 123)을(를) 참조하십시오.

- 기존 데이터에서 부트스트랩하려면 기존 클러스터를 기본 클러스터로 사용하여 글로벌 데이터 스토어를 생성합니다. 기존 클러스터를 보조 클러스터로 추가하는 것은 지원하지 않습니다. 클러스터를 보조 클러스 터로 추가하는 프로세스로 데이터가 지워져 데이터가 손실될 수 있습니다.
- 파라미터 업데이트는 글로벌 데이터 스토어에 속한 클러스터의 로컬 파라미터 그룹을 수정할 때 모든 클러 스터에 적용됩니다.
- 리전 클러스터를 수직(확장 및 축소) 및 수평(확장 및 축소)으로 확장할 수 있습니다. 글로벌 데이터 스토어 를 수정하여 클러스터를 확장할 수 있습니다. 그러면 글로벌 데이터 스토어의 모든 리전 클러스터가 중단 없이 확장됩니다.
- 글로벌 데이터 스토어는 고객 마스터 키, 미사용 데이터 암호화, 전송 중 데이터 암호화 및 Redis AUTH를 지원합니다. 리전 간 통신에 대한 보안은 VPC 피어링을 통해 제공됩니다.

Note

글로벌 데이터 스토어는 다음 규정에 따라 pub/sub 메시지를 지원합니다.

- 클러스터 모드가 비활성화된 경우 pub/sub가 완전히 지원됩니다. 기본 AWS 리전의 마스터에 게 시된 이벤트는 보조 AWS 리전으로 전파됩니다.
- 클러스터 모드가 활성화된 경우 다음 사항이 적용됩니다.
 - 키스페이스에 없는 게시된 이벤트의 경우 동일한 AWS 리전의 구독자만 이벤트를 수신합니다.
 - 게시된 키스페이스 이벤트의 경우 모든 AWS 리전의 구독자가 이벤트를 수신합니다.

글로벌 데이터 스토리지 사용(콘솔)

콘솔을 사용하여 글로벌 데이터 스토어를 생성하려면 다음 2단계 프로세스를 수행합니다.

- 1. 기존 클러스터를 사용하거나 새 클러스터를 생성하여 기본 클러스터를 생성합니다. 엔진은 Redis 5.0.6 이 상이어야 합니다.
- 2. Redis 5.0.6 이상의 엔진을 사용하여 서로 다른 AWS 리전에 최대 2개의 보조 클러스터를 추가합니다.

다음 절차에서는 Redis용 글로벌 데이터 스토어를 생성하고 Redis용 ElastiCache 콘솔을 사용하여 다른 작 업을 수행하는 방법에 대해 설명합니다.

주제

- 기존 클러스터를 사용하여 글로벌 데이터 스토어 생성 (p. 117)
- 새 기본 클러스터를 사용하여 새 글로벌 데이터 스토어 생성 (p. 119)
- 글로벌 데이터 스토어 세부 정보 보기 (p. 121)
- 글로벌 데이터 스토어에 리전 추가 (p. 122)
- 글로벌 데이터 스토어 수정 (p. 123)
- 보조 클러스터를 기본 클러스터로 승격 (p. 123)
- 글로벌 데이터 스토어에서 리전 제거 (p. 124)
- 글로벌 데이터 스토어 삭제 (p. 124)

기존 클러스터를 사용하여 글로벌 데이터 스토어 생성

이 시나리오에서는 기존 클러스터를 사용하여 새 글로벌 데이터 스토어의 기본 클러스터 역할을 합니다. 그런 다음 별도의 AWS 리전에 보조 읽기 전용 클러스터를 생성합니다. 이 보조 클러스터는 기본 클러스터에서 자동 및 비동기 업데이트를 받습니다.

Important

기존 클러스터는 Redis 5.0.6 이상의 엔진을 사용해야 합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 글로벌 데이터 스토리지 사용(콘솔)

기존 클러스터를 사용하여 글로벌 데이터 스토어를 생성하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 Redis를 선택한 후 클러스터를 선택합니다.
- 3. 작업에 대해 글로벌 데이터 스토어 설정을 선택합니다.
- 4. 글로벌 데이터 스토어 설정 페이지에서 다음을 수행합니다.
 - 글로벌 데이터 스토어 이름 접미사 값 입력: 이 접미사는 글로벌 데이터 스토어의 고유 이름을 생성하는 데 사용됩니다. 여기에서 지정하는 접미사를 사용하여 글로벌 데이터 스토어를 검색할 수 있습니다.
 - (선택 사항) 설명 값을 입력합니다.
- 5. 보조 클러스터 세부 정보에서 클러스터가 저장될 다른 AWS 리전을 선택합니다.
- 6. Redis 설정에서 클러스터의 이름에 대한 값과 필요에 따라 설명에 값을 입력합니다.
- 7. 다음 옵션을 그대로 유지하십시오. 기본 클러스터 구성과 일치하도록 미리 채워져 있으므로 변경할 수 없습니다.
 - 엔진 버전
 - 노드 유형
 - 파라미터 그룹

Note

ElastiCache는 제공된 파라미터 그룹의 값에서 새 파라미터 그룹을 자동으로 생성하고 새 파라미터 그룹을 클러스터에 적용합니다. 글로벌 데이터 스토어의 파라미터를 수정하려면 이새 파라미터 그룹을 사용합니다. 자동 생성된 각 파라미터 그룹은 하나의 클러스터에만 연결되므로 하나의 글로벌 데이터 스토어에만 연결됩니다.

- 샤드 수
- 저장된 암호화

Note

고객 관리형 고객 마스터 키를 선택하고 키를 선택하여 다른 암호화 키를 제공할 수 있습니다.

- 전송 중 데이터 암호화
- Redis AUTH
- 8. (선택 사항) 필요에 따라 나머지 보조 클러스터 설정을 업데이트합니다. 기본 클러스터와 동일한 값으로 미리 채워지지만 해당 클러스터에 대한 특정 요구 사항을 충족하도록 업데이트할 수 있습니다.
 - 포트
 - 복제본 개수
 - · Subnet Group
 - 기본 가용 영역
 - 보안 그룹
 - 고객 관리형(고객 마스터 키)
 - Redis AUTH 토큰
 - 자동 백업 활성화
 - 백업 보존 기간
 - 백업 기간
 - 유지 관리 기간

9. 생성을 선택합니다. 이렇게 하면 글로벌 데이터 스토어의 상태가 생성 중으로 설정됩니다. 기본 클러스터가 글로벌 데이터 스토어와 연결되고 보조 클러스터가 연결 중 상태이면 상태가 수정 중으로 전환됩니다

기본 클러스터 및 보조 클러스터가 글로벌 데이터 스토어와 연결되면 상태가 사용 가능으로 변경됩니다. 이 시점에서 읽기 및 쓰기를 허용하는 기본 클러스터와 기본 클러스터에서 복제된 읽기를 허용하는 보조 클러스터가 있습니다.

Redis 페이지가 업데이트되어 클러스터가 글로벌 데이터 스토어의 일부인지 여부를 나타냅니다.

- 글로벌 데이터 스토어 클러스터가 속한 글로벌 데이터 스토어의 이름입니다.
- 글로벌 데이터 스토어 역할 클러스터의 역할(기본 또는 보조)입니다.

다른 AWS 리전에 보조 클러스터를 최대 1개까지 추가할 수 있습니다. 자세한 내용은 글로벌 데이터 스토어에 리전 추가 (p. 122) 단원을 참조하십시오.

새 기본 클러스터를 사용하여 새 글로벌 데이터 스토어 생성

새 글로벌 데이터 스토어를 생성하도록 선택한 경우 다음 절차를 따르십시오.

새 글로벌 데이터 스토어를 생성하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 글로벌 데이터 스토어를 선택한 다음 생성을 선택합니다.
- 3. 글로벌 데이터 스토어 생성에서 다음을 수행합니다.
 - a. 글로벌 데이터 스토어 이름 접미사를 값을 입력합니다. ElastiCache는 접미사를 사용하여 글로벌 데이터 스토어의 고유 이름을 생성합니다. 여기에서 지정하는 접미사를 사용하여 글로벌 데이터 스토어를 검색할 수 있습니다.
 - b. (선택 사항) 글로벌 데이터 스토어 설명에 값을 입력합니다.
- 4. 기본 클러스터 세부 정보에서 리전에 대해 사용 가능한 AWS 리전과 다음 옵션 중 하나를 선택합니다.
 - 새 리전 클러스터를 기본 클러스터로 생성 (p. 119)를 선택하십시오.
 - 기존 클러스터를 기본 클러스터로 사용 (p. 120)

새 리전 클러스터를 기본 클러스터로 생성

글로벌 데이터 스토어의 기본 클러스터로 사용할 새 리전 클러스터를 생성하려면 다음을 수행합니다.

새 리전 클러스터를 기본 클러스터로 생성하려면

- 1. 이름에 클러스터의 이름을 입력합니다.
- 2. (선택 사항) 설명에 클러스터에 대한 설명을 추가합니다.
- 3. 특정 요구 사항에 맞게 나머지 설정을 구성합니다. 파라미터 그룹을 선택하여 엔진 구성 값을 설정하면 해당 파라미터 그룹이 글로벌 데이터 스토어의 모든 클러스터에 적용됩니다. 파라미터 그룹 페이지에서 yes/no 글로벌 속성은 파라미터 그룹이 글로벌 데이터 스토어의 일부인지 여부를 나타냅니다.
- 4. [Next]를 선택합니다.
- 5. 보조 클러스터 세부 정보에서 클러스터가 저장될 다른 AWS 리전을 선택합니다.
- 6. Redis 설정에서 클러스터의 이름에 대한 값과 필요에 따라 설명에 값을 입력합니다.
- 7. 다음 필드는 기본 클러스터 구성과 일치하도록 미리 채워지며 변경할 수 없습니다.
 - 엔진 버전

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 글로벌 데이터 스토리지 사용(콘솔)

- 인스턴스 유형
- 노드 유형
- 샤드 수
- 파라미터 그룹

Note

ElastiCache는 제공된 파라미터 그룹의 값에서 새 파라미터 그룹을 자동으로 생성하고 새 파라미터 그룹을 클러스터에 적용합니다. 글로벌 데이터 스토어의 파라미터를 수정하려면 이 새 파라미터 그룹을 사용합니다. 자동 생성된 각 파라미터 그룹은 하나의 클러스터에만 연결되므로 하나의 글로벌 데이터 스토어에만 연결됩니다.

• 저장된 암호화

Note

고객 관리형 고객 마스터 키를 선택하고 키를 선택하여 다른 암호화 키를 제공하는 옵션이 있습니다.

- 전송 중 데이터 암호화
- · Redis AUTH

나머지 보조 클러스터 설정은 기본 클러스터와 동일한 값으로 미리 채워지지만, 다음은 해당 클러스터에 대한 특정 요구 사항을 충족하도록 업데이트할 수 있습니다.

- 포트
- 복제본 개수
- Subnet Group
- 기본 가용 영역
- 보안 그룹
- 고객 관리형(고객 마스터 키)
- Redis AUTH 토큰
- 자동 백업 활성화
- 백업 보존 기간
- 백업 기간
- 유지 관리 기간
- SNS 알림에 대한 주제
- 8. 생성을 선택합니다. 이렇게 하면 글로벌 데이터 스토어의 상태가 생성 중으로 설정됩니다. 기본 클러스터 및 보조 클러스터가 글로벌 데이터 스토어와 연결되면 상태가 사용 가능으로 변경됩니다. 읽기 및 쓰기를 허용하는 기본 클러스터와 기본 클러스터에서 복제된 읽기를 허용하는 보조 클러스터가 있습니다.

또한 Redis 페이지가 업데이트되어 클러스터가 다음을 포함하여 글로벌 데이터 스토어의 일부인지 여부를 나타냅니다.

- 글로벌 데이터 스토어 클러스터가 속한 글로벌 데이터 스토어의 이름입니다.
- 글로벌 데이터 스토어 역할 클러스터의 역할(기본 또는 보조)을 반영합니다.

다른 AWS 리전에 보조 클러스터를 최대 1개까지 추가할 수 있습니다. 자세한 내용은 글로벌 데이터 스토어에 리전 추가 (p. 122) 단원을 참조하십시오.

기존 클러스터를 기본 클러스터로 사용

글로벌 데이터 스토어 세부 정보 보기

기존 글로벌 데이터 스토어의 세부 정보를 보고 글로벌 데이터 스토어 페이지에서 수정할 수도 있습니다.

글로벌 데이터 스토어 세부 정보를 보려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 글로벌 데이터 스토어를 선택한 다음 사용 가능한 글로벌 데이터 스토어를 선택합니다.

그러면 다음 글로벌 데이터 스토어 속성을 검사할 수 있습니다.

- 글로벌 데이터 스토어 이름: 글로벌 데이터 스토어의 이름입니다.
- 설명: 글로벌 데이터 스토어에 대한 설명입니다.
- 상태: 옵션은 다음과 같습니다.
 - 생성 중
 - [Modifying]
 - 응시 가능
 - 삭제 중
 - 기본 전용 이 상태는 글로벌 데이터 스토어에 기본 클러스터만 포함되어 있음을 나타냅니다. 모든 보조 클러스터가 삭제되거나 성공적으로 생성되지 않습니다.
- 클러스터 모드: 활성화되거나 비활성화됩니다.
- Redis 엔진 버전: 글로벌 데이터 스토어를 실행하는 Redis 엔진 버전입니다.
- 인스턴스 노드 유형: 글로벌 데이터 스토어에 사용되는 노드 유형입니다.
- 미사용 데이터 암호화: 활성화되거나 비활성화됩니다.
- 전송 중 데이터 암호화: 활성화되거나 비활성화됩니다.
- Redis AUTH: 활성화되거나 비활성화됩니다.

글로벌 데이터 스토어를 다음과 같이 변경할 수 있습니다.

- 글로벌 데이터 스토어에 리전 추가 (p. 122)
- 글로벌 데이터 스토어에서 리전 제거 (p. 124)
- 보조 클러스터를 기본 클러스터로 승격 (p. 123)
- 글로벌 데이터 스토어 수정 (p. 123)

글로벌 데이터 스토어 페이지에는 글로벌 데이터 스토어를 구성하는 개별 클러스터와 각각에 해당하는 다음 속성도 나열됩니다.

- 리전 클러스터가 저장된 AWS 리전입니다.
- 역할 기본 또는 보조입니다.
- 클러스터 이름 클러스터의 이름입니다.
- 상태 옵션은 다음과 같습니다.
 - 연결 중 클러스터가 글로벌 데이터 스토어에 연결되는 중입니다.
 - 연관됨 클러스터가 글로벌 데이터 스토어에 연결되어 있습니다.
 - 연결 해제 중 글로벌 데이터 스토어 이름을 사용하여 글로벌 데이터 스토어에서 보조 클러스터를 제거 하는 중입니다. 이후에는 보조 클러스터가 더 이상 기본 클러스터에서 업데이트를 수신하지 않지만 해당 AWS 리전에서 독립 실행형 클러스터로 유지됩니다.
 - 연결 해제됨 보조 클러스터가 글로벌 데이터 스토어에서 제거되었으며 이제 AWS 리전에서 독립 실행 형 클러스터가 되었습니다.

• 글로벌 데이터 스토어 지연 – 글로벌 데이터 스토어에 보조 AWS 리전당 하나의 값을 표시합니다. 보조 리전의 마스터 노드와 기본 리전의 마스터 노드 간의 지연입니다. 클러스터 모드가 활성화된 Redis의 경우지연은 샤드 간의 최대 지연을 나타냅니다.

글로벌 데이터 스토어에 리전 추가

기존 글로벌 데이터 스토어에 최대 하나의 AWS 리전을 추가할 수 있습니다. 이 시나리오에서는 기본 클러스터로부터 자동 및 비동기 업데이트를 수신하는 별도의 AWS 리전에 읽기 전용 클러스터를 생성합니다.

글로벌 데이터 스토어에 AWS 리전을 추가하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 연니다.
- 탐색 창에서 글로벌 데이터 스토어를 선택한 다음 글로벌 데이터 스토어 이름에서 글로벌 데이터 스토어 를 선택합니다.
- 3. 리전 추가를 선택합니다.
- 4. 보조 클러스터가 상주할 AWS 리전을 선택합니다.
- Redis 설정에서 클러스터의 이름에 대한 값과 필요에 따라 설명에 값을 입력합니다.
- 6. 다음 옵션을 그대로 유지하십시오. 기본 클러스터 구성과 일치하도록 미리 채워져 있으므로 변경할 수 없습니다.
 - 엔진 버전
 - 인스턴스 유형
 - 노드 유형
 - 샤드 수
 - 파라미터 그룹

Note

ElastiCache는 제공된 파라미터 그룹의 값에서 새 파라미터 그룹을 자동으로 생성하고 새 파라미터 그룹을 클러스터에 적용합니다. 글로벌 데이터 스토어의 파라미터를 수정하려면 이새 파라미터 그룹을 사용합니다. 자동 생성된 각 파라미터 그룹은 하나의 클러스터에만 연결되므로 하나의 글로벌 데이터 스토어에만 연결됩니다.

• 저장된 암호화

Note

고객 관리형 고객 마스터 키를 선택하고 키를 선택하여 다른 암호화 키를 제공하는 옵션이 있습니다.

- 전송 중 데이터 암호화
- · Redis AUTH
- 7. (선택 사항) 필요에 따라 나머지 보조 클러스터 설정을 업데이트합니다. 기본 클러스터와 동일한 값으로 미리 채워지지만 해당 클러스터에 대한 특정 요구 사항을 충족하도록 업데이트할 수 있습니다.
 - 포트
 - 복제본 개수
 - Subnet Group
 - 기본 가용 영역
 - 보안 그룹
 - 고객 관리형(고객 마스터 키)
 - Redis AUTH 토큰
 - 자동 백업 활성화

- 백업 보존 기간
- 백업 기간
- 유지 관리 기간
- SNS 알림에 대한 주제
- 8. 추가를 선택합니다.

글로벌 데이터 스토어 수정

리전 클러스터의 속성을 수정할 수 있습니다. 보조 클러스터를 기본 클러스터로 승격하는 경우를 제외하고 글로벌 데이터 스토어에서는 하나의 수정 작업만 진행 중일 수 있습니다. 자세한 내용은 보조 클러스터를 기본 클러스터로 승격 (p. 123) 단원을 참조하십시오.

글로벌 데이터 스토어를 수정하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 글로벌 데이터 스토어를 선택한 다음 글로벌 데이터 스토어 이름에 글로벌 데이터 스토어를 선택합니다.
- 3. 수정을 선택하고 다음 옵션 중에서 선택합니다.
 - 설명 수정 글로벌 데이터 스토어에 대한 설명을 업데이트합니다.
 - 엔진 버전 수정 Redis 엔진 버전 5.0.6 이상만 사용할 수 있습니다.
 - 노드 유형 수정 리전 클러스터를 수직(확장 및 축소) 및 수평(확장 및 축소)으로 확장합니다. 옵션에는 R5 및 M5 노드 패밀리가 포함됩니다. 노드 유형에 대한 자세한 내용은 지원되는 노드 유형 (p. 54) 단원을 참조하십시오.
 - 자동 장애 조치 수정 장애 조치를 활성화하거나 비활성화합니다. 장애 조치를 활성화하고 리전 클러 스터의 기본 노드가 죽으면 ElastiCache는 리전 복제본 중 하나로 장애 조치합니다. 자세한 내용은 자 동 장애 조치를 참조하십시오.

클러스터 모드가 활성화된 Redis 클러스터의 경우:

- 샤드 추가 추가할 샤드 수를 입력하고 선택적으로 하나 이상의 가용 영역을 지정합니다.
- 샤드 삭제 각 AWS 리전에서 삭제할 샤드를 선택합니다.
- 샤드 재분배 슬롯 분포를 재분배하여 클러스터의 기존 샤드 간에 균일한 분포를 보장합니다.

글로벌 데이터 스토어의 파라미터를 수정하려면 글로벌 데이터 스토어에 대한 멤버 클러스터의 파라미터 그룹을 수정합니다. ElastiCache는 이 변경 사항을 해당 글로벌 데이터 스토어 내의 모든 클러스터에 적용합니다. 해당 클러스터의 파라미터 그룹을 수정하려면 Redis 콘솔 또는 ModicCachCluster API를 사용합니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오. 글로벌 데이터 스토어 내에 포함된 클러스터의 파라미터 그룹을 수정하면 해당 글로벌 데이터 스토어 내의 모든 클러스터에 적용됩니다.

전체 파라미터 그룹 또는 특정 파라미터 재설정하려면 ResetCacheParameterGroup API를 사용합니다.

보조 클러스터를 기본 클러스터로 승격

성능 문제가 발생하여 기본 클러스터 또는 AWS 리전을 사용할 수 없게 되면 보조 클러스터를 기본 클러스터로 승격할 수 있습니다. 다른 수정이 진행 중이더라도 언제든지 승격이 허용됩니다. 또한 여러 승력을 병렬로 실행할 수 있으며 글로벌 데이터 스토어가 최종적으로 하나의 기본 클러스터가 됩니다. 여러 보조 클러스터를 동시에 승격하는 경우 Redis용 ElastiCache에서 궁극적으로 하나의 클러스터가 기본 클러스터가 되는 것을 보장하지 않습니다.

보조 클러스터를 기본 클러스터로 승격하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 Redis 아래 글로벌 데이터 스토어를 선택합니다.
- 3. 세부 정보를 보려면 글로벌 데이터 스토어 이름을 선택합니다.
- 4. 보조 클러스터 선택
- 5. 기본 클러스터로 승격을 선택합니다.

그러면 Promoting a region to primary will make the cluster in this region as read/writable. Are you sure you want to promote the *secondary* cluster to primary? 경고와 함께 결정을 확인하라는 메시지가 표시됩니다.

The current primary cluster in *primary region* will become secondary and will stop accepting writes after this operation completes. Please ensure you update your application stack to direct traffic to the new primary region.

6. 승력을 계속하려면 확인을 선택하고 그렇지 않으면 취소를 선택합니다.

확인하려면 글로벌 데이터 스토어가 수정 중 상태로 전환되어 승격이 완료될 때까지 사용할 수 없습니다.

글로벌 데이터 스토어에서 리전 제거

다음 절차를 사용하여 글로벌 데이터 스토어에서 AWS 리전을 제거할 수 있습니다.

글로벌 데이터 스토어에서 AWS 리전을 제거하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 Redis 아래 글로벌 데이터 스토어를 선택합니다.
- 3. 글로벌 데이터 스토어를 선택합니다.
- 4. 제거할 리전을 선택합니다.
- 5. 리전 제거를 선택합니다.

Note

이 옵션은 보조 클러스터에만 사용할 수 있습니다.

그러면 Removing the region will remove your only available cross region replica for the primary cluster. Your primary cluster will no longer be set up for disaster recovery and improved read latency in remote region. Are you sure you want to remove the selected region from the global datastore? 경고와함께 결정을 확인하라는 메시지가 표시됩니다.

6. 승력을 계속하려면 확인을 선택하고 그렇지 않으면 취소를 선택합니다.

확인을 선택하면 AWS 리전이 제거되고 보조 클러스터는 더 이상 복제 업데이트를 수신하지 않습니다.

글로벌 데이터 스토어 삭제

글로벌 데이터 스토어를 삭제하려면 먼저 모든 보조 클러스터를 제거합니다. 자세한 내용은 글로벌 데이터 스토어에서 리전 제거 (p. 124) 단원을 참조하십시오. 이렇게 하면 글로벌 데이터 스토어가 기본 전용 상태로 유지됩니다.

글로벌 데이터 스토어를 삭제하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 Redis 아래 글로벌 데이터 스토어를 선택합니다.
- 3. 글로벌 데이터 스토어 이름에서 삭제할 글로벌 데이터 스토어를 선택한 다음 삭제를 선택합니다.

그러면 Are you sure you want to delete this Global Datastore? 경고와 함께 결정을 확인하라는 메시지가 표시됩니다.

4. 삭제를 선택합니다.

글로벌 데이터 스토어가 삭제 중 상태로 전환됩니다.

글로벌 데이터 저장소(CLI) 사용

AWS CLI(AWS Command Line Interface)를 사용하면 명령줄에서 여러 AWS 제품을 관리하고 스크립트를 통해 자동화할 수 있습니다. 임시(일회성) 작업에 AWS CLI를 사용할 수 있습니다.

AWS CLI 다운로드 및 구성

AWS CLI는 Windows, macOS 또는 Linux에서 실행됩니다. 다음 절차에 따라 다운로드 및 구성합니다.

CLI를 다운로드, 설치 및 구성하려면

- 1. AWS 명령줄 인터페이스 웹 페이지에서 AWS CLI를 다운로드합니다.
- 2. AWS Command Line Interface 사용 설명서의 AWS CLI 설치 및 AWS CLI 구성 지침을 따릅니다.

글로벌 데이터 스토어에 AWS CLI 사용

글로벌 데이터 스토어를 사용하려면 다음 CLI 작업을 사용합니다.

create-global-replication-group

```
aws elasticache create-global-replication-group \
--global-replication-group-id-suffix my global datastore \
--primary-replication-group-id sample-repl-group \
--global-replication-group-description an optional description of the global datastore
```

• create-replication-group – 이 작업을 사용하여 글로벌 데이터 스토어의 이름을 --globalreplication-group-id 파라미터에 제공하여 글로벌 데이터 스토어에 대한 보조 클러스터를 생성합니다.

```
aws elastiche create-replication-group \
--replication-group-id secondary replication group name \
--replication-group-description "Replication group description" \
--global-replication-group-id global datastore name \
```

· describe-global-replication-groups

```
aws elasticache describe-global-replication-groups \
--global-replication-group-id my global datastore \
--show-member-info an optional parameter that returns a list of the primary and secondary clusters that make up the global datastore
```

· modify-global-replication-group

```
aws elasticache modify-global-replication-group \
--global-replication-group-id my global datastore \
--automatic-failover-enabled yes/no
--cache-node-type node type
--engine-version engine version
--apply-immediately
--global-replication-group-description description
```

· delete-global-replication-group

```
aws elasticache delete-global-replication-group \
   --global-replication-group-id my global datastore \
   --retain-primary-replication-group defaults to true
```

disassociate-global-replication-group

```
aws elasticache disassociate-global-replication-group \
--global-replication-group-id my Global Datastore \
--replication-group-id my secondary cluster \
--replication-group-region the AWS Region in which the secondary cluster resides
```

· failover-global-replication-group

```
aws elasticache failover-replication-group \
--global-replication-group-id my global datastore \
--primary-region The AWS Region of the primary cluster \
--primary-replication-group-id The name of the global datastore, including the suffix.
```

increase-node-groups-in-global-replication-group

```
aws elasticache increase-node-groups-in-global-replication-group \
    --apply-immediately yes\
    --global-replication-group-id global-replication-group-name \
    --node-group-count 3
```

· decrease-node-groups-in-global-replication-group

```
aws elasticache decrease-node-groups-in-global-replication-group \
    --apply-immediately yes\
    --global-replication-group-id global-replication-group-name \
    --node-group-count 3
```

rebalance-shards-in-global-replication-group

```
aws elasticache rebalance-shards-in-global-replication-group \
    --apply-immediately yes\
    --global-replication-group-id global-replication-group-name \
```

도움말을 사용하여 사용 가능한 모든 명령 Redis용 ElastiCache을 나열합니다.

```
aws elasticache help
```

도움말을 사용하면 특정 명령을 설명하고 그 사용법에 대해 자세히 알아볼 수도 있습니다.

aws elasticache create-global-replication-group help

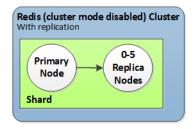
고가용성을 위한 복제 그룹 사용

단일 노드 Amazon ElastiCache Redis 클러스터는 제한된 데이터 보호 서비스(AOF)를 사용할 수 있는 인 메모리 개체입니다. 어떤 이유로든 클러스터에 장애가 발생하면 클러스터의 모든 데이터가 손실됩니다. 그러나 Redis 엔진을 실행 중인 경우 2 ~ 6개의 노드를 복제본이 있는 클러스터로 그룹화할 수 있습니다. 이 복제본에서는 1 ~ 5개의 읽기 전용 노드에 해당 그룹의 단일 읽기/쓰기 기본 노드에 대한 복제본 데이터가 포함됩니다. 이 시나리오에서는 어떤 이유로든 한 노드에 장애가 발생해도 데이터가 모두 손실되지는 않습니다. 왜냐하면 한 노드가 하나 이상의 다른 노드에 복제되어 있기 때문입니다. 복제 지연 시간으로 인해 기본 읽기/쓰기 노드가 실패할 경우 일부 데이터가 손실될 수 있습니다.

다음 그래픽에 나와 있는 대로 복제 구조는 Redis 클러스터 내에 포함된 샤드(API/CLI에서는 노드 그룹이라고 함) 내에 포함되어 있습니다. Redis(클러스터 모드 비활성화됨) 클러스터는 항상 단일 샤드를 포함합니다. Redis(클러스터 모드 활성화됨) 클러스터는 클러스터의 데이터가 샤드에 분할된 최대 90개의 샤드를 포함할수 있습니다. 하나의 클러스터당 최대 90개의 노드로 구성된 더 많은 수의 샤드와 더 적은 수의 복제본을 가진 클러스터를 생성할수 있습니다. 이 클러스터 구성은 90개의 샤드 및 0개의 복제본부터 15개의 샤드 및 5개의 복제본까지 해당될수 있으며, 이는 허용되는 최대 복제본 수입니다.

Note

노드 또는 샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 한도 증가를 요청하는 방법에 대한 지침은 AWS 서비스 제한을 참조하고 한도 유형을 '인스턴스 유형별 클러스터당 노드'로 선택하십시오.



Redis(클러스터 모드 비활성화됨) 클러스터에는 샤드 1개와 복제본 노드 0~5개가 포함

복제본이 있는 클러스터에 자동 장애 조치가 포함된 다중 AZ가 활성화되어 있고 기본 노드에 장애가 발생하면 기본 노드가 읽기 전용 복제본으로 자동 장애 조치됩니다. 복제본 노드의 데이터가 비동기적으로 업데이트되기 때문에 복제본 노드를 업데이트할 때 지연 시간으로 인해 일부 데이터가 손실될 수 있습니다. 자세한 내용은 Redis 실행 시 장애 완화 (p. 395) 단원을 참조하십시오.

주제

- Redis 복제 이해 (p. 128)
- 복제: Redis(클러스터 모드 비활성화됨) 대 Redis(클러스터 모드 활성화됨) (p. 130)
- 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132)
- 동기화 및 백업 구현 방법 (p. 143)
- Redis 복제 그룹 생성 (p. 143)
- 복제 그룹의 세부 정보 보기 (p. 161)
- 복제 그룹 엔드포인트 확인 (p. 164)
- 복제 그룹 수정 (p. 165)
- 복제 그룹 삭제 (p. 166)

- 복제본 수 변경 (p. 167)
- Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본을 기본으로 승격 (p. 178)

Redis 복제 이해

Redis는 다음과 같은 2가지 방법으로 복제를 구현합니다.

- 각 노드에 클러스터의 모든 데이터를 포함하고 있는 샤드 1개가 있음—Redis(클러스터 모드 비활성화됨)
- 데이터가 최대 90개 샤드로 분할되어 있음—Redis(클러스터 모드 활성화됨)

복제 그룹의 각 샤드에는 읽기/쓰기 기본 노드 하나와 최대 5개의 읽기 전용 복제본 노드가 있습니다. 하나의 클러스터당 최대 90개의 노드로 구성된 더 많은 수의 샤드와 더 적은 수의 복제본을 가진 클러스터를 생성할 수 있습니다. 이 클러스터 구성은 90개의 샤드 및 0개의 복제본부터 15개의 샤드 및 5개의 복제본까지 해당 될 수 있으며, 이는 허용되는 최대 복제본 수입니다.

Note

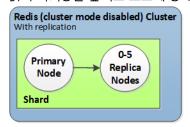
노드 또는 샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 한도 증가를 요청하는 방법에 대한 지침은 AWS 서비스 제한을 참조하고 한도 유형을 '인스턴스 유형별 클러스터당 노드'로 선택하십시오.

주제

- Redis(클러스터 모드 비활성화됨) (p. 128)
- Redis(클러스터 모드 활성화됨) (p. 129)

Redis(클러스터 모드 비활성화됨)

Redis(클러스터 모드 비활성화됨) 클러스터에는 단일 샤드가 있으며 이 샤드 내에는 Redis 노드 모음이 있습니다. 이 모음은 하나의 기본 읽기/쓰기 노드와 최대 5개의 보조 읽기 전용 복제본 노드로 구성됩니다. 각 읽기 전용 복제본은 클러스터의 기본 노드에서 가져온 데이터 사본을 유지합니다. 비동기식 복제 메커니즘은 읽기 전용 복제본이 기본 노드와 동기화되어 있는 상태를 유지하는 데 사용됩니다. 애플리케이션은 클러스터의 모든 노드로부터 읽을 수 있습니다. 애플리케이션은 기본 노드에만 쓸 수 있습니다. 읽기 전용 복제본은 읽기 처리량을 높이고 노드에 장애가 발생할 경우 데이터 손실을 방지합니다.



단일 샤드 및 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터

복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터를 사용하여 Redis 솔루션을 ElastiCache에 맞게 조정하여 읽기 집약적인 애플리케이션을 처리하거나 동일한 클러스터에서 동시에 읽는 많은 수의 클라 이언트를 지원할 수 있습니다.

Redis(클러스터 모드 비활성화됨) 클러스터의 모든 노드는 같은 리전에 있어야 합니다. 내결함성을 개선하기 위해 해당 리전 내의 여러 가용 영역에서 읽기 전용 복제본을 프로비저닝할 수 있습니다.

읽기 전용 복제본을 클러스터에 추가하면 기본 노드에 있는 모든 데이터가 새 노드로 복사됩니다. 이 시점부터 기본 노드에 데이터를 쓸 때마다 변경 사항이 모든 읽기 전용 복제본에 비동기식으로 전파됩니다.

내결함성을 개선하고 기록 시간을 단축하려면 복제본이 있는 Redis(클러스터 모드 비활성화됨) 클러스터에 대해 자동 장애 조치가 포함된 다중 AZ를 활성화합니다. 자세한 내용은 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132) 단원을 참조하십시오.

기본 노드의 역할과 복제본 중 하나의 역할을 서로 교환함으로써 Redis(클러스터 모드 비활성화됨) 클러스터 내 노드의 역할을 변경할 수 있습니다. 성능 튜닝을 위해 이런 방식을 선택할 수 있습니다. 예를 들어, 쓰기 작업이 많은 웹 애플리케이션의 경우 네트워크 지연 시간이 가장 짧은 노드를 선택할 수 있습니다. 자세한 내용은 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본을 기본으로 승격 (p. 178) 단원을 참조하십시오.

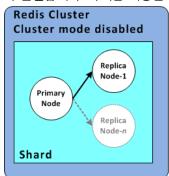
Redis(클러스터 모드 활성화됨)

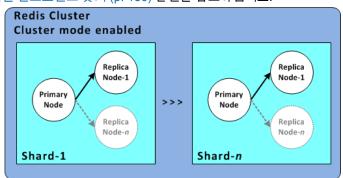
Redis(클러스터 모드 활성화됨) 클러스터는 1~90개의 샤드(API/CLI의 경우 노드 그룹)로 구성됩니다. 각 샤드에는 읽기/쓰기 기본 노드 및 최대 5개의 읽기 전용 복제본 노드가 있습니다. 이 구성은 90개의 샤드 및 0개의 복제본부터 15개의 샤드 및 5개의 복제본까지 해당될 수 있으며, 이는 허용되는 최대 복제본 수입니다. 샤드의 각 읽기 전용 복제본은 샤드의 기본 노드에서 가져온 데이터 사본을 유지합니다. 비동기식 복제 메커니 즘은 읽기 전용 복제본이 기본 노드와 동기화되어 있는 상태를 유지하는 데 사용됩니다. 애플리케이션은 클러스터의 모든 노드로부터 읽을 수 있습니다. 애플리케이션은 기본 노드에만 쓸 수 있습니다. 읽기 전용 복제 본은 읽기 확장성을 개선하고 데이터 손실을 방지합니다. 데이터는 Redis(클러스터 모드 활성화됨) 클러스터의 샤드 간에 파티셔닝됩니다.

Note

노드/샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 제한을 높이도록 요청하려면 AWS 서비스 제한을 참조하고 요청에 인스턴스 유형을 포함하십시오.

애플리케이션은 Redis(클러스터 모드 활성화됨) 클러스터의 구성 엔드포인트를 사용하여 클러스터의 노드와 연결합니다. 자세한 내용은 연결 엔드포인트 찾기 (p. 180) 단원을 참조하십시오.





다중 샤드 및 복제본 노드가 있는 Redis(클러스터 모드 활성화됨) 클러스터

Redis(클러스터 모드 활성화됨) 클러스터의 모든 노드는 같은 리전에 있어야 합니다. 내결함성을 개선하기 위해 해당 리전 내의 여러 가용 영역에서 기본 노드와 읽기 전용 복제본을 모두 프로비저닝할 수 있습니다.

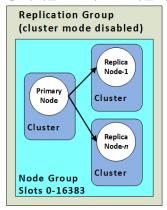
모든 Redis(클러스터 모드 활성화됨) 클러스터에는 자동 장애 조치가 포함된 다중 AZ가 필요합니다. 자세한 내용은 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132) 단원을 참조하십시오.

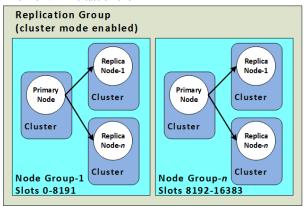
현재 Redis(클러스터 모드 활성화됨)에는 다음과 같은 몇 가지 제한이 있습니다.

- 복제본 노드는 수동으로 기본 노드로 승격할 수 없습니다.
- 자동 장애 조치가 포함된 다중 AZ가 필요합니다.
- 백업에서의 복원을 통해서만 클러스터 구조, 노드 유형 및 노드 수를 변경할 수 있습니다. 자세한 내용은 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오. 클러스터가 계속 해서 읽기 및 쓰기 요청을 처리하는 동안 Redis(클러스터 모드 활성화됨) 클러스터의 샤드 수는 동적으로 변경될 수 있습니다. 자세한 내용은 Redis(클러스터 모드 활성화됨)를 위한 온라인 리샤딩 및 샤드 재분 배 (p. 243) 단원을 참조하십시오.

복제: Redis(클러스터 모드 비활성화됨) 대 Redis(클러스터 모드 활성화됨)

Redis 버전 3.2부터는 서로 다른 두 Redis 클러스터(API/CLI의 경우 복제 그룹) 유형 중 하나를 생성할 수 있습니다. Redis(클러스터 모드 비활성화됨) 클러스터에는 최대 5개의 읽기 전용 복제본 노드가 있는 단일 샤드(API/CLI의 경우 노드 그룹)가 항상 있습니다. Redis(클러스터 모드 활성화됨) 클러스터에는 각기 읽기 전용 복제본 노드가 $1\sim5$ 개인 샤드가 최대 90개 있습니다.





Redis(클러스터 모드 비활성화됨) 및 Redis(클러스터 모드 활성화됨) 클러스터

다음 표에는 Redis(클러스터 모드 비활성화됨) 클러스터와 Redis(클러스터 모드 활성화됨) 클러스터의 중요한 차이점이 요약되어 있습니다.

Redis(클러스터 모드 비활성화됨) 클러스터와 Redis(클러스터 모드 활성화됨) 클러스터 비교

기능	Redis(클러스터 모드 비활성화됨)	Redis(클러스터 모드 활성화됨)
수정 가능	예. 복제본 노드 추가/삭제 및 노드 유형 확장을 지원합니다.	제한. 자세한 내용은 엔진 버전 업 그레이드 (p. 45) 및 Redis(클 러스터 모드 활성화됨)에서 클러 스터 조정 (p. 241) 단원을 참조 하십시오.
데이터 파티셔닝	아니요	예
샤드	1	1 ~ 90개
읽기 전용 복제본	0 ~ 5개 Important 복제본이 없으며 노드에 장애가 발생하면 전체 데 이터가 손실됩니다.	샤드당 0 ~ 5개. Important 복제본이 없으며 노드에 장애가 발생하면 전체 데 이터가 손실됩니다.
자동 장애 조치가 포함된 다중 AZ	예, 최소 1개의 복제본이 있어야합니다. 선택. 기본적으로 활성화되어 있습니다.	예. 필수.
스냅샷 (백업)	예, 단일 .rdb 파일을 생성합니다.	예, 각 샤드에 고유한 .rdb 파일을 생성합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 복제: Redis(클러스터 모드 비활성화 됨) 대 Redis(클러스터 모드 활성화됨)

기능	Redis(클러스터 모드 비활성화됨)	Redis(클러스터 모드 활성화됨)
복원	예, Redis(클러스터 모드 비활성 화됨) 클러스터에서 단일 .rdb 파 일을 사용합니다.	예, Redis(클러스터 모드 비활성 화됨) 또는 Redis(클러스터 모드 활성화됨) 클러스터에서 단일 .rdb 파일을 사용합니다.
지원되는 버전	모든 Redis 버전	Redis 3.2 이상
엔진 업그레이드 가능 여부	예, 하지만 몇 가지 제약이 있습니다. 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오.	예, 하지만 몇 가지 제약이 있습니다. 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오.
암호화	버전 3.2.6 및 4.0.10 이상.	버전 3.2.6 및 4.0.10 이상.
HIPAA 적격	버전 3.2.6 및 4.0.10 이상.	버전 3.2.6 및 4.0.10 이상.
PCI DSS 준수	버전 3.2.6 및 4.0.10 이상.	버전 3.2.6 및 4.0.10 이상.
온라인 리샤딩	해당 사항 없음	버전 3.2.10 이상.

무엇을 선택해야 할까요?

Redis(클러스터 모드 비활성화됨) 또는 Redis(클러스터 모드 활성화됨) 중에서 선택할 때는 다음 사항을 고려하십시오.

• 조정과 파티셔닝 – 비즈니스 요구는 변환합니다. 최고 수요가 발생할 때를 대비하거나 수요가 변화함에 따라 조정해야 합니다. Redis(클러스터 모드 비활성화됨)은 조정을 지원합니다. 복제본 노드를 추가하거나 삭제하여 읽기 용량을 조정하거나 더 큰 노드 유형까지 확장하여 용량을 조정할 수 있습니다. 이 두 작업은 모두 시간이 소요됩니다. 자세한 내용은 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터 조정 (p. 230) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨)는 최대 90개의 노드 그룹으로 데이터 분할을 지원합니다. 비즈니스에 변경이 필요할 때마다 샤드 수를 동적으로 변경할 수 있습니다. 파티셔닝의 이점 중 하나는 로드를 더 많은 엔드포인트로 분산시켜 최고 수요가 발생할 때 액세스 병목 현상을 줄이는 것입니다. 또한 데이터가 여러서버에 분산될 수 있으므로 더 큰 데이터 세트를 수용할 수 있습니다. 파티션 확장에 대한 자세한 내용은 Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241) 단원을 참조하십시오.

- 노드 크기와 노드 수 Redis(클러스터 모드 비활성화됨) 클러스터에는 샤드가 하나만 있기 때문에 노드 유형이 클러스터의 모든 데이터와 필요한 오버헤드를 수용할 수 있을 만큼 충분히 커야 합니다. 반면 Redis(클러스터 모드 활성화됨) 클러스터를 사용하면 데이터를 여러 샤드에 파티셔닝할 수 있기 때문에 데이터가 더 필요해도 노드 유형이 더 작을 수 있습니다. 자세한 내용은 노드 크기 선택 (p. 67) 단원을 참조하십시오.
- 읽기와 쓰기 클러스터의 기본 로드가 데이터를 읽는 애플리케이션인 경우 읽기 전용 복제본을 추가하고 삭제하여 Redis(클러스터 모드 비활성화됨) 클러스터를 조정할 수 있습니다. 그러나 5 읽기 전용 복제본의 최대치가 있습니다. 클러스터의 로드가 주로 쓰기 작업인 경우 여러 샤드가 있는 Redis(클러스터 모드 활 성화됨) 클러스터의 추가 쓰기 엔드포인트가 유용할 수 있습니다.

어떤 유형의 클러스터를 구현하도록 선택하든 현재와 미래의 요구에 적합한 노드 유형을 선택해야 합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서

작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ

작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 A7

특정 경우에 Redis용 ElastiCache는 기본 노드를 감지하여 교체합니다. 이러한 경우에는 특정 유형의 계획적 인 유지 관리 및 드물지만 기본 노드나 가용 영역에 장애가 발생하는 경우가 포함됩니다.

이러한 교체로 인해 클러스터에 약간의 가동 중지가 발생합니다. 클러스터에서 자동 장애 조치 기능을 포함 한 다중 AZ를 활성화하면 가동 중지가 최소화됩니다. 이 경우 기본 노드의 역할이 읽기 전용 복제본 중 하나 로 장애 조치됩니다. 새로운 기본 노드를 생성하고 프로비저닝할 필요가 없습니다. 이 장애 조치 및 복제본 승격을 통해 승격이 완료되는 즉시 새 기본 노드에 작성을 재개할 수 있습니다.

또한 ElastiCache는 승격된 복제본의 DNS(Domain Name Service) 이름을 전파합니다. 이렇게 하면 애플리 케이션이 기본 엔드포인트에 쓰는 경우 애플리케이션에서 엔드포인트를 변경할 필요가 없기 때문입니다. 개 별 엔드포인트를 읽을 경우 기본으로 승격된 복제본의 읽기 엔드포인트를 새 복제본의 엔드포인트로 변경이 필요합니다.

계획된 노드 교체의 경우. 유지 관리 업데이트 또는 셀프 서비스 업데이트로 인해 시작되었습니다.

- ElastiCache for Redis 클러스터의 경우, 클러스터에서 들어오는 쓰기 요청을 처리하는 중에 계획된 노드 교체가 완료됩니다.
- 다중 AZ 자동 장애 조치가 활성화되어 5.0.5 이상 엔진에서 실행 중인 Redis 클러스터 모드 비활성화 클러 스터의 경우, 클러스터에서 들어오는 쓰기 요청을 처리하는 중에 계획된 노드 교체가 완료됩니다.
- 다중 AZ 자동 장애 조치가 활성화되어 5.0.4 이상 엔진에서 실행 중인 Redis 클러스터 모드 활성화 클러스 터의 경우, DNS 업데이트와 관련하여 최대 수 초 간의 짧은 쓰기 중단이 발생할 수 있습니다. 이 프로세스 는 자동 장애 조치를 포함하는 다중 AZ를 활성화하지 않은 경우 새 기본 노드를 다시 생성하고 프로비저닝 하는 것보다 훨씬 빠릅니다.

ElastiCache 관리 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 자동 장애 조치가 있는 다중 AZ를 활성 화할 수 있습니다.

Redis 클러스터(API 및 CLI의 복제 그룹)의 장애 조치 기능이 있는 ElastiCache 다중 AZ를 활성화하면 내결 함성이 개선됩니다. 특히 클러스터의 읽기/쓰기 기본 클러스터 노드에 접속할 수 없거나 어떤 이유로든 실패 하는 경우에 특히 그렇습니다. 자동 장애 조치가 있는 다중 AZ는 복제를 지원하는 Redis 클러스터에서만 지 원됩니다.

주제

- 자동 장애 조치가 있는 Redis 다중 AZ에 대한 중요 참고 사항 (p. 132)
- 다중 AZ 및 자동 장애 조치 응답을 통한 장애 시나리오 (p. 133)
- 자동 장애 조치가 있는 다중 AZ 활성화 (p. 136)
- 자동 장애 조치가 있는 다중 AZ 테스트 (p. 139)

자동 장애 조치가 있는 Redis 다중 AZ에 대한 중요 참고 사항

다음은 자동 장애 조치가 있는 Redis 다중 AZ의 유의 사항입니다.

- 자동 장애 조치가 있는 다중 AZ는 Redis 버전 2.8.6 이상에서 지원됩니다.
- 자동 장애 조치가 있는 Redis 다중 AZ는 T1 노드 유형에서 지원되지 않습니다.
- Redis 복제는 비동기식입니다. 따라서 기본 클러스터를 복제본으로 장애 조치하면 복제 지연으로 인해 소 량의 데이터가 손실될 수 있습니다.

기본으로 승격할 복제본을 선택할 때 Redis용 ElastiCache는 최소 복제 지연 시간으로 복제본을 선택합니 다. 즉, 가장 최신 복제본을 선택합니다. 이로써 손실 데이터 양을 최소화할 수 있습니다. 복제 지연 시간이

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ

가장 짧은 복제본은 실패한 기본 노드와 같은 가용 영역에 있을 수도 있고 다른 가용 영역에 있을 수도 있습니다.

- 자동 장애 조치가 있는 다중 AZ가 비활성화된 경우에만 읽기 전용 복제본을 기본으로 수동 승격할 수 있습니다. 읽기 전용 복제본을 기본으로 승격하려면 다음 단계를 따릅니다.
 - 1. 클러스터에서 자동 장애 조치가 포함된 다중 AZ를 비활성화합니다.
 - 2. 읽기 전용 복제본을 기본으로 승격합니다.
 - 3. 자동 장애 조치가 포함된 다중 AZ를 다시 활성화합니다.

Redis(클러스터 모드 활성화됨) 클러스터에서 자동 장애 조치가 있는 다중 AZ를 비활성화할 수 없습니다. 따라서 Redis(클러스터 모드 활성화됨) 클러스터에서 수동으로 복제본을 기본으로 승격할 수 없습니다.

- Redis용 ElastiCache 자동 장애 조치가 있는 다중 AZ와 AOF(Append-Only File)를 함께 사용할 수 없습니다. 하나를 활성화하면 다른 하나를 활성화할 수 없습니다.
- 노드 장애는 드물지만 전체 가용 영역에 장애가 발생하는 경우로 인해 발생할 수 있습니다. 이 경우 장애가 발생한 기본 서버를 대체하는 복제본은 가용 영역이 백업된 경우에만 생성됩니다. 예를 들어, AZ-a에 기본 노드가 있고 AZ-b 및 AZ-c에 복제본이 있는 복제 그룹을 가정해 보겠습니다. 기본 노드에 문제가 발생하면 복제 지연 시간이 가장 짧은 사용 가능한 복제본을 기본 노드로 승격시킵니다. 그런 다음 ElastiCache는 AZ-a가 백업되어 사용할 수 있는 경우에만 AZ-a(장애가 발생한 기본이 있는 위치)에 새 복제본을 생성합니다.
- 고객이 실행한 기본 재부팅은 자동 장애 조치를 트리거하지 않습니다. 다른 재부팅 및 장애는 자동 장애 조 치를 트리거합니다.
- 기본을 재부팅할 때마다 온라인 상태가 되면 데이터가 지워집니다. 읽기 전용 복제본은 기본 클러스터가 지워진 것을 확인하면 데이터 복제본을 지우기 때문에 데이터가 손실됩니다.
- 읽기 전용 복제본이 승격된 후 다른 복제본은 새 기본 복제본과 동기화됩니다. 초기 동기화 후에는 복제본의 콘텐츠가 삭제되고 새 기본의 데이터가 동기화되어 복제본에 액세스하지 못하는 동안 짧은 중단이 발생합니다. 또한 이 동기화 프로세스로 인해 복제본과 동기화되는 동안 기본에 임시 로드가 증가합니다. 이 동작은 Redis의 기본 동작이며 ElastiCache 다중 AZ에 고유하지 않습니다. 이 Redis 동작에 대한 자세한 내용은 http://redis.io/topics/replication을 참조하십시오.

Important

- Redis 버전 2.8.22 이상에서는 외부 복제본이 허용되지 않습니다.
- Redis 버전 2.8.22 이전 버전의 경우 자동 장애 조치가 있는 다중 AZ가 활성화된 ElastiCache Redis 클러스터에 외부 Redis 복제본을 연결하지 않는 것이 좋습니다. 이는 지원하지 않는 구성으로 ElastiCache가 장애 조치 및 복구를 제대로 수행하지 못하는 문제를 유발할 수 있습니다. 외부 Redis 복제본을 ElastiCache 클러스터에 연결해야 하는 경우, 연결 전 자동 장애 조치가 있는다중 AZ가 비활성화되어 있는지 확인해야 합니다.

다중 AZ 및 자동 장애 조치 응답을 통한 장애 시나리오

자동 장애 조치가 있는 다중 AZ를 도입하기 전에 ElastiCache는 장애가 발생한 노드를 재생성하고 재프로비 저닝하여 클러스터의 장애가 발생한 노드를 탐지해 교체합니다. 자동 장애 조치가 있는 다중 AZ를 활성화하면 장애가 발생한 기본 노드가 복제 지연 시간이 가장 짧은 복제본으로 장애 조치됩니다. 선택한 복제본이 자동으로 승격되기 때문에 새 기본 노드를 생성하고 프로비저닝하는 것보다 훨씬 빠릅니다. 이 프로세스는 보통 클러스터에 다시 작성하려면 몇 초 정도 소요됩니다.

자동 장애 조치가 있는 다중 AZ가 활성화되면 ElastiCache가 기본 노드의 상태를 지속적으로 모니터링합니다. 기본 노드에 장애가 발생하면 장애 특성에 따라 다음 작업 중 하나가 수행됩니다.

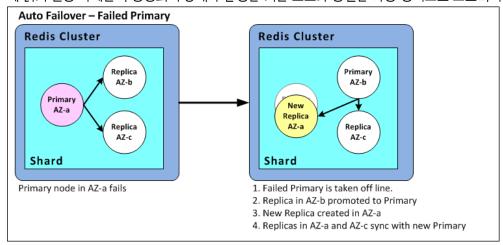
장애 시나리오

- 기본 노드에만 장애가 발생한 경우 (p. 134)
- 기본 노드 및 일부 읽기 전용 복제본에 장애가 발생한 경우 (p. 135)

• 전체 클러스터에 장애가 발생한 경우 (p. 135)

기본 노드에만 장애가 발생한 경우

기본 노드에만 장애가 발생한 경우 복제 지연 시간이 가장 짧은 읽기 전용 복제본이 기본 노드로 승격되고 대체 읽기 전용 복제본이 생성되어 장애가 발생한 기본 노드와 동일한 가용 영역으로 프로비저닝됩니다.



장애가 발생한 기본 노드에 대한 자동 장애 조치

기본 노드에만 장애가 발생한 경우 자동 장애 조치가 있는 ElastiCache 다중 AZ는 다음과 같습니다.

- 1. 장애가 발생한 기본 노드는 오프라인 상태로 전환됩니다.
- 2. 복제 지연 시간이 가장 짧은 읽기 전용 복제본을 기본으로 승격시킵니다.

승격 프로세스가 완료되는 즉시 쓰기를 재개할 수 있으며 일반적으로 몇 초 정도 소요됩니다. 애플리케이션이 기본 엔드포인트에 쓰는 경우 ElastiCache가 승격된 복제본의 DNS 이름을 전파하기 때문에 쓰기 또는 읽기에 대한 엔드포인트를 변경할 필요가 없습니다.

3. 대체 읽기 전용 복제본을 시작하고 프로비저닝합니다.

장애가 발생한 기본 노드가 있는 가용 영역에서 대체 읽기 전용 복제본을 시작하여 노드 배포를 유지합니다.

4. 복제본이 새 기본 노드와 동기화됩니다.

새 복제본을 사용할 수 있게 되면 다음 효과에 유의하십시오.

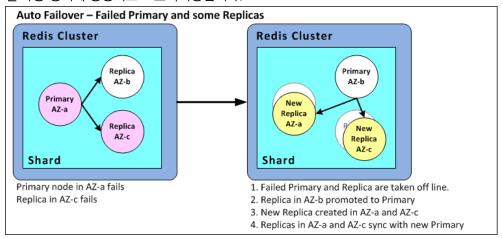
- 기본 엔드포인트 새 기본 노드의 DNS 이름이 기본 엔드포인트로 전파되므로 애플리케이션을 변경할 필요가 없습니다.
- 읽기 엔드포인트 리더 엔드포인트는 새 복제본 노드를 가리키도록 자동으로 업데이트됩니다.

클러스터의 엔드포인트를 찾는 방법에 대한 정보는 다음 항목을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 181)
- 복제 그룹의 엔드포인트 찾기 (AWS CLI) (p. 185)
- 복제 그룹의 엔드포인트 찾기 (ElastiCache API) (p. 187)

기본 노드 및 일부 읽기 전용 복제본에 장애가 발생한 경우

기본 복제본 및 하나 이상의 복제본에 장애가 발생하면 지연 시간이 가장 짧은 사용 가능한 복제본이 기본 클 러스터로 승격됩니다. 또한 기본으로 승격된 복제본 및 장애가 발생한 노드로 새로운 읽기 전용 복제본이 동 일 가용 영역에 생성되고 프로비저닝됩니다.



기본 노드와 일부 읽기 전용 복제본에 장애가 발생한 경우 ElastiCache 다중 AZ가 수행하는 작업은 다음과 같습니다.

- 1. 장애가 발생한 기본 노드 및 읽기 전용 복제본이 오프라인 상태로 전환됩니다.
- 2. 복제 지연 시간이 가장 짧은 사용 가능한 복제본을 기본 노드로 승격시킵니다.

승격 프로세스가 완료되는 즉시 쓰기를 재개할 수 있으며 일반적으로 몇 초 정도 소요됩니다. 애플리케이션이 기본 엔드포인트에 쓰는 경우 ElastiCache가 승격된 복제본의 DNS 이름을 전파하기 때문에 쓰기에대한 엔드포인트를 변경할 필요가 없습니다.

3. 교체용 복제본을 생성하고 프로비저닝합니다.

장애가 발생한 노드의 가용 영역에서 교체용 복제본을 생성하여 노드 배포를 유지합니다.

4. 모든 클러스터가 새 기본 노드와 동기화됩니다.

새 노드를 사용할 수 있게 되면 애플리케이션을 다음과 같이 변경해야 합니다.

- 기본 엔드포인트 기본 노드의 DNS 이름이 기본 엔드포인트로 전파되므로 애플리케이션을 변경하지 마십시오.
- 읽기 엔드포인트 읽기 엔드포인트는 새 복제본의 새 복제본 노드를 가리키도록 자동으로 업데이트됩니다.

복제 그룹의 엔드포인트를 찾는 방법에 대한 정보는 다음 항목을 참조하십시오.

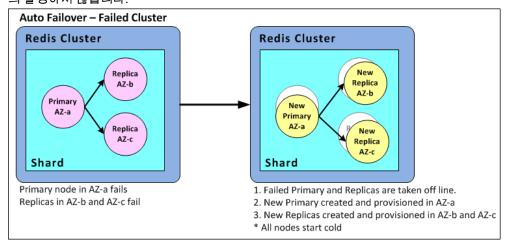
- Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 181)
- 복제 그룹의 엔드포인트 찾기 (AWS CLI) (p. 185)
- 복제 그룹의 엔드포인트 찾기 (ElastiCache API) (p. 187)

전체 클러스터에 장애가 발생한 경우

모든 것에 장애가 발생하면 모든 노드를 동일한 가용 영역에 원본 노드로 재생성하고 프로비저닝합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ

이 시나리오에서는 클러스터의 모든 노드에 장애가 발생하여 클러스터의 모든 데이터가 손실됩니다. 이는 거의 발생하지 않습니다.



전체 클러스터에 장애가 발생한 경우 ElastiCache 다중 AZ는 다음 작업을 수행합니다.

- 1. 장애가 발생한 기본 노드 및 읽기 전용 복제본이 오프라인 상태로 전환됩니다.
- 2. 대체 기본 노드를 생성하고 프로비저닝합니다.
- 3. 교체용 복제본을 생성하고 프로비저닝합니다.

장애가 발생한 노드의 가용 영역에서 대체를 생성하여 노드 배포를 유지합니다.

전체 클러스터에 장애가 발생했으므로 데이터가 손실되고 모든 새 노드가 콜드를 시작합니다.

각각의 교체 노드에는 교체하는 노드와 동일한 엔드포인트가 있기 때문에 애플리케이션에서 엔드포인트를 변경할 필요가 없습니다.

복제 그룹의 엔드포인트를 찾는 방법에 대한 정보는 다음 항목을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 181)
- 복제 그룹의 엔드포인트 찾기 (AWS CLI) (p. 185)
- 복제 그룹의 엔드포인트 찾기 (ElastiCache API) (p. 187)

내결함성 수준을 높이려면 다른 가용 영역에 기본 노드 및 읽기 전용 복제본을 생성하는 것이 좋습니다.

자동 장애 조치가 있는 다중 AZ 활성화

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 클러스터(API 또는 CLI, 복제 그룹)를 생성하거나 수정하면 자동 장애 조치가 있는 다중 AZ를 활성화할 수 있습니다.

사용할 수 있는 읽기 복제본이 하나 이상 있는 Redis(클러스터 모드 비활성화됨) 클러스터에서만 자동 장애조치가 있는 다중 AZ를 활성화할 수 있습니다. 읽기 전용 복제본이 있는지 여부와 관계없이 모든 Redis(클러스터 모드 활성화됨) 클러스터에는 자동 장애 조치가 있는 다중 AZ가 필요합니다. 읽기 전용 복제본이 없는 클러스터는 고가용성 또는 내결함성을 제공하지 않습니다. 복제하여 클러스터를 생성에 대한 정보는 Redis 복제 그룹 생성 (p. 143)을 참조하십시오. 복제하여 있는 클러스터에 읽기 전용 복제본 추가에 대한 정보는 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본 추가 (p. 176)를 참조하십시오.

주제

- 자동 장애 조치가 있는 다중 AZ 활성화 (콘솔) (p. 137)
- 자동 장애 조치가 있는 다중 AZ 활성화 (AWS CLI) (p. 137)

• 자동 장애 조치가 있는 다중 AZ 활성화 (ElastiCache API) (p. 138)

자동 장애 조치가 있는 다중 AZ 활성화 (콘솔)

새 Redis 클러스터를 생성하거나 기존 Redis 클러스터를 복제하여 수정할 때 ElastiCache 콘솔을 사용하여 자동 장애 조치가 있는 다중 AZ를 활성화할 수 있습니다.

Redis(클러스터 모드 활성화됨) 클러스터에서 자동 장애 조치가 있는 다중 AZ는 기본적으로 활성화되어 있으며 비활성화할 수 없습니다.

ElastiCache 콘솔을 사용하여 클러스터 생성 시 자동 장애 조치가 있는 다중 AZ 활성화

이 프로세스에 대한 자세한 내용은 Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69)을 참조하십시오. 복제본이 하나 이상 있어야 하고 자동 장애 조치가 있는 다중 AZ를 활성화해야 합니다.

기존 클러스터에서 자동 장애 조치가 있는 다중 AZ 활성화(콘솔)

이 프로세스에 대한 자세한 내용은 클러스터 수정 AWS Management 콘솔 사용 (p. 87)단원을 참조하십시오.

자동 장애 조치가 있는 다중 AZ 활성화 (AWS CLI)

다음 코드 예제에서는 AWS CLI를 사용하여 복제 그룹 redis12에 대해 자동 장애 조치가 있는 다중 AZ를 활성화합니다.

Important

복제 그룹 redis12가 이미 존재해야 하며 사용할 수 있는 읽기 전용 복제본이 하나 이상 있어야 합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group \
    --replication-group-id redis12 \
    --automatic-failover-enabled \
    --apply-immediately
```

Windows의 경우:

```
aws elasticache modify-replication-group ^
--replication-group-id redis12 ^
--automatic-failover-enabled ^
--apply-immediately
```

이 명령의 JSON 출력은 다음과 같아야 합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ

```
"CacheNodeId": "0001",
                        "ReadEndpoint": {
                             "Port": 6379,
                             "Address": "redis12-001.v5r9dc.0001.usw2.cache.amazonaws.com"
                        "CacheClusterId": "redis12-001"
                    },
                        "CurrentRole": "replica",
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Port": 6379,
                            "Address": "redis12-002.v5r9dc.0001.usw2.cache.amazonaws.com"
                        "CacheClusterId": "redis12-002"
                    }
                ٦,
                "NodeGroupId": "0001",
                "PrimaryEndpoint": {
                    "Port": 6379,
                    "Address": "redis12.v5r9dc.ng.0001.usw2.cache.amazonaws.com"
                }
            }
        ],
        "ReplicationGroupId": "redis12",
        "SnapshotRetentionLimit": 1,
        "AutomaticFailover": "enabling",
        "SnapshotWindow": "07:00-08:00",
        "SnapshottingClusterId": "redis12-002",
        "MemberClusters": [
            "redis12-001",
            "redis12-002"
        "PendingModifiedValues": {}
   }
}
```

자세한 내용은 AWS CLI 명령 참조의 다음 항목을 참조하십시오.

- create-cache-cluster
- · create-replication-group
- AWS CLI Command Reference의 modify-replication-group

자동 장애 조치가 있는 다중 AZ 활성화 (ElastiCache API)

다음 코드 예제에서는 ElastiCache API를 사용하여 복제 그룹 redis12에 대해 자동 장애 조치가 있는 다중 AZ를 활성화합니다.

Note

이 예제를 사용하려면 복제 그룹 redis12가 이미 존재해야 하며 사용할 수 있는 읽기 전용 복제본 이 하나 이상 있어야 합니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=ModifyReplicationGroup
    &ApplyImmediately=true
    &AutoFailover=true
    &ReplicationGroupId=redis12
    &Version=2015-02-02
    &SignatureVersion=4
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ

&SignatureMethod=HmacSHA256 &Timestamp=20140401T192317Z &X-Amz-Credential=<credential>

자세한 내용은 ElastiCache API 참조에서 다음 주제들을 참조하십시오.

- CreateCacheCluster
- CreateReplicationGroup
- ModifyReplicationGroup

자동 장애 조치가 있는 다중 AZ 테스트

자동 장애 조치가 있는 다중 AZ를 활성화한 후에는 ElastiCache 콘솔, AWS CLI 및 ElastiCache API를 사용하여 이를 테스트할 수 있습니다.

테스트 시 다음 사항에 유의하십시오.

- 이 작업을 통해 24시간 동안 최대 5개의 샤드(ElastiCache API 및 AWS CLI의 경우 노드 그룹)에서 자동 장애 조치를 테스트할 수 있습니다.
- 다른 클러스터(API 및 CLI의 복제 그룹이라고 함)에 있는 샤드에서 이 작업을 동시에 호출할 수 있습니다.
- 동일한 Redis(클러스터 모드 활성화됨) 복제 그룹의 다른 샤드에서 이 작업을 여러 번 호출하면 후속 호출을 하기 전에 첫 번째 노드 교체가 완료되어야 합니다.
- Amazon ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 이벤트를 점검하면 노드 교체가 완료되었는지 확인할 수 있습니다. 발생 순서대로 나열되어 있는 아래 목록에서 다음과 같은 자동 장애 조 치 관련 이벤트를 찾습니다.
 - 1. 복제 그룹 메시지: Test Failover API called for node group <node-group-id>
 - 2. 캐시 클러스터 메시지: Failover from master node <primary-node-id> to replica node <node-id> completed
 - 3. 복제 그룹 메시지: Failover from master node <primary-node-id> to replica node <node-id> completed
 - 4. 캐시 클러스터 메시지: Recovering cache nodes <node-id>
 - 5. 캐시 클러스터 메시지: Finished recovery for cache nodes <node-id>

자세한 내용은 다음을 참조하십시오.

- ElastiCache 이벤트 보기 (p. 370)(출처: ElastiCache 사용 설명서)
- ElastiCache API 참조의 DescribeEvents
- AWS CLI Command Reference describe-events

자동 장애 조치 테스트

- AWS Management 콘솔를 사용하여 자동 장애 조치 테스트 (p. 139)
- AWS CLI를 사용하여 자동 장애 조치 테스트 (p. 140)
- ElastiCache API를 사용하여 자동 장애 조치 테스트 (p. 142)

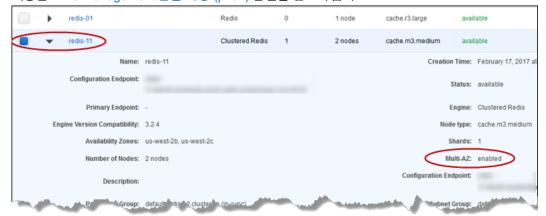
AWS Management 콘솔를 사용하여 자동 장애 조치 테스트

다음 절차에서는 자동 장애 조치를 테스트하는 과정을 안내합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ

자동 장애 조치를 테스트하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
- 3. Redis 클러스터 목록에서 테스트할 클러스터 왼쪽에 있는 확인란을 선택합니다. 이 클러스터에는 읽기 전용 복제본 노드가 하나 이상 있어야 합니다.
- 4. [Details] 영역에서 이 클러스터가 다중 AZ 활성 상태인지 확인합니다. 해당 클러스터가 다중 AZ 활성 상태가 아닌 경우 다른 클러스터를 선택하거나 다중 AZ를 활성화하도록 클러스터를 수정합니다. 자세한 내용은 AWS Management 콘솔 사용 (p. 87) 단원을 참조하십시오.



5. Redis(클러스터 모드 비활성화됨)의 경우 클러스터 이름을 선택합니다.

Redis(클러스터 모드 활성화됨)에서 다음을 수행합니다.

- a. 클러스터의 이름을 선택합니다.
- b. [Shards] 페이지에서 장애 조치를 테스트할 샤드(API 및 CLI의 노드 그룹이라고 함)에 대해 샤드 이름을 선택합니다.
- 6. 노드 페이지에서 [Failover Primary]를 선택합니다.
- 7. 기본 노드를 장애 조치하려면 [Continue]를 선택하고 작업을 취소하여 기본 노드를 장애 조치하지 않으려면 [Cancel]을 선택합니다.

장애 조치 프로세스 중에 콘솔은 노드 상태를 계속해서 사용 가능으로 표시합니다. 장애 조치 테스트 진행률을 추적하려면 콘솔 탐색 창에서 [Events]를 선택합니다. [Events] 탭에서 장애 조치의 시작(Test Failover API called) 및 완료(Recovery completed)를 나타내는 이벤트를 주시합니다.

AWS CLI를 사용하여 자동 장애 조치 테스트

AWS CLI 작업 test-failover을 사용하여 자동 장애 조치가 있는 모든 다중 AZ가 활성화된 클러스터에서 자동 장애 조치를 테스트할 수 있습니다.

파라미터

- --replication-group-id 필수 사항. 테스트할 복제 그룹(콘솔, 클러스터)입니다.
- --node-group-id 필수 사항. 자동 장애 조치를 테스트할 노드 그룹의 이름입니다. 24시간 동안 최대 5 개의 노드 그룹을 테스트할 수 있습니다.

다음 예제에서는 AWS CLI를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터 redis00의 노드 그룹 redis00-0003에 대한 자동 장애 조치를 테스트합니다.

Example 자동 장애 조치 테스트

Linux, macOS, or Unix의 경우:

```
aws elasticache test-failover \
   --replication-group-id redis00 \
   --node-group-id redis00-0003
```

Windows의 경우:

```
aws elasticache test-failover ^
--replication-group-id redis00 ^
--node-group-id redis00-0003
```

이전 명령의 출력은 다음과 같습니다.

```
{
    "ReplicationGroup": {
        "Status": "available",
        "Description": "1 shard, 3 nodes (1 + 2 replicas)",
        "NodeGroups": [
                "Status": "available",
                "NodeGroupMembers": [
                        "CurrentRole": "primary",
                        "PreferredAvailabilityZone": "us-west-2c",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Port": 6379,
                            "Address": "redis1x3-001.7ekv3t.0001.usw2.cache.amazonaws.com"
                        "CacheClusterId": "redis1x3-001"
                    },
                        "CurrentRole": "replica",
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Port": 6379,
                            "Address": "redis1x3-002.7ekv3t.0001.usw2.cache.amazonaws.com"
                        "CacheClusterId": "redis1x3-002"
                    },
                        "CurrentRole": "replica",
                        "PreferredAvailabilityZone": "us-west-2b",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Port": 6379,
                            "Address": "redis1x3-003.7ekv3t.0001.usw2.cache.amazonaws.com"
                        "CacheClusterId": "redis1x3-003"
                    }
                ],
                "NodeGroupId": "0001",
                "PrimaryEndpoint": {
                    "Port": 6379,
                    "Address": "redis1x3.7ekv3t.ng.0001.usw2.cache.amazonaws.com"
            }
        ],
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ

장애 조치 진행률을 추적하려면 AWS CLI describe-events 작업을 사용하십시오.

자세한 내용은 다음을 참조하십시오.

- AWS CLI 명령 참조의 test-failover
- AWS CLI Command Reference의 describe-events

ElastiCache API를 사용하여 자동 장애 조치 테스트

ElastiCache API 작업 TestFailover을 사용하여 자동 장애 조치가 있는 다중 AZ가 활성화된 모든 클러스터에서 자동 장애 조치를 테스트할 수 있습니다.

파라미터

- ReplicationGroupId 필수 사항. 테스트할 복제 그룹(콘솔, 클러스터)입니다.
- NodeGroupId 필수 사항. 자동 장애 조치를 테스트할 노드 그룹의 이름입니다. 24시간 동안 최대 5개의 노드 그룹을 테스트할 수 있습니다.

다음 예제에서는 복제 그룹(콘솔, 클러스터에서) redis00의 노드 그룹 redis00-0003에 대한 자동 장애 조치를 테스트합니다.

Example 자동 장애 조치 테스트

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=TestFailover
    &NodeGroupId=redis00-0003
    &ReplicationGroupId=redis00
    &Version=2015-02-02
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20140401T192317Z
    &X-Amz-Credential=<credential>
```

장애 조치 진행률을 추적하려면 ElastiCache API DescribeEvents 작업을 사용하십시오.

자세한 내용은 다음을 참조하십시오.

- ElastiCache API 참조 의 TestFailover
- ElastiCache API 참조 의 DescribeEvents

동기화 및 백업 구현 방법

지원되는 모든 버전의 Redis는 기본 클러스터와 복제본 클러스터 간의 백업 및 동기화를 지원합니다. 그러나백업 및 동기화가 구현되는 방식은 Redis 버전에 따라 다릅니다.

Redis 버전 2.8.22 이상

버전 2.8.22 이상에서 Redis 복제는 두 가지 방법 중 하나를 선택합니다. 자세한 내용은 Redis 2.8.22 이전 버전 (p. 143) 및 Redis용 ElastiCache 백업 및 복원 (p. 187) 단원을 참조하십시오.

포크 없는 프로세스 중 쓰기 로드가 많으면 변경 사항이 너무 많이 누적되어 성공적인 스냅샷을 방해하는 일이 발생하지 않도록 클러스터에 대한 쓰기가 지연됩니다.

Redis 2.8.22 이전 버전

2.8.22 이전 버전의 Redis 백업 및 동기화는 3단계 프로세스입니다.

- 1. 포크하고 백그라운드 프로세스에서 클러스터 데이터를 디스크에 직렬화합니다. 그러면 특정 시점 스냅 샷이 생성됩니다.
- 2. 포그라운드에서 클라이언트 출력 버퍼에 변경 로그를 누적합니다.

Important

변경 로그가 클라이언트 출력 버퍼 크기를 초과하면 백업 또는 동기화가 실패합니다. 자세한 내용은 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404) 단원을 참조하십시오.

마지막으로 캐시 데이터와 변경 로그를 순서대로 복제본 클러스터에 전송합니다.

Redis 복제 그룹 생성

복제본 노드가 있는 클러스터를 생성하기 위한 다음과 같은 옵션이 있습니다. 한 옵션은 기본 노드로 사용되는 복제본이 있는 클러스터와 연결이 안된 사용 가능한 Redis(클러스터 모드 비활성화됨) 클러스터가 이미있을 때 적용됩니다. 다른 옵션은 클러스터와 읽기 전용 복제본으로 기본 노드를 생성해야할 때 적용됩니다. 현재는 Redis(클러스터 모드 활성화됨) 클러스터를 처음부터 생성해야 합니다.

옵션 1: 가용 Redis(클러스터 모드 비활성화됨) 클러스터를 사용하여 복제 그룹 생성 (p. 144)

기존 단일 노드 Redis(클러스터 모드 비활성화됨) 클러스터를 활용하려면 이 옵션을 사용합니다. 이 기존 노드를 새 클러스터의 기본 노드로 지정하고 클러스터에 1개~5개의 읽기 전용 복제본을 개별적으로 추가합니다. 기존 클러스터가 활성 상태인 경우 읽기 복제본은 생성되는 대로 해당 클러스터와 동기화됩니다. 가용 Redis(클러스터 모드 비활성화됨) 클러스터를 사용하여 복제 그룹 생성 (p. 144) 단원을 참조하십시오.

Important

기존 클러스터를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터를 생성할 수 없습니다. ElastiCache 콘솔을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터(API/CLI의 경우 복제 그룹)를 생성하려면 Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 72) 단원을 참 조하십시오.

옵션 2: Redis 복제 그룹을 처음부터 새로 생성 (p. 148)

클러스터의 기본으로 사용하는 가용 Redis(클러스터 모드 비활성화됨) 클러스터가 없거나 Redis(클러스터 모드 활성화됨) 클러스터를 생성하고 싶은 경우 이 옵션을 사용합니다. Redis 복제 그룹을 처음부터 새로 생성 (p. 148) 단원을 참조하십시오.

가용 Redis(클러스터 모드 비활성화됨) 클러스터를 사용하여 복제 그룹 생성

사용 가능한 클러스터는 기존 단일 노드 Redis 클러스터입니다. 현재 Redis(클러스터 모드 활성화됨)에서는 사용 가능한 단일 노드 클러스터를 사용하여 복제본이 있는 클러스터를 생성할 수 없습니다. Redis(클러스터 모드 활성화됨) 클러스터를 생성하려면 Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 154) 단원을 참조하십시오.

다음 절차는 Redis(클러스터 모드 비활성화됨) 단일 노드 클러스터가 있는 경우에만 사용할 수 있습니다. 이 클러스터의 노드는 새 클러스터의 기본 노드가 됩니다. 새 클러스터의 기본 노드로 사용할 수 있는 Redis 복제 그룹을 처음부터 새로 생성 (p. 148) 클러스터가 없는 경우 Redis(클러스터 모드 비활성화됨) 단원을 참조하십시오.

사용 가능한 Redis 클러스터 (콘솔)을(를) 사용하여 복제 그룹 생성

AWS Management 콘솔 사용 (p. 91) 항목을 참조하십시오.

사용 가능한 Redis 캐시 클러스터(AWS CLI)를 사용하여 복제 그룹 생성

AWS CLI를 사용할 때 기본 노드에 대해 사용 가능한 Redis 캐시 클러스터를 사용하는 경우 읽기 전용 복제 본이 있는 복제 그룹을 생성하는 두 단계가 있습니다.

AWS CLI를 사용하는 경우 사용 가능한 독립형 노드를 클러스터의 기본 노드인 --primary-cluster-id로 지정하고, CLI 명령 create-replication-group을 사용해 클러스터에 필요한 노드 수를 지정해 복제 그룹을 생성합니다. 다음 파라미터를 포함합니다.

--replication-group-id

생성하는 복제 그룹의 이름입니다. 이 파라미터의 값은 추가되는 노드의 이름을 지정하는 기준으로 사용되는데, --replication-group-id 끝에 3자리 일련 번호가 추가됩니다. 예: sample-replgroup-001.

Redis(클러스터 모드 비활성화됨) 복제 그룹 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

--replication-group-description

복제 그룹에 대한 설명입니다.

--num-cache-clusters

이 클러스터에 있는 노드의 수. 이 값에는 기본 노드가 포함됩니다. 이 파라미터의 최대값은 six입니다. --primary-cluster-id

이 복제 그룹의 기본 노드가 될 사용 가능한 Redis(클러스터 모드 비활성화됨) 클러스터 노드의 이름입니다.

이 복제 그룹에서 전송 중 데이터 암호화 또는 미사용 데이터 암호화를 활성화하려면 --trasitencryption-enabled 또는 --at-rest-encryption-enabled 파라미터 중 하나 또는 둘 다를 추가하고 다음 조건을 충족해야 합니다.

- 클러스터에서 3.2.6 또는 4.0.10 버전 Redis를 실행하고 있어야 합니다.
- 복제 그룹은 Amazon VPC에 생성되어야 합니다.

- 또한 --cache-subnet-group 파라미터도 포함해야 합니다.
- 또한 이 복제 그룹에서 작업을 수행하는 데 필요한 AUTH 토큰(암호)에 고객이 지정한 문자열 값이 있는 --auth-token 파라미터도 포함해야 합니다.

다음 명령은 사용 가능한 Redis(클러스터 모드 비활성화됨) 클러스터 redis01을 복제 그룹의 기본 노드로 사용해 복제 그룹 sample-repl-group을 생성합니다. 이렇게 하면 읽기 전용 복제본인 새 노드 2개가 생성됩니다. redis01의 설정(즉, 파라미터 그룹, 보안 그룹, 노드 유형, 엔진 버전 등)은 복제 그룹의 모든 노드에 적용됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-replication-group \
--replication-group-id sample-repl-group \
--replication-group-description "demo cluster with replicas" \
--num-cache-clusters 3 \
--primary-cluster-id redis01
```

Windows의 경우:

```
aws elasticache create-replication-group ^
--replication-group-id sample-repl-group ^
--replication-group-description "demo cluster with replicas" ^
--num-cache-clusters 3 ^
--primary-cluster-id rediso1
```

사용할 파라미터에 대한 자세한 내용은 AWS CLI 항목 create-replication-group를 참조하십시오.

다음으로 복제 그룹에 읽기 전용 복제본을 추가합니다.

복제 그룹이 생성된 후 create-cache-cluster 명령을 사용하여 해당 복제 그룹에 1~5개의 읽기 전용 복 제본을 추가하여 다음 파라미터를 포함해야 합니다.

--cache-cluster-id

복제 그룹에 추가하는 클러스터의 이름입니다.

클러스터 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이폰 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

--replication-group-id

이 캐시 클러스터를 추가하는 복제 그룹의 이름입니다.

--cache-cluster-id 파라미터 값만 변경하여 복제 그룹에 추가할 각 읽기 전용 복제본마다 이 명령을 반 복합니다.

Note

복제 그룹에는 읽기 전용 복제본이 최대 5개로 제한됩니다. 읽기 전용 복제본 5개가 이미 있는 복제 그룹에 읽기 전용 복제본을 추가하려고 하면 작업이 실패합니다.

다음 코드는 읽기 전용 복제본 my-replica01을 복제 그룹 sample-repl-group에 추가합니다. 기본 클러스터의 설정(즉, –파라미터 그룹, 보안 그룹, 노드 유형 등–)은 복제 그룹에 적용될 때 노드에도 적용됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-cache-cluster \
   --cache-cluster-id my-replica01 \
   --replication-group-id sample-repl-group
```

Windows의 경우:

```
aws elasticache create-cache-cluster ^
--cache-cluster-id my-replica01 ^
--replication-group-id sample-repl-group
```

이 명령의 출력은 다음과 같습니다.

```
"ReplicationGroup": {
        "Status": "creating",
        "Description": "demo cluster with replicas",
        "ClusterEnabled": false,
        "ReplicationGroupId": "sample-repl-group",
        "SnapshotRetentionLimit": 1,
        "AutomaticFailover": "disabled",
        "SnapshotWindow": "00:00-01:00",
        "SnapshottingClusterId": "redis01",
        "MemberClusters": [
            "sample-repl-group-001",
            "sample-repl-group-002",
            "redis01"
        ],
        "CacheNodeType": "cache.m4.large",
        "PendingModifiedValues": {}
   }
}
```

자세한 내용은 AWS CLI 주제를 참조하십시오.

- · create-replication-group
- · modify-replication-group

독립형 Redis(클러스터 모드 비활성화됨) 클러스터에 복제본 추가 (ElastiCache API)

ElastiCache API를 사용하는 경우 사용 가능한 독립형 노드를 클러스터의 기본 노드인 PrimaryClusterId로 지정하고, CLI 명령 CreateReplicationGroup을 사용해 클러스터에 필요한 노 드 수를 지정해 복제 그룹을 생성합니다. 다음 파라미터를 포함합니다.

ReplicationGroupId

생성하는 복제 그룹의 이름입니다. 이 파라미터의 값은 추가되는 노드의 이름을 지정하는 기준으로 사용 되는데, ReplicationGroupId 끝에 3자리 일련 번호가 추가됩니다. 예: sample-repl-group-001.

Redis(클러스터 모드 비활성화됨) 복제 그룹 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

ReplicationGroupDescription

복제본이 있는 클러스터에 대한 설명입니다.

NumCacheClusters

이 클러스터에 있는 노드의 수. 이 값에는 기본 노드가 포함됩니다. 이 파라미터의 최대값은 six입니다. PrimaryClusterId

이 클러스터의 기본 노드가 될 사용 가능한 Redis(클러스터 모드 비활성화됨) 클러스터의 이름입니다.

이 복제 그룹에서 전송 중 데이터 암호화 또는 미사용 데이터 암호화를 활성화하려면 TransitEncryptionEnabled=true 또는 AtRestEncryptionEnabled=true 파라미터 중 하나 또는 둘 다를 추가하고 다음 조건을 충족해야 합니다.

- 클러스터에서 3.2.6, 4.0.10 또는 그 이상의 Redis 버전을 실행하고 있어야 합니다.
- 복제 그룹은 Amazon VPC에 생성되어야 합니다.
- 또한 CacheSubnetGroup 파라미터도 포함해야 합니다.
- 또한 이 복제 그룹에서 작업을 수행하는 데 필요한 AUTH 토큰(암호)에 고객이 지정한 문자열 값이 있는 AuthToken 파라미터도 포함해야 합니다.

다음 명령은 사용 가능한 Redis(클러스터 모드 비활성화됨) 클러스터 redis01을 복제 그룹의 기본 노드로 사용해 복제본 sample-repl-group으로 클러스터를 생성합니다. 이렇게 하면 읽기 전용 복제본인 새 노드 2개가 생성됩니다. redis01의 설정(즉, 파라미터 그룹, 보안 그룹, 노드 유형, 엔진 버전 등)은 복제 그룹의 모든 노드에 적용됩니다.

https://elasticache.us-west-2.amazonaws.com/
 ?Action=CreateReplicationGroup
 &Engine=redis
 &EngineVersion=3.2.4
 &ReplicationGroupDescription=Demo%20cluster%20with%20replicas
 &ReplicationGroupId=sample-repl-group
 &PrimaryClusterId=redis01
 &Version=2015-02-02
 &SignatureVersion=4
 &SignatureMethod=HmacSHA256
 &Timestamp=20150202T192317Z
 &X-Amz-Credential=<credential>

자세한 내용은 ElastiCache APL 주제를 참조하십시오.

- CreateReplicationGroup
- ModifyReplicationGroup

다음으로 복제 그룹에 읽기 전용 복제본을 추가합니다.

복제 그룹이 생성된 후 CreateCacheCluster 작업을 사용하여 해당 복제 그룹에 1~5개의 읽기 전용 복제 본을 추가하여 다음 파라미터를 포함해야 합니다.

CacheClusterId

복제 그룹에 추가하는 클러스터의 이름입니다.

클러스터 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.

• 끝에 하이픈이 올 수 없습니다.

ReplicationGroupId

이 캐시 클러스터를 추가하는 복제 그룹의 이름입니다.

CacheClusterId 파라미터 값만 변경하여 복제 그룹에 추가할 각 읽기 전용 복제본마다 이 작업을 반복합니다.

다음 코드는 읽기 전용 복제본 myReplica01을 복제 그룹 myReplGroup에 추가합니다. 기본 클러스터의 설정(-즉, 파라미터 그룹, 보안 그룹, 노드 유형 등-)은 복제 그룹에 적용될 때 노드에도 적용됩니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=CreateCacheCluster
&CacheClusterId=myReplica01
&ReplicationGroupId=myReplGroup
&SignatureMethod=HmacSHA256
&SignatureVersion=4
&Version=2015-02-02
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=[your-access-key-id]/20150202/us-west-2/elasticache/aws4_request
&X-Amz-Date=20150202T170651Z
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
&X-Amz-Signature=[signature-value]

사용할 파라미터에 대한 자세한 내용은 ElastiCache API 항목 CreateCacheCluster를 참조하십시오.

Redis 복제 그룹을 처음부터 새로 생성

이 항목에서는 기존 Redis 클러스터를 기본으로 사용하지 않고 Redis 복제 그룹을 생성하는 방법에 대해 설명합니다. ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 비활성화 됨) 또는 Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 생성할 수 있습니다.

계속하기 전에 Redis(클러스터 모드 비활성화됨)을 생성할지 아니면 Redis(클러스터 모드 활성화됨) 복제 그룹을 생성할지를 결정합니다. 결정에 대한 지침은 복제: Redis(클러스터 모드 비활성화됨) 대 Redis(클러스터 모드 활성화됨) (p. 130)를 참조하십시오.

주제

- Redis(클러스터 모드 비활성화됨) Redis 복제 그룹을 처음부터 새로 생성 (p. 148)
- Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (p. 153)

Redis(클러스터 모드 비활성화됨) Redis 복제 그룹을 처음부터 새로 생성

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 생성할 수 있습니다. Redis(클러스터 모드 비활성화됨) 복제 그룹에는 항상 하나의 노드 그룹, 하나의 기본 클러스터 및 최대 5개의 읽기 전용 복제본이 있습니다. 이 구성은 90개의 샤드 및 0개의 복제본부터 15개의 샤드 및 5개의 복제본까지 해당될 수 있으며, 이는 허용되는 최대 복제본 수입니다. Redis(클러스터 모드 비활성화됨) 복제 그룹은 데이터 파티셔닝을 지원하지 않습니다.

Note

노드/샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 제한을 높이도록 요청하려면 AWS 서비스 제한을 참조하고 요청에 인스턴스 유형을 포함하십시오.

Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성

• Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69) 6.i단계에서 복제본 노드를 하나 이상 지정합니다.

- Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 149)
- Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성 (p. 151)

Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI)

다음 절차에서는 AWS CLI를 사용하여 Redis(클러스터 모드 비활성화됨) 복제 그룹을 생성합니다.

Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 생성할 때 AWS CLI create-replicationgroup 명령을 한 번만 호출하여 복제 그룹과 해당 노드를 모두 생성합니다. 다음 파라미터를 포함합니다.

--replication-group-id

생성하는 복제 그룹의 이름입니다.

Redis(클러스터 모드 비활성화됨) 복제 그룹 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

--replication-group-description

복제 그룹에 대한 설명입니다.

--num-cache-clusters

이 복제 그룹, 기본 및 읽기 전용 복제본과 함께 생성하려는 노드의 수입니다.

다중 AZ(--automatic-failover-enabled)를 활성화하는 경우 --num-cache-clusters의 값은 2 이상이어야 합니다.

--cache-node-type

복제 그룹에 있는 각 노드의 노드 유형입니다.

ElastiCache는 다음 노드 유형을 지원합니다. 일반적으로, 현재 세대 유형은 이전 세대의 동급 제품에 비해 더 많은 메모리와 컴퓨팅 파워를 더 저렴하게 제공합니다.

- 범용:
 - 현재 세대:

M5 노드 유형: cache.m5.large, cache.m5.xlarge, cache.m5.2xlarge, cache.m5.4xlarge, cache.m5.12xlarge, cache.m5.24xlarge

M4 노드 유형: cache.m4.large, cache.m4.xlarge, cache.m4.2xlarge, cache.m4.4xlarge, cache.m4.10xlarge

T2 노드 유형: cache.t2.micro, cache.t2.small, cache.t2.medium

• 이전 세대: (권장하지 않음)

T1 노드 유형: cache.t1.micro

M1 노드 유형: cache.ml.small, cache.ml.medium, cache.ml.large, cache.ml.xlarge

M3 노드 유형: cache.m3.medium, cache.m3.large, cache.m3.xlarge, cache.m3.2xlarge

- 컴퓨팅 최적화:
 - 이전 세대: (권장하지 않음)

C1 노드 유형: cache.c1.xlarge

- 메모리 최적화:
 - 현재 세대:

R5 노드 유형: cache.r5.large, cache.r5.xlarge, cache.r5.2xlarge, cache.r5.4xlarge, cache.r5.12xlarge, cache.r5.24xlarge

R4 노드 유형: cache.r4.large, cache.r4.xlarge, cache.r4.2xlarge, cache.r4.4xlarge, cache.r4.8xlarge, cache.r4.16xlarge

• 이전 세대: (권장하지 않음)

M2 노드 유형: cache.m2.xlarge, cache.m2.2xlarge, cache.m2.4xlarge

R3 노드 유형: cache.r3.large, cache.r3.xlarge, cache.r3.2xlarge, cache.r3.4xlarge, cache.r3.8xlarge

추가 노드 유형 정보

- 현재 세대의 모든 인스턴스 유형은 기본적으로 Amazon VPC에서 생성됩니다.
- Redis AOF(append-only files)는 T1 또는 T2 인스턴스에서 지원되지 않습니다.
- Redis 다중 AZ 및 자동 장애 조치는 T1 인스턴스에서 지원되지 않습니다.
- Redis 구성 변수 appendonly 및 appendfsync는 Redis 버전 2.8.22 이상에서 지원되지 않습니다.

--cache-parameter-group

엔진 버전에 해당하는 파라미터 그룹을 지정합니다. Redis 3.2.4 이상을 실행하는 경우 default.redis3.2 파라미터 그룹 또는 default.redis3.2에서 파생된 파라미터 그룹을 지정하여 Redis(클러스터 모드 비활성화됨) 복제 그룹을 생성합니다. 자세한 내용은 Redis 특정 파라미터 (p. 274) 단원을 참조하십시오.

--에지

redis

--engine-version

다양한 기능 세트를 사용하려면 최신 엔진 버전을 선택합니다.

-00#을 복제 그룹 이름 뒤에 붙이면 복제 그룹 이름에서 노드 이름이 파생됩니다. 예를 들어, 복제 그룹 이름 myReplGroup을 사용하는 경우 기본 이름은 myReplGroup-001이 되고, 읽기 전용 복제본 이름은 myReplGroup-002에서 myReplGroup-006 사이가 됩니다.

이 복제 그룹에서 전송 중 데이터 암호화 또는 미사용 데이터 암호화를 활성화하려면 --trasitencryption-enabled 또는 --at-rest-encryption-enabled 파라미터 중 하나 또는 둘 다를 추가하 고 다음 조건을 충족해야 합니다.

- 복제 그룹에서 3.2.6 또는 4.0.10 버전 Redis를 실행하고 있어야 합니다.
- 복제 그룹은 Amazon VPC에 생성되어야 합니다.
- 또한 --cache-subnet-group 파라미터도 포함해야 합니다.
- 또한 이 복제 그룹에서 작업을 수행하는 데 필요한 AUTH 토큰(암호)에 고객이 지정한 문자열 값이 있는 --auth-token 파라미터도 포함해야 합니다.

다음 작업은 세 개의 노드(기본 한 개와 복제본 두 개)가 있는 Redis(클러스터 모드 비활성화됨) 복제 그룹 sample-repl-group을 생성합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-replication-group \
--replication-group-id sample-repl-group \
--replication-group-description "Demo cluster with replicas" \
--num-cache-clusters 3 \
--cache-node-type cache.m4.large \
--cache-parameter-group default.redis3.2 \
--engine redis \
--engine-version 3.2.4
```

Windows의 경우:

```
aws elasticache create-replication-group ^
--replication-group-id sample-repl-group ^
--replication-group-description "Demo cluster with replicas" ^
--num-cache-clusters 3 ^
--cache-node-type cache.m4.large ^
--cache-parameter-group default.redis3.2 ^
--engine redis ^
--engine-version 3.2.4
```

이 명령의 출력은 다음과 같습니다.

```
{
    "ReplicationGroup": {
       "Status": "creating",
        "Description": "Demo cluster with replicas",
        "ClusterEnabled": false,
        "ReplicationGroupId": "sample-repl-group",
        "SnapshotRetentionLimit": 0,
        "AutomaticFailover": "disabled",
        "SnapshotWindow": "01:30-02:30",
        "MemberClusters": [
            "sample-repl-group-001",
            "sample-repl-group-002",
            "sample-repl-group-003"
        "CacheNodeType": "cache.m4.large",
        "PendingModifiedValues": {}
   }
}
```

사용하려는 파라미터에 대한 자세한 내용은 AWS CLI 항목 create-replication-group를 참조하십시오.

Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성

다음 절차에서는 ElastiCache API를 사용하여 Redis(클러스터 모드 비활성화됨) 복제 그룹을 생성합니다.

Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 생성할 때 ElastiCache API CreateReplicationGroup 명령을 한 번만 호출하여 복제 그룹과 해당 노드를 모두 생성합니다. 다음 파 라미터를 포함합니다.

ReplicationGroupId

생성하는 복제 그룹의 이름입니다.

Redis(클러스터 모드 활성화됨) 복제 그룹 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.

• 끝에 하이픈이 올 수 없습니다.

ReplicationGroupDescription

복제 그룹에 대한 설명입니다.

NumCacheClusters

이 복제 그룹, 기본 및 읽기 전용 복제본과 함께 생성하려는 총 노드 수입니다.

다중 AZ(AutomaticFailoverEnabled=true)를 활성화하는 경우 NumCacheClusters의 값은 2 이상이어야 합니다.

CacheNodeType

복제 그룹에 있는 각 노드의 노드 유형입니다.

ElastiCache는 다음 노드 유형을 지원합니다. 일반적으로, 현재 세대 유형은 이전 세대의 동급 제품에 비해 더 많은 메모리와 컴퓨팅 파워를 더 저렴하게 제공합니다.

- 범용:
 - 현재 세대:

M5 노드 유형: cache.m5.large, cache.m5.xlarge, cache.m5.2xlarge, cache.m5.4xlarge, cache.m5.12xlarge, cache.m5.24xlarge

M4 노드 유형: cache.m4.large, cache.m4.xlarge, cache.m4.2xlarge, cache.m4.4xlarge, cache.m4.10xlarge

T2 노드 유형: cache.t2.micro, cache.t2.small, cache.t2.medium

• 이전 세대: (권장하지 않음)

T1 노드 유형: cache.t1.micro

M1 노드 유형: cache.ml.small, cache.ml.medium, cache.ml.large, cache.ml.xlarge

M3 노드 유형: cache.m3.medium, cache.m3.large, cache.m3.xlarge, cache.m3.2xlarge

- 컴퓨팅 최적화:
 - 이전 세대: (권장하지 않음)

C1 노드 유형: cache.c1.xlarge

- 메모리 최적화:
 - 현재 세대:

R5 노드 유형: cache.r5.large, cache.r5.xlarge, cache.r5.2xlarge, cache.r5.4xlarge, cache.r5.12xlarge, cache.r5.24xlarge

R4 노드 유형: cache.r4.large, cache.r4.xlarge, cache.r4.2xlarge, cache.r4.4xlarge, cache.r4.8xlarge, cache.r4.16xlarge

• 이전 세대: (권장하지 않음)

M2 노드 유형: cache.m2.xlarge, cache.m2.2xlarge, cache.m2.4xlarge

R3 노드 유형: cache.r3.large, cache.r3.xlarge, cache.r3.2xlarge, cache.r3.4xlarge, cache.r3.8xlarge

추가 노드 유형 정보

- 현재 세대의 모든 인스턴스 유형은 기본적으로 Amazon VPC에서 생성됩니다.
- Redis AOF(append-only files)는 T1 또는 T2 인스턴스에서 지원되지 않습니다.

- Redis 다중 AZ 및 자동 장애 조치는 T1 인스턴스에서 지원되지 않습니다.
- Redis 구성 변수 appendonly 및 appendfsync는 Redis 버전 2.8.22 이상에서 지원되지 않습니다.

CacheParameterGroup

엔진 버전에 해당하는 파라미터 그룹을 지정합니다. Redis 3.2.4 이상을 실행하는 경우 default.redis3.2 파라미터 그룹 또는 default.redis3.2에서 파생된 파라미터 그룹을 지정하여 Redis(클러스터 모드 비활성화됨) 복제 그룹을 생성합니다. 자세한 내용은 Redis 특정 파라미터 (p. 274) 단원을 참조하십시오.

에진

redis

EngineVersion

3.2.4

-00#을 복제 그룹 이름 뒤에 붙이면 복제 그룹 이름에서 노드 이름이 파생됩니다. 예를 들어, 복제 그룹 이름 myRep1Group을 사용하는 경우 기본 이름은 myRep1Group-001이 되고, 읽기 전용 복제본 이름은 myRep1Group-002에서 myRep1Group-006 사이가 됩니다.

이 복제 그룹에서 전송 중 데이터 암호화 또는 미사용 데이터 암호화를 활성화하려면 TrasitEncryptionEnabled=true 또는 AtRestEncryptionEnabled=true 파라미터 중 하나 또는 둘 다를 추가하고 다음 조건을 충족해야 합니다.

- 복제 그룹에서 3.2.6 또는 4.0.10 버전 Redis를 실행하고 있어야 합니다.
- 복제 그룹은 Amazon VPC에 생성되어야 합니다.
- 또한 CacheSubnetGroup 파라미터도 포함해야 합니다.
- 또한 이 복제 그룹에서 작업을 수행하는 데 필요한 AUTH 토큰(암호)에 고객이 지정한 문자열 값이 있는 AuthToken 파라미터도 포함해야 합니다.

다음 작업은 세 개의 노드(기본 한 개와 복제본 두 개)가 있는 Redis(클러스터 모드 비활성화됨) 복제 그룹 myRep1Group을 생성합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=CreateReplicationGroup
&CacheNodeType=cache.m4.large
&CacheParameterGroup=default.redis3.2
&Engine=redis
&EngineVersion=3.2.4
&NumCacheClusters=3
&ReplicationGroupDescription=test%20group
&ReplicationGroupId=myReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureWethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

사용할 파라미터에 대한 자세한 내용은 ElastiCache API 주제 CreateReplicationGroup를 참조하십시오.

Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성

ElastiCache콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터(API/CLI: 복제 그룹)를 생성할 수 있습니다. Redis(클러스터 모드 활성화됨) 복제 그룹에는 1 ~ 90개의 샤드(API/CLI의 경우 노드 그룹), 각 샤드의 기본 클러스터 1개 및 각 샤드의 최대 5개의 읽기 전용 복제본이 있습니다.하나의 클러스터당 최대 90개의 노드로 구성된 더 많은 수의 샤드와 더 적은 수의 복제본을 가진 클러스터를

생성할 수 있습니다. 이 클러스터 구성은 90개의 샤드 및 0개의 복제본부터 15개의 샤드 및 5개의 복제본까지 해당될 수 있으며, 이는 허용되는 최대 복제본 수입니다.

Note

노드 또는 샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 한도 증가를 요청하는 방법에 대한 지침은 AWS 서비스 제한을 참조하고 한도 유형을 '인스턴스 유형별 클러스터당 노드'로 선택하십시오.

Redis(클러스터 모드 활성화됨)에서 클러스터 생성

- Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 154)
- Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154)
- Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158)

Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔)

Redis(클러스터 모드 활성화됨) 클러스터를 생성하려면 Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 72) 단원을 참조하십시오. Cluster Mode enabled (Scale Out)(클러스터 모드 활성화(스케일 아웃))에서 클러스터 모드를 활성화하고 두 개 이상의 샤드와 한 개의 복제본 노드를 지정합니다.

Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI)

다음 절차에서는 AWS CLI를 사용하여 Redis(클러스터 모드 활성화됨) 복제 그룹을 생성합니다.

Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 생성할 때 AWS CLI create-replication-group 명령을 한 번만 호출하여 복제 그룹과 해당 노드를 모두 생성합니다. 다음 파라미터를 포함합니다.

--replication-group-id

생성하는 복제 그룹의 이름입니다.

Redis(클러스터 모드 활성화됨) 복제 그룹 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

--replication-group-description

복제 그룹에 대한 설명입니다.

--cache-node-type

복제 그룹에 있는 각 노드의 노드 유형입니다.

ElastiCache는 다음 노드 유형을 지원합니다. 일반적으로, 현재 세대 유형은 이전 세대의 동급 제품에 비해 더 많은 메모리와 컴퓨팅 파워를 더 저렴하게 제공합니다.

- 범용:
 - 현재 세대:

M5 노드 유형: cache.m5.large, cache.m5.xlarge, cache.m5.2xlarge, cache.m5.4xlarge, cache.m5.12xlarge, cache.m5.24xlarge

M4 노드 유형: cache.m4.large, cache.m4.xlarge, cache.m4.2xlarge, cache.m4.4xlarge, cache.m4.10xlarge

T2 노드 유형: cache.t2.micro, cache.t2.small, cache.t2.medium

• 이전 세대: (권장하지 않음)

T1 노드 유형: cache.t1.micro

M1 노드 유형: cache.m1.small, cache.m1.medium, cache.m1.large, cache.m1.xlarge

M3 노드 유형: cache.m3.medium, cache.m3.large, cache.m3.xlarge, cache.m3.2xlarge

- 컴퓨팅 최적화:
 - 이전 세대: (권장하지 않음)

C1 노드 유형: cache.c1.xlarge

- 메모리 최적화:
 - 현재 세대:

R5 노드 유형: cache.r5.large, cache.r5.xlarge, cache.r5.2xlarge, cache.r5.4xlarge, cache.r5.12xlarge, cache.r5.24xlarge

R4 노드 유형: cache.r4.large, cache.r4.xlarge, cache.r4.2xlarge, cache.r4.4xlarge, cache.r4.8xlarge, cache.r4.16xlarge

• 이전 세대: (권장하지 않음)

M2 노드 유형: cache.m2.xlarge, cache.m2.2xlarge, cache.m2.4xlarge

R3 노드 유형: cache.r3.large, cache.r3.xlarge, cache.r3.2xlarge, cache.r3.4xlarge, cache.r3.8xlarge

추가 노드 유형 정보

- 현재 세대의 모든 인스턴스 유형은 기본적으로 Amazon VPC에서 생성됩니다.
- Redis AOF(append-only files)는 T1 또는 T2 인스턴스에서 지원되지 않습니다.
- Redis 다중 AZ 및 자동 장애 조치는 T1 인스턴스에서 지원되지 않습니다.
- Redis 구성 변수 appendonly 및 appendfsync는 Redis 버전 2.8.22 이상에서 지원되지 않습니다.

--cache-parameter-group

default.redis3.2.cluster.on 파라미터 그룹 또는 default.redis3.2.cluster.on에서 파생된 파라미터 그룹을 지정하여 Redis(클러스터 모드 활성화됨) 복제 그룹을 생성합니다. 자세한 내용은 Redis 3.2.4 파라미터 변경 사항 (p. 282) 단원을 참조하십시오.

--엔진

redis

--engine-version

3.2.4

- --num-node-groups
 - 이 복제 그룹의 노드 그룹 수입니다. 유효한 값은 1 ~ 90입니다.

Note

노드/샤드 한도는 클러스터당 최대 250개로 늘릴 수 있습니다. 한도 증가를 요청하는 방법에 대한 지침은 AWS 서비스 제한을 참조하고 한도 유형을 '인스턴스 유형별 클러스터당 노드'로 선택하십시오.

--replicas-per-node-group

각 노드 그룹의 복제본 노드 수입니다. 유효한 값은 0~5입니다.

이 복제 그룹에서 전송 중 데이터 암호화 또는 미사용 데이터 암호화를 활성화하려면 --transit-encryption-enabled 또는 --at-rest-encryption-enabled 파라미터 중 하나 또는 둘 다를 추가하고 다음 조건을 충족해야 합니다.

- 복제 그룹에서 3.2.6 또는 4.0.10 버전 Redis를 실행하고 있어야 합니다.
- 복제 그룹은 Amazon VPC에 생성되어야 합니다.
- 또한 --cache-subnet-group 파라미터도 포함해야 합니다.
- 또한 이 복제 그룹에서 작업을 수행하는 데 필요한 AUTH 토큰(암호)에 고객이 지정한 문자열 값이 있는 --auth-token 파라미터도 포함해야 합니다.

다음 작업은 세 개의 노드 그룹/샤드(--num-node-groups)가 있는 Redis(클러스터 모드 활성화됨) 복제 그룹 sample-repl-group을 생성합니다. 이 복제 그룹은 각각 3개의 노드, 즉 기본 노드 1개와 읽기 전용 복제본 2개(--replicas-per-node-group)로 구성됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-replication-group \
--replication-group-id sample-repl-group \
--replication-group-description "Demo cluster with replicas" \
--num-node-groups 3 \
--replicas-per-node-group 2 \
--cache-node-type cache.m4.large \
--cache-parameter-group default.redis3.2.cluster.on \
--engine redis \
--engine-version 3.2.4
--security-group-ids SECURITY_GROUP_ID
--cache-subnet-group-name SUBNET_GROUP_NAME>
```

Windows의 경우:

```
aws elasticache create-replication-group ^
    --replication-group-id sample-repl-group ^
    --replication-group-description "Demo cluster with replicas" ^
    --num-node-groups 3 ^
    --replicas-per-node-group 2 ^
    --cache-node-type cache.m4.large ^
    --cache-parameter-group default.redis3.2.cluster.on ^
    --engine redis ^
    --engine-version 3.2.4
    --security-group-ids SECURITY_GROUP_ID
    --cache-subnet-group-name SUBNET_GROUP_NAME>
```

앞에 나온 명령은 다음 출력을 생성합니다.

Redis(클러스터 모드 활성화됨) 복제본 그룹을 처음부터 생성할 때 2개의 노드 그룹(콘솔의 경우 샤드)을 구성하는 다음 예제와 같이 --node-group-configuration 파라미터를 사용하여 클러스터의 각 샤드를 구성할 수 있습니다. 첫 번째 샤드에는 2개의 노드(기본 1개, 읽기 전용 복제본 1개)가 있습니다. 두 번째 샤드에는 세 개의 노드(기본 한 개와 읽기 전용 복제본 두 개)가 있습니다.

--node-group-configuration

각 노드 그룹의 구성입니다. --node-group-configuration 파라미터는 다음 필드로 구성됩니다.

• PrimaryAvailabilityZone – 이 노드 그룹의 기본 노드가 있는 가용 영역입니다. 이 파라미터가 생략되면 ElastiCache는 기본 노드의 가용 영역을 선택합니다.

예: us-west-2a.

• ReplicaAvailabilityZones – 읽기 전용 복제본이 있는 가용 영역의 쉼표로 구분된 목록입니다. 이 목록의 가용 영역 수는 ReplicaCount 값과 일치해야 합니다. 이 파라미터가 생략되면 ElastiCache는 복제본 노드의 가용 영역을 선택합니다.

예: "us-west-2a,us-west-2b,us-west-2c"

- ReplicaCount 이 노드 그룹의 복제본 노드 수입니다.
- Slots 노드 그룹의 키스페이스를 지정하는 문자열입니다. 문자열 형식은 startKey-endKey입니다. 이 파라미터가 생략되면 ElastiCache는 키를 노드 그룹 간에 균등하게 할당합니다.

여: "0-4999"

다음 작업은 2개의 노드 그룹/샤드(--num-node-groups)이 있는 Redis(클러스터 모드 활성화됨) 복제 그룹 new-group을 생성합니다. 위 예제와 달리 각 노드 그룹은 다른 노드 그룹(--node-group-configuration)과 다르게 구성됩니다.

Linux, macOS, or Unix의 경우:

Windows의 경우:

앞에 나온 작업은 다음 출력을 생성합니다.

```
"ReplicationGroup": {
        "Status": "creating",
        "Description": "Sharded replication group",
        "ReplicationGroupId": "rc-rg",
        "SnapshotRetentionLimit": 8,
        "AutomaticFailover": "enabled",
        "SnapshotWindow": "10:00-11:00",
        "MemberClusters": [
            "rc-rq-0001-001",
            "rc-rg-0001-002",
            "rc-rg-0002-001",
            "rc-rq-0002-002",
            "rc-rg-0002-003"
        "PendingModifiedValues": {}
    }
}
```

사용할 파라미터에 대한 자세한 내용은 AWS CLI 주제 create-replication-group를 참조하십시오.

Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API)

다음 절차에서는 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 복제 그룹을 생성합니다.

Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 생성할 때 ElastiCache API CreateReplicationGroup 명령을 한 번만 호출하여 복제 그룹과 해당 노드를 모두 생성합니다. 다음 파 라미터를 포함합니다.

ReplicationGroupId

생성하는 복제 그룹의 이름입니다.

Redis(클러스터 모드 활성화됨) 복제 그룹 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

ReplicationGroupDescription

복제 그룹에 대한 설명입니다.

NumNodeGroups

이 복제 그룹과 함께 생성할 노드 그룹 수입니다. 유효한 값은 1~90입니다.

ReplicasPerNodeGroup

각 노드 그룹의 복제본 노드 수입니다. 유효한 값은 1~5입니다.

NodeGroupConfiguration

각 노드 그룹의 구성입니다. NodeGroupConfiguration 파라미터는 다음 필드로 구성됩니다.

• PrimaryAvailabilityZone – 이 노드 그룹의 기본 노드가 있는 가용 영역입니다. 이 파라미터가 생략되면 ElastiCache는 기본 노드의 가용 영역을 선택합니다.

예: us-west-2a.

- ReplicaAvailabilityZones 읽기 전용 복제본이 있는 가용 영역 목록입니다. 이 목록의 가용 영역 수는 ReplicaCount 값과 일치해야 합니다. 이 파라미터가 생략되면 ElastiCache는 복제본 노드의 가용 영역을 선택합니다.
- ReplicaCount 이 노드 그룹의 복제본 노드 수입니다.
- Slots 노드 그룹의 키스페이스를 지정하는 문자열입니다. 문자열 형식은 startKey-endKey입니다. 이 파라미터가 생략되면 ElastiCache는 키를 노드 그룹 간에 균등하게 할당합니다.

여: "0-4999"

CacheNodeType

복제 그룹에 있는 각 노드의 노드 유형입니다.

ElastiCache는 다음 노드 유형을 지원합니다. 일반적으로, 현재 세대 유형은 이전 세대의 동급 제품에 비해 더 많은 메모리와 컴퓨팅 파워를 더 저렴하게 제공합니다.

- 범용:
 - 현재 세대:

M5 노드 유형: cache.m5.large, cache.m5.xlarge, cache.m5.2xlarge, cache.m5.4xlarge, cache.m5.12xlarge, cache.m5.24xlarge

M4 노드 유형: cache.m4.large, cache.m4.xlarge, cache.m4.2xlarge, cache.m4.4xlarge, cache.m4.10xlarge

T2 노드 유형: cache.t2.micro, cache.t2.small, cache.t2.medium

• 이전 세대: (권장하지 않음)

T1 노드 유형: cache.t1.micro

M1 노드 유형: cache.m1.small, cache.m1.medium, cache.m1.large, cache.m1.xlarge

M3 노드 유형: cache.m3.medium, cache.m3.large, cache.m3.xlarge, cache.m3.2xlarge

- 컴퓨팅 최적화:
 - 이전 세대: (권장하지 않음)

C1 노드 유형: cache.c1.xlarge

- 메모리 최적화:
 - 현재 세대:

R5 노드 유형: cache.r5.large, cache.r5.xlarge, cache.r5.2xlarge, cache.r5.4xlarge, cache.r5.12xlarge, cache.r5.24xlarge

R4 노드 유형: cache.r4.large, cache.r4.xlarge, cache.r4.2xlarge, cache.r4.4xlarge, cache.r4.8xlarge, cache.r4.16xlarge

• 이전 세대: (권장하지 않음)

M2 노드 유형: cache.m2.xlarge, cache.m2.2xlarge, cache.m2.4xlarge

R3 노드 유형: cache.r3.large, cache.r3.xlarge, cache.r3.2xlarge, cache.r3.4xlarge, cache.r3.8xlarge

추가 노드 유형 정보

- 현재 세대의 모든 인스턴스 유형은 기본적으로 Amazon VPC에서 생성됩니다.
- Redis AOF(append-only files)는 T1 또는 T2 인스턴스에서 지원되지 않습니다.
- Redis 다중 AZ 및 자동 장애 조치는 T1 인스턴스에서 지원되지 않습니다.
- Redis 구성 변수 appendonly 및 appendfsync는 Redis 버전 2.8.22 이상에서 지원되지 않습니다.

CacheParameterGroup

default.redis3.2.cluster.on 파라미터 그룹 또는 default.redis3.2.cluster.on에서 파생된 파라미터 그룹을 지정하여 Redis(클러스터 모드 활성화됨) 복제 그룹을 생성합니다. 자세한 내용은 Redis 3.2.4 파라미터 변경 사항 (p. 282) 단원을 참조하십시오.

엔진

redis

EngineVersion

3.2.4

이 복제 그룹에서 전송 중 데이터 암호화 또는 미사용 데이터 암호화를 활성화하려면 TrasitEncryptionEnabled=true 또는 AtRestEncryptionEnabled=true 파라미터 중 하나 또는 둘 다를 추가하고 다음 조건을 충족해야 합니다.

- 복제 그룹에서 3.2.6 또는 4.0.10 버전 Redis를 실행하고 있어야 합니다.
- 복제 그룹은 Amazon VPC에 생성되어야 합니다.
- 또한 CacheSubnetGroup 파라미터도 포함해야 합니다.
- 또한 이 복제 그룹에서 작업을 수행하는 데 필요한 AUTH 토큰(암호)에 고객이 지정한 문자열 값이 있는 AuthToken 파라미터도 포함해야 합니다.

읽기 쉽도록 줄바꿈이 추가됩니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=CreateReplicationGroup
&CacheNodeType=cache.m4.large
&CacheParemeterGroup=default.redis3.2.cluster.on
&Engine=redis
&EngineVersion=3.2.4
&NumNodeGroups=3
&ReplicasPerNodeGroup=2
&ReplicationGroupDescription=test%20group
&ReplicationGroupId=myReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

사용할 파라미터에 대한 자세한 내용은 ElastiCache API 항목 CreateReplicationGroup를 참조하십시오.

복제 그룹의 세부 정보 보기

복제 그룹의 세부 정보를 보려는 경우가 있습니다. ElastiCache 콘솔, ElastiCache용 AWS CLI 또는 ElastiCache API를 사용할 수 있습니다. 콘솔 프로세스는 Redis(클러스터 모드 비활성화됨) 및 Redis(클러스 터 모드 활성화됨)에서 다릅니다.

복제 그룹의 세부 정보 보기

- 복제본으로 Redis(클러스터 모드 비활성화됨) 세부 정보 보기 (p. 161)
 - Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (콘솔) (p. 161)
 - Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (AWS CLI) (p. 161)
 - Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (ElastiCache API) (p. 161)
- 복제 그룹의 세부 정보 보기:Redis(클러스터 모드 활성화됨) (p. 161)
 - Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (콘솔) (p. 161)
 - Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (AWS CLI) (p. 162)
 - Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (ElastiCache API) (p. 162)
- 복제 그룹의 세부 정보 보기 (AWS CLI) (p. 162)
- 복제 그룹의 세부 정보 보기 (ElastiCache API) (p. 163)

복제본으로 Redis(클러스터 모드 비활성화됨) 세부 정보 보기

ElastiCache 콘솔, ElastiCache용 AWS CLI, ElastiCache API를 사용하여 복제본(API/CLI의 경우 복제 그룹) 이 있는 Redis(클러스터 모드 비활성화됨) 클러스터의 세부 정보를 볼 수 있습니다.

Redis(클러스터 모드 비활성화됨) 클러스터 세부 정보 보기

- Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (콘솔) (p. 161)
- Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (AWS CLI) (p. 161)
- Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (ElastiCache API) (p. 161)

Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (콘솔)

ElastiCache 콘솔을 사용하여 복제본이 있는 Redis(클러스터 모드 비활성화됨) 클러스터의 세부 정보를 보는 방법은 주제 Redis(클러스터 모드 비활성화됨) 클러스터 세부 정보 보기 (콘솔) (p. 78)를 참조하십시오.

Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (AWS CLI)

Redis(클러스터 모드 비활성화됨) 복제 그룹의 세부 정보를 표시하는 AWS CLI 예제에 대한 내용은 를 복제 그룹의 세부 정보 보기 (AWS CLI) (p. 162) 단원을 참조하십시오.

Redis(클러스터 모드 비활성화됨) 복제 그룹 세부 정보 보기 (ElastiCache API)

Redis(클러스터 모드 비활성화됨) 복제 그룹의 세부 정보를 표시하는 ElastiCache 예제에 대한 내용은 를 복제 그룹의 세부 정보 보기 (ElastiCache API) (p. 163) 단원을 참조하십시오.

복제 그룹의 세부 정보 보기:Redis(클러스터 모드 활성화됨)

Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (콘솔)

ElastiCache 콘솔을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터의 세부 정보를 확인하는 방법은 Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (콘솔) (p. 79) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (AWS CLI)

Redis(클러스터 모드 활성화됨) 복제 그룹의 세부 정보를 표시하는 ElastiCache CLI 예제에 대한 내용은 복제 그룹의 세부 정보 보기 (AWS CLI) (p. 162) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨) 클러스터 세부 정보 보기 (ElastiCache API)

Redis(클러스터 모드 활성화됨) 복제 그룹의 세부 정보를 표시하는 ElastiCache 예제에 대한 내용은 를 복제 그룹의 세부 정보 보기 (ElastiCache API) (p. 163) 단원을 참조하십시오.

복제 그룹의 세부 정보 보기 (AWS CLI)

AWS CLI describe-global-replication-groups 명령을 사용하여 복제의 세부 정보를 볼 수 있습니다. 목록을 구체화하려면 다음과 같은 선택적 파라미터를 사용합니다. 파라미터가 생략되면 최대 100개의 복제 그룹에 대한 세부 정보가 반환됩니다.

선택 사항 매개 변수

- --global-replication-group-id 이 파라미터를 사용하여 특정 복제 그룹의 세부 정보를 나열합니다. 지정한 복제 그룹의 노드 그룹이 둘 이상인 경우 결과가 노드 그룹별로 그룹화되어 반환됩니다.
- --max-items 이 파라미터를 사용하여 나열된 복제 그룹 수를 제한합니다. --max-items의 값은 20 이상 또는 100 이하여야 합니다.

Example

다음 코드는 최대 100개의 복제 그룹에 대한 세부 정보를 나열합니다.

```
aws elasticache describe-replication-groups
```

다음 코드는 sample-repl-group의 세부 정보를 나열합니다.

```
aws elasticache describe-replication-groups --replication-group-id sample-repl-group
```

다음 코드는 sample-repl-group의 세부 정보를 나열합니다.

```
aws elasticache describe-replication-groups --replication-group-id sample-repl-group
```

다음 코드는 최대 25개의 복제 그룹에 대한 세부 정보를 나열합니다.

```
aws elasticache describe-replication-groups --max-items 25
```

이 작업의 출력은 다음과 같습니다(JSON 형식).

```
"CacheNodeId": "0001",
                     "ReadEndpoint": {
                        "Port": 6379,
                        "Address": "rq-name-001.1abc4d.0001.usw2.cache.amazonaws.com"
                     "CacheClusterId": "rg-name-001"
                  },
                  {
                     "CurrentRole": "replica",
                     "PreferredAvailabilityZone": "us-west-2b",
                     "CacheNodeId": "0001",
                     "ReadEndpoint": {
                        "Port": 6379,
                        "Address": "rg-name-002.1abc4d.0001.usw2.cache.amazonaws.com"
                     "CacheClusterId": "rg-name-002"
                  },
                     "CurrentRole": "replica",
                     "PreferredAvailabilityZone": "us-west-2c",
                     "CacheNodeId": "0001",
                     "ReadEndpoint": {
                        "Port": 6379,
                        "Address": "rq-name-003.1abc4d.0001.usw2.cache.amazonaws.com"
                     "CacheClusterId": "rg-name-003"
                  }
               ],
               "NodeGroupId": "0001",
               "PrimaryEndpoint": {
                  "Port": 6379,
                  "Address": "rg-name.labc4d.ng.0001.usw2.cache.amazonaws.com"
            }
         ٦,
         "ReplicationGroupId": "rg-name",
         "AutomaticFailover": "enabled",
         "SnapshottingClusterId": "rg-name-002",
         "MemberClusters": [
            "rg-name-001",
            "rg-name-002",
            "rg-name-003"
         ٦,
         "PendingModifiedValues": {}
      },
      {
      ... some output omitted for brevity
   ]
}
```

자세한 내용은 ElastiCache 항목에 대한 AWS CLI describe-replication-groups를 참조하십시오.

복제 그룹의 세부 정보 보기 (ElastiCache API)

AWS CLI DescribeReplicationGroups 작업을 사용하여 복제의 세부 정보를 볼 수 있습니다. 목록을 구체화하려면 다음과 같은 선택적 파라미터를 사용합니다. 파라미터가 생략되면 최대 100개의 복제 그룹에 대한 세부 정보가 반환됩니다.

선택 사항 매개 변수

• ReplicationGroupId – 이 파라미터를 사용하여 지정한 복제 그룹의 세부 정보를 나열합니다. 지정한 복제 그룹의 노드 그룹이 둘 이상인 경우 결과가 노드 그룹별로 그룹화되어 반환됩니다. • MaxRecords – 이 파라미터를 사용하여 나열된 복제 그룹 수를 제한합니다. MaxRecords의 값은 20 이상 또는 100 이하여야 합니다. 기본값은 100입니다.

Example

다음 코드는 최대 100개의 복제 그룹에 대한 세부 정보를 나열합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReplicationGroups
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

다음 코드는 myReplGroup의 세부 정보를 나열합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReplicationGroups
&ReplicationGroupId=myReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

다음 코드는 클러스터 최대 25개의 세부 정보를 나열합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReplicationGroups
&MaxRecords=25
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 ElastiCache API 참조 항목 DescribeReplicationGroups를 참조하십시오.

복제 그룹 엔드포인트 확인

애플리케이션은 복제 그룹에 있는 어떤 노드에도 연결할 수 있습니다. 단, 애플리케이션에 해당 노드에 대한 DNS 엔드포인트 및 포트 번호가 있어야 합니다. 실행 중인 복제 그룹이 Redis(클러스터 모드 비활성화됨)인지, 아니면 Redis(클러스터 모드 활성화됨) 복제 그룹인지에 따라 관심을 갖게 되는 엔트포인트가 달라집니다.

Redis(클러스터 모드 비활성화됨)

복제본이 있는 Redis(클러스터 모드 비활성화됨) 클러스터에는 세 가지 유형의 엔드포인트(기본 엔드포인트, 리더 엔드포인트 및 노드 엔드포인트)가 있습니다. 기본 엔드포인트는 항상 클러스터의 기본 노드로 확인되는 DNS 이름입니다. 기본 엔드포인트는 읽기 전용 복제본을 기본 역할로 승격하는 것과 같은 클러스터 변경의 영향을 받지 않습니다. 쓰기 활동의 경우 애플리케이션을 직접 기본에 연결하는 대신 기본 엔드포인트에 연결하는 것이 좋습니다.

리더 엔드포인트는 Redis용 ElastiCache 클러스터의 모든 읽기 전용 복제본에 대한 수신 연결을 고르게 분할합니다. 애플리케이션이 연결을 생성하는 시기 또는 애플리케이션에서 연결을 다시 사용하는 방법과 같은 추가 요소가 트래픽 분산을 결정합니다. 리더 엔드포인트는 복제본이 추가 또는 제거되는 클러스터의 변경 사

항을 실시간으로 반영합니다. Redis용 ElastiCache 클러스터의 여러 읽기 전용 복제본을 다양한 AWS 가용 영역(AZ)에 두어 리더 엔드포인트의 가용성을 높일 수 있습니다.

읽기 활동의 경우 애플리케이션은 클러스터의 어떤 노드에도 연결할 수 있습니다. 기본 엔드포인트와 달리, 노드 엔드포인트는 특정 엔드포인트로 확인됩니다. 복제본을 추가하거나 삭제하는 것과 같이 클러스터를 변 경하면 애플리케이션에서 노드 엔드포인트를 업데이트해야 합니다.

Redis(클러스터 모드 활성화됨)

복제본이 있는 Redis(클러스터 모드 활성화됨) 클러스터의 엔드포인트 구조는 Redis(클러스터 모드 비활성화됨) 클러스터의 엔드포인트 구조와 다릅니다. 왜냐하면 이러한 클러스터에는 여러 샤드(API/CLI의 경우 노드 그룹)가 있으며 이는 기본 노드도 여러 개임을 의미하기 때문입니다. Redis(클러스터 모드 활성화됨)에는 클러스터의 모든 기본 엔드포인트 및 노드 엔드포인트를 "아는" 구성 엔드포인트가 있습니다. 애플리케이션은 이 구성 엔드포인트에 연결됩니다. 애플리케이션이 클러스터의 구성 엔드포인트에서 쓰거나 읽을 때마다백그라운드에서 Redis는 키가 속하는 샤드와 해당 샤드에서 사용할 엔드포인트를 확인합니다. 이 모든 것이애플리케이션에 매우 투명하게 진행됩니다.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 클러스터의 엔드포인트를 찾을 수 있습니다.

복제 그룹 엔드포인트 찾기

복제 그룹의 엔드포인트를 찾으려면 다음 항목 중 하나를 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 181)
- Redis(클러스터 모드 활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 182)
- 복제 그룹의 엔드포인트 찾기 (AWS CLI) (p. 185)
- 복제 그룹의 엔드포인트 찾기 (ElastiCache API) (p. 187)

복제 그룹 수정

중요한 제약

• 현재, ElastiCache는 Redis(클러스터 모드 활성화됨) 복제 그룹 수정을 제한적으로 지원하고 있습니다. 예를 들면 API 작업 ModifyReplicationGroup (CLI의 경우 modifyreplication-group)을 사용하여 엔진 버전을 변경할 수 있습니다. API 작업을 통해 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드(노드 그룹) 수를 수정할 수 있습니다 ModifyReplicationGroupShardConfiguration (CLI: modify-replication-group-shard-configuration). 자세한 내용은 Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨) 클러스터의 다른 부분을 수정하려면 변경 사항을 통합하는 새 클러스터를 사용해 클러스터를 새로 만들어야 합니다.

• Redis(클러스터 모드 비활성화됨) 및 Redis(클러스터 모드 활성화됨) 클러스터 및 복제 그룹을 최신 엔진 버전으로 업그레이드할 수 있습니다. 하지만 기존의 클러스터 또는 복제 그룹을 삭제하고 새로 만들지 않는 한 이전 엔진 버전으로 다운그레이드할 수 없습니다. 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 비활성화됨) 클러스터 의 설정을 수정할 수 있습니다. 현재 ElastiCache에서는 Redis(클러스터 모드 활성화됨) 복제 그룹에 대한 제한된 수의 수정을 지원합니다. 다른 수정의 경우에는 현재 복제 그룹의 백업을 생성한 후 해당 백업을 사용해 새 Redis(클러스터 모드 활성화됨) 복제 그룹을 시드해야 합니다.

주제

- AWS Management 콘솔 사용 (p. 166)
- AWS CLI 사용 (p. 166)

• ElastiCache API 사용 (p. 166)

AWS Management 콘솔 사용

Redis(클러스터 모드 비활성화됨) 클러스터를 수정하는 방법은 ElastiCache 클러스터 수정 (p. 87) 단원을 참조하십시오.

AWS CLI 사용

다음 AWS CLI 명령은 기존 Redis 복제 그룹에서 다중 AZ를 활성화합니다. 동일한 명령을 사용하여 복제 그룹을 수정할 수 있습니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group \
    --replication-group-id myReplGroup \
    --automatic-failover-enabled
```

Windows의 경우:

```
aws elasticache modify-replication-group ^
    --replication-group-id myReplGroup ^
    --automatic-failover-enabled
```

AWS CLI modify-replication-group 명령에 대한 자세한 내용은 modify-replication-group 단원을 참조하십시오.

ElastiCache API 사용

다음 ElastiCache API 작업은 기존 Redis 복제 그룹에서 다중 AZ를 활성화합니다. 동일한 작업을 사용하여 복제 그룹을 수정할 수 있습니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=ModifyReplicationGroup
    &AutomaticFailoverEnabled=true
    &ReplicationGroupId=myReplGroup
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20141201T220302Z
    &Version=2014-12-01
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20141201T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Credential=<credential>
    &X-Amz-Signature=<signature>
```

ElastiCache API ModifyReplicationGroup 작업에 대한 자세한 내용은 ModifyReplicationGroup 단원을 참조하십시오.

복제 그룹 삭제

복제본이 있는 클러스터(API/CLI에서는 복제 그룹이라고 함)이 더 이상 필요하지 않으면 삭제할 수 있습니다. 복제 그룹을 삭제할 때 ElastiCache는 해당 그룹의 노드를 모두 삭제합니다.

작업을 시작하면 중단하거나 취소할 수 없습니다.

Warning

Redis용 ElastiCache 클러스터를 삭제하는 경우 수동 스냅샷은 보존됩니다. 클러스터를 삭제하기 전에 최종 스냅샷을 생성할 수 있는 옵션도 있습니다. 자동 캐시 스냅샷은 보존되지 않습니다.

복제 그룹 (콘솔) 삭제

복제본이 있는 클러스터를 삭제하려면 클러스터 삭제하기 (p. 102)를 참조하십시오.

복제 그룹 삭제 (AWS CLI)

delete-replication-group 명령을 사용해 복제 그룹을 삭제합니다.

aws elasticache delete-replication-group --replication-group-id my-repgroup

결정을 확인하라는 메시지가 나타납니다. 즉시 작업을 시작하려면 [y](예)를 입력합니다. 프로세스가 시작되면 되돌릴 수 없습니다.

After you begin deleting this replication group, all of its nodes will be deleted as well.

Are you sure you want to delete this replication group? [Ny]y

REPLICATIONGROUP my-repgroup My replication group deleting

복제 그룹 삭제 (ElastiCache API)

ReplicationGroup 파라미터를 사용하여 DeleteReplicationGroup을 호출하십시오.

Example

https://elasticache.us-west-2.amazonaws.com/
?Action=DeleteReplicationGroup
&ReplicationGroupId=my-repgroup
&Version=2014-12-01
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Signature=<signature>

Note

RetainPrimaryCluster 파라미터를 true로 설정하면 모든 읽기 전용 복제본이 삭제되지만 기본 클러스터는 보존됩니다.

복제본 수 변경

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용해 Redis 복제 그룹의 읽기 전용 복제본 수를 동적으로 늘리거나 줄일 수 있습니다. 복제 그룹이 Redis(클러스터 모드 활성화됨) 복제 그룹인 경우 복제본 수를 늘리거나 줄일 샤드(노드 그룹)를 선택할 수 있습니다.

Redis 복제 그룹의 복제본 수를 동적으로 변경하려면 다음 표에서 상황에 맞는 작업을 선택하십시오.

방법	Redis(클러스터 모드 활성화됨)의 경우	Redis(클러스터 모드 비활성화됨) 의 경우
복제본 추가	샤드의 복제본 수 늘리 기 (p. 168)	샤드의 복제본 수 늘리 기 (p. 168)
		Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제 본 추가 (p. 176)
복제본 삭제	샤드의 복제본 수 줄이 기 (p. 172)	샤드의 복제본 수 줄이 기 (p. 172)
		Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제 본 삭제 (p. 178)

샤드의 복제본 수 늘리기

Redis(클러스터 모드 활성화됨) 샤드 또는 Redis(클러스터 모드 비활성화됨) 복제 그룹의 복제본 수를 최대 5 개까지 늘릴 수 있습니다. AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용해 늘릴 수 있습니다.

주제

- AWS Management 콘솔 사용 (p. 168)
- AWS CLI 사용 (p. 169)
- ElastiCache API 사용 (p. 170)

AWS Management 콘솔 사용

다음 절차는 콘솔을 사용해 Redis(클러스터 모드 활성화됨) 복제 그룹의 복제본 수를 늘립니다.

Redis 샤드의 복제본 수 늘리는 방법

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 Redis를 선택한 후 복제본을 추가할 복제 그룹의 이름을 선택합니다.
- 3. 복제본을 추가할 각 샤드의 상자를 선택합니다.
- 4. Add replicas(복제본 추가)를 선택합니다.
- 5. Add Replicas to Shards(샤드에 복제본 추가) 페이지를 완료합니다.
 - New number of replicas/shard(새 복제본/샤드 수)에 선택한 모든 샤드에 있도록 하려는 복제본 수를 입력합니다. 이 값은 Current Number of Replicas per shard(샤드당 현재 복제본 수)보다 크거나 같아 야 하며 5보다 작거나 같아야 합니다. 최소한 두 개의 복제본을 사용하는 것이 좋습니다.
 - 가용 영역에서 ElastiCache가 각각의 새 복제본에 대해 가용 영역을 선택하도록 하려면 기본 설정 없음을 선택하고 각각의 새 복제본에 대해 가용 영역을 선택하려면 Specify Availability Zones(가용 영역지정)를 선택합니다.

Specify Availability Zones(가용 영역 지정)를 선택할 경우 목록을 사용해 각각의 새 복제본에 대해 가용 영역을 지정하십시오.

6. 추가를 선택하여 복제본을 추가하거나 취소를 선택하여 작업을 취소합니다.

AWS CLI 사용

Redis 샤드의 복제본 수를 늘리려면 다음 파라미터와 함께 increase-replica-count 명령을 사용합니다.

- --replication-group-id 필수 사항. 복제본 수를 늘리려는 복제 그룹을 식별합니다.
- --apply-immediately 또는 --no-apply-immediately 필수 사항. 복제본 수를 즉시 늘릴 것인지 (--apply-immediately) 아니면 다음 번 유지 관리 기간에 늘릴 것인지(--no-apply-immediately) 지정합니다. 현재 --no-apply-immediately는 지원되지 않습니다.
- --new-replica-count 선택 사항. 완료된 경우 원하는 복제본 노드의 수를 최대 5개까지 지정합니다. 노드 그룹이 하나만 있는 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 또는 모든 노드 그룹에 동일 한 수의 복제본이 있도록 하려는 Redis(클러스터 모드 활성화됨) 그룹에 대해 이 파라미터를 사용합니다. 이 값이 노드 그룹의 현재 복제본 수보다 크지 않은 경우 호출이 실패하고 예외가 발생합니다.
- --replica-configuration 선택 사항. 각 노드 그룹에 대해 독립적으로 복제본 수와 가용 영역을 설정할 수 있도록 합니다. 각 노드 그룹을 독립적으로 구성하려는 경우 Redis(클러스터 모드 활성화됨) 그룹에 대해 이 파라미터를 사용하십시오.
 - --replica-configuration에는 다음의 선택 멤버 3개가 있습니다.
 - NodeGroupId 구성하는 노드 그룹의 4자리 ID입니다. Redis(클러스터 모드 비활성화됨) 복제 그룹의 경우 샤드 ID는 항상 0001입니다. Redis(클러스터 모드 활성화됨) 노드 그룹(샤드)의 ID를 찾으려면 샤드 ID를 검색하는 방법 (p. 112) 단원을 참조하십시오.
 - NewReplicaCount 이 작업이 끝날 때 이 노드 그룹에 둘 복제본의 수입니다. 값은 현재 복제본 수보다 커야 하며, 최대 5개까지입니다. 이 값이 노드 그룹의 현재 복제본 수보다 크지 않은 경우 호출이 실패하고 예외가 발생합니다.
 - PreferredAvailabilityZones 복제 그룹의 노드가 있을 가용 영역을 지정하는 PreferredAvailabilityZone 문자열의 목록입니다. PreferredAvailabilityZone 값의 수는 기본 노드를 고려하여 NewReplicaCount에 1을 더한 값과 같아야 합니다. 이 --replicaconfiguration 멤버가 생략되면 Redis용 ElastiCache는 각각의 새 복제본에 대해 가용 영역을 선택합니다.

Important

호출에 --new-replica-count 또는 --replica-configuration 파라미터를 포함해야 하지만, 둘 다 포함해서는 안 됩니다.

Example

다음은 복제 그룹 sample-repl-group의 복제본 수를 3으로 늘리는 예입니다. 예제가 완료되면 각 노드 그룹에 복제본 3개가 있습니다. 단일 노드 그룹의 Redis(클러스터 모드 비활성화됨) 그룹이든 여러 노드 그룹의 Redis(클러스터 모드 활성화됨) 그룹이든 관계없이 이 숫자가 적용됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache increase-replica-count \
    --replication-group-id sample-repl-group \
    --new-replica-count 3 \
    --apply-immediately
```

Windows의 경우:

```
aws elasticache increase-replica-count ^
    --replication-group-id sample-repl-group ^
    --new-replica-count 3 ^
    --apply-immediately
```

다음은 복제 그룹 sample-repl-group의 복제본 수를 지정된 2개의 노드 그룹에 대해 지정된 값으로 늘리는 예입니다. 여러 노드 그룹이 있는 경우 이는 Redis(클러스터 모드 활성화됨) 복제 그룹입니다. 선택적 PreferredAvailabilityZones를 지정할 때 나열된 가용 영역 수는 NewReplicaCount에 1 이상을 더한 값과 같아야 합니다. 이러한 접근 방식은 NodeGroupId에서 식별한 그룹에 대한 기본 노드를 설명합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache increase-replica-count \
    --replication-group-id sample-repl-group \
    --replica-configuration \
    NodeGroupId=0001, NewReplicaCount=2, PreferredAvailabilityZones=us-east-1a, us-east-1c, us-east-1b \
    NodeGroupId=0003, NewReplicaCount=3, PreferredAvailabilityZones=us-east-1a, us-east-1b, us-east-1c, us-east-1c \
    --apply-immediately
```

Windows의 경우:

CLI를 사용하여 복제본 수를 늘리는 방법에 대한 자세한 내용은 Amazon ElastiCache 명령줄 레퍼런스의 increase-replica-count를 참조하십시오.

ElastiCache API 사용

Redis 샤드의 복제본 수를 늘리려면 다음 파라미터와 함께 IncreaseReplicaCount 작업을 사용합니다.

- ReplicationGroupId 필수 사항. 복제본 수를 늘리려는 복제 그룹을 식별합니다.
- ApplyImmediately 필수 사항. 복제본 수를 즉시 늘릴 것인지(ApplyImmediately=True) 아니면 다음 번 유지 관리 기간에 늘릴 것인지(ApplyImmediately=False) 지정합니다. 현재 ApplyImmediately=False는 지원되지 않습니다.
- NewReplicaCount 선택 사항. 완료된 경우 원하는 복제본 노드의 수를 최대 5개까지 지정합니다. 노드 그룹이 하나만 있는 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 또는 모든 노드 그룹에 동일한 수 의 복제본이 있도록 하려는 Redis(클러스터 모드 활성화됨) 그룹에 대해 이 파라미터를 사용합니다. 이 값 이 노드 그룹의 현재 복제본 수보다 크지 않은 경우 호출이 실패하고 예외가 발생합니다.
- ReplicaConfiguration 선택 사항. 각 노드 그룹에 대해 독립적으로 복제본 수와 가용 영역을 설정할수 있도록 합니다. 각 노드 그룹을 독립적으로 구성하려는 경우 Redis(클러스터 모드 활성화됨) 그룹에 대해 이 파라미터를 사용하십시오.

ReplicaConfiguraion에는 다음의 선택 멤버 3개가 있습니다.

- NodeGroupId 구성하는 노드 그룹의 4자리 ID입니다. Redis(클러스터 모드 비활성화됨) 복제 그룹의 경우 노드 그룹(샤드) ID는 항상 0001입니다. Redis(클러스터 모드 활성화됨) 노드 그룹(샤드)의 ID를 찾으려면 샤드 ID를 검색하는 방법 (p. 112) 단원을 참조하십시오.
- NewReplicaCount 이 작업이 끝날 때 이 노드 그룹에 둘 복제본의 수입니다. 값은 현재 복제본 수보다 커야 하며 최대 5개까지입니다. 이 값이 노드 그룹의 현재 복제본 수보다 크지 않은 경우 호출이 실패하고 예외가 발생합니다.
- PreferredAvailabilityZones 복제 그룹의 노드가 있을 가용 영역을 지정하는 PreferredAvailabilityZone 문자열의 목록입니다. PreferredAvailabilityZone

값의 수는 기본 노드를 고려하여 NewReplicaCount에 1을 더한 값과 같아야 합니다. 이 ReplicaConfiguration 멤버가 생략되면 Redis용 ElastiCache는 각각의 새 복제본에 대해 가용 영역을 선택합니다.

Important

호출에 NewReplicaCount 또는 ReplicaConfiguration 파라미터를 포함해야 하지만, 둘 다 포함해서는 안 됩니다.

Example

east-1c

다음은 복제 그룹 sample-repl-group의 복제본 수를 3으로 늘리는 예입니다. 예제가 완료되면 각 노드 그룹에 복제본 3개가 있습니다. 단일 노드 그룹의 Redis(클러스터 모드 비활성화됨) 그룹이든 여러 노드 그룹의 Redis(클러스터 모드 활성화됨) 그룹이든 관계없이 이 숫자가 적용됩니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=IncreaseReplicaCount
&ApplyImmediately=True
&NewReplicaCount=3
&ReplicationGroupId=sample-repl-group
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

다음은 복제 그룹 sample-repl-group의 복제본 수를 지정된 2개의 노드 그룹에 대해 지정된 값으로 늘리는 예입니다. 여러 노드 그룹이 있는 경우 이는 Redis(클러스터 모드 활성화됨) 복제 그룹입니다. 선택적 PreferredAvailabilityZones를 지정할 때 나열된 가용 영역 수는 NewReplicaCount에 1 이상을 더한 값과 같아야 합니다. 이러한 접근 방식은 NodeGroupId에서 식별한 그룹에 대한 기본 노드를 설명합니다.

https://elasticache.us-west-2.amazonaws.com/ ?Action=IncreaseReplicaCount &ApplyImmediately=True &ReplicaConfiguration.ConfigureShard.1.NodeGroupId=0001 &ReplicaConfiguration.ConfigureShard.1.NewReplicaCount=2 &ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.1=useast-1a &ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.2=useast-1c &ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.3=useast-1b &ReplicaConfiguration.ConfigureShard.2.NodeGroupId=0003 &ReplicaConfiguration.ConfigureShard.2.NewReplicaCount=3 &ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.1=useast-1a &ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.2=useast-1b &ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.3=us-&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.4=us-

&ReplicationGroupId=sample-repl-group

&Version=2015-02-02 &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20150202T192317Z &X-Amz-Credential=<credential>

API를 사용하여 복제본 수를 늘리는 방법에 대한 자세한 내용은 Amazon ElastiCache API Reference의 IncreaseReplicaCount를 참조하십시오.

샤드의 복제본 수 줄이기

Redis(클러스터 모드 활성화됨)에 대한 샤드 또는 Redis(클러스터 모드 비활성화됨)에 대한 복제 그룹의 복 제본 수를 줄일 수 있습니다.

- Redis(클러스터 모드 비활성화됨)의 경우 자동 장애 조치가 포함된 다중 AZ가 활성화된 경우 1로, 활성화되지 않은 경우 0으로 복제본 수를 줄일 수 있습니다.
- Redis(클러스터 모드 활성화됨)의 경우 복제본 수를 0으로 줄일 수 있습니다. 그러나 기본 노드가 실패할 경우 복제본으로 장애 조치를 수행할 수 없습니다.

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용해 노드 그룹(샤드) 또는 복제 그룹의 복제 본 수를 줄일 수 있습니다.

주제

- AWS Management 콘솔 사용 (p. 172)
- AWS CLI 사용 (p. 173)
- ElastiCache API 사용 (p. 174)

AWS Management 콘솔 사용

다음 절차는 콘솔을 사용해 Redis(클러스터 모드 활성화됨) 복제 그룹의 복제본 수를 줄입니다.

Redis 샤드의 복제본 수 줄이는 방법

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 Redis를 선택한 후 복제본을 삭제할 복제 그룹의 이름을 선택합니다.
- 3. 복제본 노드를 제거할 각 샤드의 상자를 선택합니다.
- 4. Delete replicas(복제본 삭제)를 선택합니다.
- 5. Delete Replicas from Shards(샤드에서 복제본 삭제) 페이지를 완료합니다.
 - a. New number of replicas/shard(새 복제본/샤드 수)에 선택한 샤드에 있도록 하려는 복제본 수를 입력합니다. 이 숫자는 1보다 크거나 같아야 합니다. 샤드마다 최소한 두 개의 복제본을 사용하는 것이 좋습니다.
 - b. 삭제를 선택하여 복제본을 삭제하거나 취소를 선택하여 작업을 취소합니다.

Important

- 삭제할 복제본 노드를 지정하지 않으면 Redis용 ElastiCache에서 삭제할 복제본 노드를 자동으로 선택합니다. 이렇게 하는 동안 Redis용 ElastiCache는 복제 그룹의 다중 AZ 아키텍처를 유지하고. 마스터를 사용하여 최소 복제 지연 시간으로 복제본을 유지하려고 시도합니다.
- 복제 그룹의 기본 노드 또는 마스터 노드는 삭제할 수 없습니다. 기본 노드를 삭제하도록 지정하면 작업이 실패하고, 기본 노드가 삭제되도록 선택되었음을 나타내는 오류 이벤트가 발생합니다.

AWS CLI 사용

Redis 샤드의 복제본 수를 줄이려면 다음 파라미터와 함께 decrease-replica-count 명령을 사용합니다.

- --replication-group-id 필수 사항. 복제본 수를 줄이려는 복제 그룹을 식별합니다.
- --apply-immediately 또는 --no-apply-immediately 필수 사항. 복제본 수를 즉시 줄일 것인지 (--apply-immediately) 아니면 다음 번 유지 관리 기간에 줄일 것인지(--no-apply-immediately) 지정합니다. 현재 --no-apply-immediately는 지원되지 않습니다.
- --new-replica-count 선택 사항. 원하는 복제본 노드의 수를 지정합니다. --new-replica-count의 값은 유효해야 하며, 노드 그룹의 현재 복제본 수보다 작아야 합니다. 허용된 최소값은 샤드의 복제본 수 줄이기 (p. 172) 단원을 참조하십시오. --new-replica-count의 값이 이 요구 사항을 충족하지 않는 경우 호출이 실패합니다.
- --replicas-to-remove 선택 사항. 제거할 복제본 노드를 지정하는 노드 ID 목록을 포함합니다.
- --replica-configuration 선택 사항. 각 노드 그룹에 대해 독립적으로 복제본 수와 가용 영역을 설정할 수 있도록 합니다. 각 노드 그룹을 독립적으로 구성하려는 경우 Redis(클러스터 모드 활성화됨) 그룹에 대해 이 파라미터를 사용하십시오.
 - --replica-configuration에는 다음의 선택 멤버 3개가 있습니다.
 - NodeGroupId 구성하는 노드 그룹의 4자리 ID입니다. Redis(클러스터 모드 비활성화됨) 복제 그룹의 경우 샤드 ID는 항상 0001입니다. Redis(클러스터 모드 활성화됨) 노드 그룹(샤드)의 ID를 찾으려면 샤드 ID를 검색하는 방법 (p. 112) 단원을 참조하십시오.
 - NewReplicaCount 선택적 파라미터로, 원하는 복제본 노드의 수를 지정합니다. NewReplicaCount의 값은 유효해야 하며, 노드 그룹의 현재 복제본 수보다 작아야 합니다. 허용된 최 소값은 샤드의 복제본 수 줄이기 (p. 172) 단원을 참조하십시오. NewReplicaCount의 값이 이 요구 사항을 충족하지 않는 경우 호출이 실패합니다.
 - PreferredAvailabilityZones 복제 그룹의 노드가 있는 가용 영역을 지정하는 PreferredAvailabilityZone 문자열의 목록입니다. PreferredAvailabilityZone 값의 수는 기본 노드를 고려하여 NewReplicaCount에 1을 더한 값과 같아야 합니다. 이 --replica-configuration 멤버가 생략되면 Redis용 ElastiCache는 각각의 새 복제본에 대해 가용 영역을 선택합니다.

Important

--new-replica-count, --replicas-to-remove 또는 --replica-configuration 파라미터 중 하나만 포함해야 합니다.

Example

다음은 --new-replica-count를 사용해 복제 그룹 sample-repl-group의 복제본 수를 1로 줄이는 예입니다. 예제가 완료되면 각 노드 그룹에 복제본 1개가 있습니다. 단일 노드 그룹의 Redis(클러스터 모드 비활성화됨) 그룹이든 여러 노드 그룹의 Redis(클러스터 모드 활성화됨) 그룹이든 관계없이 이 숫자가 적용됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache decrease-replica-count
    --replication-group-id sample-repl-group \
    --new-replica-count 1 \
    --apply-immediately
```

Windows의 경우:

aws elasticache decrease-replica-count ^

```
--replication-group-id sample-repl-group ^
--new-replica-count 1 ^
--apply-immediately
```

다음은 노드 그룹에서 지정된 복제본 2개(0001 및 0003)를 제거하여 복제 그룹 sample-repl-group의 복제본 수를 줄이는 예입니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache decrease-replica-count \
    --replication-group-id sample-repl-group \
    --replicas-to-remove 0001,0003 \
    --apply-immediately
```

Windows의 경우:

```
aws elasticache decrease-replica-count ^
    --replication-group-id sample-repl-group ^
    --replicas-to-remove 0001,0003 \
    --apply-immediately
```

다음은 --replica-configuration을 사용해 복제 그룹 sample-repl-group의 복제본 수를 지정된 2 개의 노드 그룹에 대해 지정된 값으로 줄이는 예입니다. 여러 노드 그룹이 있는 경우 이는 Redis(클러스터 모드 활성화됨) 복제 그룹입니다. 선택적 PreferredAvailabilityZones를 지정할 때 나열된 가용 영역 수는 NewReplicaCount에 1 이상을 더한 값과 같아야 합니다. 이러한 접근 방식은 NodeGroupId에서 식별한 그룹에 대한 기본 노드를 설명합니다.

Linux, macOS, or Unix의 경우:

Windows의 경우:

CLI를 사용하여 복제본 수를 줄이는 방법에 대한 자세한 내용은 Amazon ElastiCache 명령줄 레퍼런스의 decrease-replica-count를 참조하십시오.

ElastiCache API 사용

Redis 샤드의 복제본 수를 줄이려면 다음 파라미터와 함께 DecreaseReplicaCount 작업을 사용합니다.

• ReplicationGroupId – 필수 사항. 복제본 수를 줄이려는 복제 그룹을 식별합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 복제본 수 변경

- ApplyImmediately 필수 사항. 복제본 수를 즉시 줄일 것인지(ApplyImmediately=True) 아니면 다음 번 유지 관리 기간에 줄일 것인지(ApplyImmediately=False) 지정합니다. 현재 ApplyImmediately=False는 지원되지 않습니다.
- NewReplicaCount 선택 사항. 원하는 복제본 노드의 수를 지정합니다. NewReplicaCount의 값은 유효해야 하며, 노드 그룹의 현재 복제본 수보다 작아야 합니다. 허용된 최소값은 샤드의 복제본 수 줄이기 (p. 172) 단원을 참조하십시오. --new-replica-count의 값이 이 요구 사항을 충족하지 않는 경우호출이 실패합니다.
- ReplicasToRemove 선택 사항. 제거할 복제본 노드를 지정하는 노드 ID 목록을 포함합니다.
- ReplicaConfiguration 선택 사항. 각 노드 그룹에 대해 독립적으로 복제본 수와 가용 영역을 설정할 수 있도록 허용하는 노드 그룹의 목록을 포함합니다. 각 노드 그룹을 독립적으로 구성하려는 경우 Redis(클러스터 모드 활성화됨) 그룹에 대해 이 파라미터를 사용하십시오.

ReplicaConfiguration에는 다음의 선택 멤버 3개가 있습니다.

- NodeGroupId 구성하는 노드 그룹의 4자리 ID입니다. Redis(클러스터 모드 비활성화됨) 복제 그룹의 경우 노드 그룹 ID는 항상 0001입니다. Redis(클러스터 모드 활성화됨) 노드 그룹(샤드)의 ID를 찾으려면 샤드 ID를 검색하는 방법 (p. 112) 단원을 참조하십시오.
- NewReplicaCount 이 작업이 끝날 때 이 노드 그룹에 둘 복제본의 수입니다. 값은 현재 복제본 수보다 작아야 하며, 자동 장애 조치가 포함된 다중 AZ가 활성화된 경우 1 또는 자동 장애 조치가 있는 다중 AZ가 활성화되지 않은 경우 0의 최소값으로 줄입니다. 이 값이 노드 그룹의 현재 복제본 수보다 작지 않은 경우 호출이 실패하고 예외가 발생합니다.
- PreferredAvailabilityZones 복제 그룹의 노드가 있는 가용 영역을 지정하는 PreferredAvailabilityZone 문자열의 목록입니다. PreferredAvailabilityZone 값의 수는 기본 노드를 고려하여 NewReplicaCount에 1을 더한 값과 같아야 합니다. 이 ReplicaConfiguration 멤버가 생략되면 Redis용 ElastiCache는 각각의 새 복제본에 대해 가용 영역을 선택합니다.

Important

NewReplicaCount, ReplicaSToRemove 또는 ReplicaConfiguration 파라미터 중 하나만 포함해야 합니다.

Example

다음은 NewReplicaCount를 사용해 복제 그룹 sample-repl-group의 복제본 수를 1로 줄이는 예입니다. 예제가 완료되면 각 노드 그룹에 복제본 1개가 있습니다. 단일 노드 그룹의 Redis(클러스터 모드 비활성화됨) 그룹이든 여러 노드 그룹의 Redis(클러스터 모드 활성화됨) 그룹이든 관계없이 이 숫자가 적용됩니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DecreaseReplicaCount
&ApplyImmediately=True
&NewReplicaCount=1
&ReplicationGroupId=sample-repl-group
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

다음은 노드 그룹에서 지정된 복제본 2개(0001 및 0003)를 제거하여 복제 그룹 sample-repl-group의 복제본 수를 줄이는 예입니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DecreaseReplicaCount
&ApplyImmediately=True
&ReplicasToRemove.ReplicaToRemove.1=0001

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 복제본 수 변경

&ReplicasToRemove.ReplicaToRemove.2=0003 &ReplicationGroupId=sample-repl-group &Version=2015-02-02 &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20150202T192317Z &X-Amz-Credential=<credential>

다음은 ReplicaConfiguration을 사용해 복제 그룹 sample-repl-group의 복제본 수를 지정된 2개의 노드 그룹에 대해 지정된 값으로 줄이는 예입니다. 여러 노드 그룹이 있는 경우 이는 Redis(클러스터 모드활성화됨) 복제 그룹입니다. 선택적 PreferredAvailabilityZones를 지정할 때 나열된 가용 영역 수는 NewReplicaCount에 1 이상을 더한 값과 같아야 합니다. 이러한 접근 방식은 NodeGroupId에서 식별한그룹에 대한 기본 노드를 설명합니다.

```
https://elasticache.us-west-2.amazonaws.com/
      ?Action=DecreaseReplicaCount
      &ApplyImmediately=True
      &ReplicaConfiguration.ConfigureShard.1.NodeGroupId=0001
      &ReplicaConfiguration.ConfigureShard.1.NewReplicaCount=1
&ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.1=us-
east-1a
&ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.2=us-
      &ReplicaConfiguration.ConfigureShard.2.NodeGroupId=0003
      &ReplicaConfiguration.ConfigureShard.2.NewReplicaCount=2
&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.1=us-
&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.2=us-
&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.4=us-
      &ReplicationGroupId=sample-repl-group
      &Version=2015-02-02
      &SignatureVersion=4
      &SignatureMethod=HmacSHA256
      &Timestamp=20150202T192317Z
      &X-Amz-Credential = < credential >
```

API를 사용하여 복제본 수를 줄이는 방법에 대한 자세한 내용은 Amazon ElastiCache API Reference의 DecreaseReplicaCount를 참조하십시오.

Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복 제본 추가

다음 주제의 정보는 Redis(클러스터 모드 비활성화됨) 복제 그룹에만 적용됩니다.

읽기 트래픽이 증가함에 따라 이러한 읽기를 더 많은 노드로 분산시켜 어느 한 노드에 대한 읽기 압력을 줄이려고 할 수 있습니다. 이 주제에서는 Redis(클러스터 모드 비활성화됨) 클러스터에 읽기 전용 복제본을 추가하는 방법을 확인할 수 있습니다.

Redis(클러스터 모드 비활성화됨) 복제 그룹은 최대 5개의 읽기 전용 복제본을 가질 수 있습니다. 읽기 전용 복제본 5개가 이미 있는 복제 그룹에 읽기 전용 복제본을 추가하려고 하면 작업이 실패합니다.

Redis(클러스터 모드 활성화됨) 복제 그룹에 복제본을 추가하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 복제본 수 변경

- Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241)
- 샤드의 복제본 수 늘리기 (p. 168)

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 읽기 전용 복제본을 Redis(클러스터 모드 비활성화됨) 클러스터에 추가할 수 있습니다.

관련 주제

- 클러스터에 노드 추가 (p. 91)
- 복제 그룹에 읽기 전용 복제본 추가 (AWS CLI) (p. 177)
- API를 사용해 복제 그룹에 읽기 전용 복제본 추가 (p. 177)

복제 그룹에 읽기 전용 복제본 추가 (AWS CLI)

Redis(클러스터 모드 비활성화됨) 복제 그룹에 읽기 전용 복제본을 추가하려면 --replication-group-id 파라미터와 함께 AWS CLI create-cache-cluster 명령을 사용해 클러스터(노드)를 추가할 복제 그룹을 지정합니다.

다음 예제에서는 my-read replica클러스터를 생성하고 해당 클러스터를 my-replication-group 복제 그룹에 추가합니다. 읽기 전용 복제본의 노드 유형, 파라미터 그룹, 보안 그룹, 유지 관리 기간 및 기타 설정이 my-replication-group의 다른 노드와 동일해집니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-cache-cluster \
    --cache-cluster-id my-read replica \
    --replication-group-id my-replication-group
```

Windows의 경우:

```
aws elasticache create-cache-cluster ^
--cache-cluster-id my-read replica ^
--replication-group-id my-replication-group
```

CLI를 사용하여 읽기 전용 복제본을 추가하는 방법에 대한 자세한 내용은 Amazon ElastiCache 명령줄 레퍼 런스의 create-cache-cluster를 참조하십시오.

API를 사용해 복제 그룹에 읽기 전용 복제본 추가

Redis(클러스터 모드 비활성화됨) 복제 그룹에 읽기 전용 복제본을 추가하려면 ReplicationGroupId 파라미터와 함께 ElastiCache CreateCacheCluster 작업을 사용해 클러스터(노드)를 추가할 복제 그룹을 지정합니다.

다음 예제에서는 myReadReplica클러스터를 생성하고 해당 클러스터를 myReplicationGroup 복제 그룹에 추가합니다. 읽기 전용 복제본의 노드 유형, 파라미터 그룹, 보안 그룹, 유지 관리 기간 및 기타 설정이 myReplicationGroup의 다른 노드와 동일해집니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=CreateCacheCluster
&CacheClusterId=myReadReplica
&ReplicationGroupId=myReplicationGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 읽기 전용 복제본 승격

&X-Amz-Credential=<credential>

API를 사용하여 읽기 전용 복제본을 추가하는 방법에 대한 자세한 내용은 Amazon ElastiCache API Reference의 CreateCacheCluster를 참조하십시오.

Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복 제본 삭제

다음 주제의 정보는 Redis(클러스터 모드 비활성화됨) 복제 그룹에만 적용됩니다.

Redis 복제 그룹에서 읽기 트래픽이 변경되면 읽기 전용 복제본을 추가하거나 제거하려고 할 수 있습니다. Redis(클러스터 모드 비활성화됨) 복제 그룹에서 노드를 제거하는 것은 클러스터를 삭제하는 것과 동일하지 만 다음과 같은 제한이 있습니다.

- 복제 그룹에서 기본을 제거할 수 없습니다. 기본을 삭제하려면 다음을 수행하십시오.
 - 1. 읽기 전용 복제본을 기본으로 승격합니다. 기본으로 읽기 전용 복제본 승격에 대한 자세한 내용은 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본을 기본으로 승격 (p. 178)을 참조하십시오.
 - 2. 이전 기본을 삭제합니다. 이 메서드에 대한 제한 사항은 다음 요점을 참조하십시오.
- 복제 그룹에서 다중 AZ가 활성화된 경우 이 복제 그룹에서 마지막 읽기 전용 복제본을 제거할 수 없습니다. 이 경우 다음과 같이 합니다.
 - 1. 복제 그룹을 수정하여 다중 AZ를 비활성화합니다. 자세한 내용은 복제 그룹 수정 (p. 165) 단원을 참 조하십시오.
 - 2. 읽기 전용 복제본을 삭제합니다.

ElastiCache 콘솔, ElastiCache용 AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 비활성화됨) 복제 그룹에서 읽기 전용 복제본을 제거할 수 있습니다.

Redis 복제 그룹의 클러스터 삭제에 대한 지침은 다음을 참조하십시오.

- AWS Management 콘솔 사용 (p. 102)
- AWS CLI 사용 (p. 103)
- ElastiCache API 사용 (p. 103)
- Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241)
- 샤드의 복제본 수 줄이기 (p. 172)

Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽 기 전용 복제본을 기본으로 승격

다음 주제의 정보는 Redis(클러스터 모드 비활성화됨) 복제 그룹에만 적용됩니다.

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용해 Redis(클러스터 모드 비활성화됨) 읽기 전용 복제본을 기본으로 승격할 수 있습니다. 자동 장애 조치가 포함된 다중 AZ가 Redis(클러스터 모드 비활성화됨) 복제 그룹에서 활성화되어 있는 동안에는 읽기 전용 복제본을 기본으로 승격할 수 없습니다. 다중 AZ가 활성화된 복제 그룹에서 Redis(클러스터 모드 비활성화됨) 복제본을 기본으로 승격하려면 다음을 수행하십시오.

- 1. 다중 AZ를 비활성하도록 복제 그룹을 수정합니다(수정할 때 모든 클러스터가 동일 가용 영역에 있을 필요는 없음). 자세한 내용은 복제 그룹 수정 (p. 165) 단원을 참조하십시오.
- 2. 읽기 전용 복제본을 기본으로 승격합니다.

3. 다중 AZ를 다시 활성화하도록 복제 그룹을 수정합니다.

Redis 2.6.13 이하 버전을 실행하는 복제 그룹에서는 자동 장애 조치가 포함된 다중 AZ를 사용할 수 없습니다.

AWS Management 콘솔 사용

다음 절차에서는 콘솔을 사용해 복제본 노드를 기본 노드로 승격합니다.

읽기 전용 복제본을 기본으로 승격하려면 (console)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 승격하려는 복제본이 다중 AZ가 활성화된 Redis(클러스터 모드 비활성화됨) 복제 그룹의 멤버이면 계속 진행하기 전에 복제 그룹을 수정하여 다중 AZ를 비활성화해야 합니다. 자세한 내용은 복제 그룹 수정 (p. 165) 단원을 참조하십시오.
- 3. Redis를 선택한 후 클러스터 목록에서 수정할 복제 그룹을 선택합니다. 이 복제 그룹은 "Clustered Redis" 엔진이 아닌 "Redis" 엔진에서 실행되어야 하며, 2개 이상의 노드가 있어야 합니다.
- 4. 노드 목록에서 기본으로 승격할 복제본 노드를 선택한 후 작업에서 Promote(승격)를 선택합니다.
- 5. Promote Read Replica(읽기 전용 복제본 승격) 대화 상자에서 다음을 수행합니다.
 - a. Apply Immediately(즉시 적용)에서 예를 선택하여 읽기 전용 복제본을 즉시 승격하거나 아니요를 선택하여 클러스터의 다음 번 유지 관리 기간에 승격합니다.
 - b. [Promote]를 선택하여 읽기 전용 복제본을 승격하거나 [Cancel]을 선택하여 작업을 취소합니다.
- 6. 승격 프로세스를 시작하기 전에 클러스터에 다중 AZ가 활성화된 경우 복제 그룹의 상태가 사용 가능으로 될 때까지 기다린 후 클러스터를 수정하여 다중 AZ를 재활성화합니다. 자세한 내용은 복제 그룹 수정 (p. 165) 단원을 참조하십시오.

AWS CLI 사용

복제 그룹에 다중 AZ가 활성화되어 있으면 읽기 전용 복제본을 기본으로 승격할 수 없습니다. 일부 경우에 승격하려는 복제본은 다중 AZ가 활성화되어 있는 복제 그룹의 일원일 수 있습니다. 이러한 경우 계속하기 전에 다중 AZ를 비활성화하도록 복제 그룹을 수정해야 합니다. 수정할 때 모든 클러스터가 동일 가용 영역에 있을 필요는 없습니다. 복제 그룹 수정에 대한 자세한 내용은 복제 그룹 수정 (p. 165)을 참조하십시오.

다음 AWS CLI 명령은 복제 그룹 sample-repl-group을 수정하여 읽기 전용 복제본 my-replica-1을 복제 그룹의 기본으로 만듭니다.

Linux. macOS. or Unix의 경우:

```
aws elasticache modify-replication-group \
    --replication-group-id sample-repl-group \
    --primary-cluster-id my-replica-1
```

Windows의 경우:

```
aws elasticache modify-replication-group ^
    --replication-group-id sample-repl-group ^
    --primary-cluster-id my-replica-1
```

복제 그룹 수정에 대한 자세한 내용은 Amazon ElastiCache 명령줄 레퍼런스의 modify-replication-group을 참조하십시오.

ElastiCache API 사용

복제 그룹에 다중 AZ가 활성화되어 있으면 읽기 전용 복제본을 기본으로 승격할 수 없습니다. 일부 경우에 승격하려는 복제본은 다중 AZ가 활성화되어 있는 복제 그룹의 일원일 수 있습니다. 이러한 경우 계속하기 전에 다중 AZ를 비활성화하도록 복제 그룹을 수정해야 합니다. 수정할 때 모든 클러스터가 동일 가용 영역에 있을 필요는 없습니다. 복제 그룹 수정에 대한 자세한 내용은 복제 그룹 수정 (p. 165)을 참조하십시오.

다음 ElastiCache API 작업은 읽기 전용 복제본 myReplica-1을 복제 그룹의 기본으로 만드는 복제 그룹 myReplGroup을 수정합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=ModifyReplicationGroup
&ReplicationGroupId=myReplGroup
&PrimaryClusterId=myReplica-1
&Version=2014-12-01
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Signature=<signature>

복제 그룹 수정에 대한 자세한 내용은 Amazon ElastiCache API Reference의 ModifyReplicationGroup을 참 조하십시오.

연결 엔드포인트 찾기

애플리케이션에서 엔드포인트를 사용하여 클러스터에 연결합니다. 엔드포인트는 노드나 클러스터의 고유한 주소입니다.

사용할 엔드포인트

- Redis 독립 실행형 노드 읽기 작업과 쓰기 작업에 모두 노드의 엔드포인트를 사용합니다.
- Redis(클러스터 모드 비활성화됨) 클러스터에서는 모든 쓰기 작업에 기본 엔드포인트를 사용합니다. 리더 엔드포인트를 사용하여 모든 읽기 전용 복제본 사이에 수신 연결을 고르게 분할합니다. 읽기 작업에는 개별 노드 엔드포인트(API/CLI에서는 읽기 엔드포인트라고 함)를 사용하십시오.
- Redis(클러스터 모드 활성화됨) 클러스터에서는 모든 작업에 클러스터의 구성 엔드포인트를 사용합니다. Redis 클러스터를 지원하는 클라이언트를 사용해야 합니다(Redis 3.2). 개별 노드 엔드포인트(API/CLI에 서는 읽기 엔드포인트라고 함)에서 계속 읽을 수 있습니다.

다음 단원에서는 실행 중인 엔진에 필요한 엔드포인트를 찾는 방법을 안내합니다.

주제

- Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 181)
- Redis(클러스터 모드 활성화됨) 클러스터 엔드포인트 찾기 (콘솔) (p. 182)
- 엔드포인트 찾기 (AWS CLI) (p. 184)

• 엔드포인트 찾기 (ElastiCache API) (p. 186)

Redis(클러스터 모드 비활성화됨) 클러스터 엔드포인트 찾기 (콘솔)

Redis(클러스터 모드 비활성화됨) 클러스터에 노드가 하나 뿐이면 노드의 엔드포인트가 읽기와 쓰기에 모두 사용됩니다. Redis(클러스터 모드 비활성화됨) 클러스터에는 여러 노드가 있는 경우, 세 가지 유형의 엔드포 인트(기본 엔드포인트, 리더 엔드포인트 및 노드 엔드포인트)가 있습니다.

기본 엔드포인트는 항상 클러스터의 기본 노드로 확인되는 DNS 이름입니다. 기본 엔드포인트는 읽기 전용 복제본을 기본 역할로 승격하는 것과 같은 클러스터 변경의 영향을 받지 않습니다. 쓰기 활동의 경우 애플리 케이션을 직접 기본에 연결하는 대신 기본 엔드포인트에 연결하는 것이 좋습니다.

리더 엔드포인트는 Redis용 ElastiCache 클러스터의 모든 읽기 전용 복제본에 대한 수신 연결을 고르게 분할합니다. 애플리케이션이 연결을 생성하는 시기 또는 애플리케이션에서 연결을 다시 사용하는 방법과 같은 추가 요소가 트래픽 분산을 결정합니다. 리더 엔드포인트는 복제본이 추가 또는 제거되는 클러스터의 변경 사항을 실시간으로 반영합니다. Redis용 ElastiCache 클러스터의 여러 읽기 전용 복제본을 다양한 AWS 가용영역(AZ)에 두어 리더 엔드포인트의 가용성을 높일 수 있습니다.

읽기 활동의 경우 애플리케이션은 클러스터의 어떤 노드에도 연결할 수 있습니다. 기본 엔드포인트와 달리, 노드 엔드포인트는 특정 엔드포인트로 확인됩니다. 복제본을 추가하거나 삭제하는 것과 같이 클러스터를 변 경하면 애플리케이션에서 노드 엔드포인트를 업데이트해야 합니다.

Redis(클러스터 모드 비활성화됨) 클러스터의 엔드포인트를 찾으려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.

클러스터 화면에 Redis(클러스터 모드 비활성화됨) 및 Redis(클러스터 모드 활성화됨) 클러스터 목록이나타납니다.

3. 클러스터의 기본 엔드포인트 및/또는 리더 엔드포인트를 찾으려면 클러스터 이름 왼쪽의 상자를 선택합니다.



Redis(클러스터 모드 비활성화됨) 클러스터의 기본 및 리더 엔드포인트

클러스터에 노드가 하나 뿐이면 기본 엔드포인트가 없으며 다음 단계에서 계속할 수 있습니다.

4. Redis(클러스터 모드 비활성화됨) 클러스터에 복제본 노드가 있으면 클러스터 이름을 선택하여 클러스터의 복제본 노드 엔드포인트를 찾을 수 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Redis(클러스터 모드 활성화됨) 엔 드포인트(콘솔)의 엔드포인트 찾기

프로먼트(C ᆯ)의 먼트로먼트 늦게 노드 화면에 클러스터의 각 노드, 기본 및 복제본이 엔드포인트와 함께 나열됩니다.

Node Name	▲ Status	Current Role	Port	Endpoint	
test-no-001	available	primary	6379	8	amazo
test-no-002	available	replica	6379	s	amazo
test-no-003	available	replica	6379	8	amazo

Redis(클러스터 모드 비활성화됨) 클러스터의 노드 엔드포인트

- 5. 엔드포인트를 클립보드에 복사하려면
 - a. 한 번에 하나씩 엔드포인트를 찾고 복사할 엔드포인트를 선택합니다.
 - b. 강조 표시한 엔드포인트를 마우스 오른쪽 버튼으로 클릭한 후 컨텍스트 메뉴에서 [Copy]를 선택합니다.

강조 표시한 엔드포인트가 클립보드에 복사됩니다.

Redis(클러스터 모드 비활성화됨) 기본 엔드포인트는 다음과 같습니다. 전송 중 데이터 암호화가 활성화되어 있는지 여부에 따라 차이가 있습니다.

전송 중 데이터 암호화가 비활성화된 경우

clusterName.xxxxxx.nodeId.regionAndAz.cache.amazonaws.com:port
redis-01.7abc2d.0001.usw2.cache.amazonaws.com:6379

전송 중 데이터 암호화가 활성화된 경우

master.clusterName.xxxxxx.regionAndAz.cache.amazonaws.com:port
master.ncit.ameaqx.use1.cache.amazonaws.com:6379

Redis(클러스터 모드 활성화됨) 클러스터 엔드포인트 찾기 (콘솔)

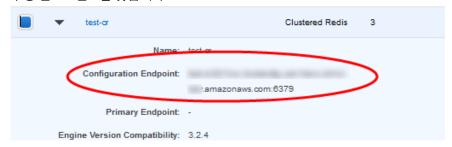
읽기 작업과 쓰기 작업에 모두 구성 엔드포인트를 사용하십시오. Redis가 액세스할 클러스터 노드를 결정합니다.

다음 절차에서는 Redis(클러스터 모드 활성화됨) 클러스터 엔드포인트를 찾고 복사하는 방법을 보여줍니다.

Redis(클러스터 모드 활성화됨) 클러스터의 구성 엔드포인트를 찾으려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
 - 어느 Redis 버전이라도 실행하는 클러스터 목록이 표시됩니다.
- 3. 클러스터 목록에서 "클러스터링된 Redis"를 실행하는 클러스터 왼쪽의 상자를 선택합니다.
 - 화면이 확장되면 선택한 클러스터의 세부 정보가 표시됩니다.

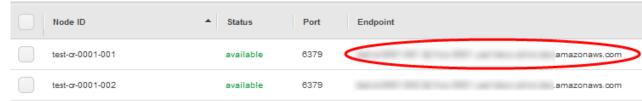
4. 구성 엔드포인트를 찾습니다.



Redis(클러스터 모드 활성화됨) 클러스터의 구성 엔드포인트

Redis(클러스터 모드 활성화됨) 클러스터의 노드 엔드포인트를 찾으려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
 - 어느 Redis 버전이라도 실행하는 클러스터 목록이 표시됩니다.
- 3. 클러스터 목록에서 "클러스터링된 Redis"를 실행하는 클러스터의 클러스터 이름을 선택합니다.
 - 샤드 페이지가 열립니다.
- 4. 노드 엔드포인트의 샤드 이름을 선택합니다.
 - 샤드의 노드 목록에 각 노드의 엔드포인트가 표시됩니다.
- 5. Endpoint 열을 찾고 각 노드의 엔드포인트를 읽습니다.



Redis(클러스터 모드 활성화됨) 클러스터의 노드 엔드포인트

엔드포인트를 클립보드에 복사하려면

- 1. 위에 나온 절차를 사용하여 복사할 엔드포인트를 찾습니다.
- 2. 복사할 엔드포인트를 강조 표시합니다.
- 3. 강조 표시한 엔드포인트를 마우스 오른쪽 버튼으로 클릭하고 컨텍스트 메뉴에서 [Copy]를 선택합니다. 강조 표시한 엔드포인트가 클립보드에 복사됩니다.

Redis(클러스터 모드 활성화됨) 구성 엔드포인트는 다음과 같습니다.

전송 중 데이터 암호화가 비활성화된 경우

```
clusterName.xxxxxx.regionAndAz.cache.amazonaws.com:port
rce.ameaqx.use1.cache.amazonaws.com:6379
```

전송 중 데이터 암호화가 활성화된 경우

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 엔드포인트 찾기 (AWS CLI)

```
clustercfg.clusterName.xxxxxx.regionAndAz.cache.amazonaws.com:port
clustercfg.rce.ameaqx.use1.cache.amazonaws.com:6379
```

엔드포인트 찾기 (AWS CLI)

Amazon ElastiCache용 AWS CLI를 사용하여 노드, 클러스터 및 복제 그룹의 엔드포인트를 찾을 수 있습니다.

주제

- 노드 및 클러스터의 엔드포인트 찾기 (AWS CLI) (p. 184)
- 복제 그룹의 엔드포인트 찾기 (AWS CLI) (p. 185)

노드 및 클러스터의 엔드포인트 찾기 (AWS CLI)

AWS CLI 사용하여 describe-cache-clusters 명령으로 클러스터 및 해당 노드의 엔드포인트를 찾을수 있습니다. Redis 클러스터의 경우 명령이 클러스터 엔드포인트를 반환합니다. 또한 선택적 파라미터 --show-cache-node-info를 포함할 경우 명령이 클러스터에 있는 개별 노드의 엔드포인트를 반환합니다.

Example

다음 명령을 통해 단일 노드 Redis(클러스터 모드 비활성화됨) 클러스터인 mycluster의 클러스터 정보를 검 색할 수 있습니다.

Important

파라미터 --cache-cluster-id는 Redis 복제 그룹에서 단일 노드 Redis(클러스터 모드 비활 성화됨) 클러스터 ID 또는 특정 노드와 함께 사용할 수 있습니다. Redis 복제 그룹의 --cachecluster-id는 0001와 같은 4자리 값입니다. --cache-cluster-id가 Redis 복제 그룹에 있는 클러스터(노드)의 ID인 경우 replication-group-id가 출력에 포함됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-cache-clusters \
    --cache-cluster-id redis-cluster \
    --show-cache-node-info
```

Windows의 경우:

```
aws elasticache describe-cache-clusters ^
--cache-cluster-id redis-cluster ^
--show-cache-node-info
```

위 작업의 출력은 다음과 같습니다(JSON 형식).

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 엔드포인트 찾기 (AWS CLI)

```
"CacheNodeCreateTime": "2018-04-25T18:19:28.241Z",
                    "CacheNodeStatus": "available",
                    "CacheNodeId": "0001",
                    "Endpoint": {
                         "Address": "redis-cluster.ameaqx.0001.use1.cache.amazonaws.com",
                        "Port": 6379
                    "ParameterGroupStatus": "in-sync"
                }
            ٦,
            "AtRestEncryptionEnabled": false,
            "CacheClusterId": "redis-cluster",
            "TransitEncryptionEnabled": false,
            "CacheParameterGroup": {
                "ParameterApplyStatus": "in-sync",
                "CacheNodeIdsToReboot": [],
                "CacheParameterGroupName": "default.redis3.2"
            "NumCacheNodes": 1,
            "PreferredAvailabilityZone": "us-east-1b",
            "AutoMinorVersionUpgrade": true,
            "Engine": "redis",
            "AuthTokenEnabled": false,
            "PendingModifiedValues": {},
            "PreferredMaintenanceWindow": "tue:08:30-tue:09:30",
            "CacheSecurityGroups": [],
            "CacheSubnetGroupName": "default",
            "CacheNodeType": "cache.t2.small",
            "EngineVersion": "3.2.10",
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
            "CacheClusterCreateTime": "2018-04-25T18:19:28.241Z"
        }
    ]
}
```

자세한 내용은 describe-cache-clusters 단원을 참조하십시오.

복제 그룹의 엔드포인트 찾기 (AWS CLI)

AWS CLI를 사용하여 describe-replication-groups 명령으로 복제 그룹 및 해당 클러스터의 엔드포 인트를 찾을 수 있습니다. 이 명령은 리더 엔드포인트와 함께, 복제 그룹의 기본 엔드포인트와 복제 그룹에 있는 모든 클러스터(노드)의 목록 및 해당 엔드포인트를 반환합니다.

다음 작업은 복제 그룹 myrep1group의 기본 엔드포인트 및 리더 엔드포인트를 검색합니다. 모든 쓰기 작업에 기본 엔드포인트를 사용합니다.

```
aws elasticache describe-replication-groups \
--replication-group-id myreplgroup
```

Windows의 경우:

```
aws elasticache describe-replication-groups ^
--replication-group-id myreplgroup
```

이 작업의 출력은 다음과 같습니다(JSON 형식).

```
{
    "ReplicationGroups": [
    {
        "Status": "available",
        "Description": "test",
```

```
"NodeGroups": [
            "Status": "available",
               "NodeGroupMembers": [
                     "CurrentRole": "primary",
                     "PreferredAvailabilityZone": "us-west-2a",
                     "CacheNodeId": "0001",
                     "ReadEndpoint": {
                        "Port": 6379,
                        "Address": "myreplgroup-001.1abc4d.0001.usw2.cache.amazonaws.com"
                     "CacheClusterId": "myreplgroup-001"
                  },
                     "CurrentRole": "replica",
                     "PreferredAvailabilityZone": "us-west-2b",
                     "CacheNodeId": "0001",
                     "ReadEndpoint": {
                        "Port": 6379,
                        "Address": "myreplgroup-002.1abc4d.0001.usw2.cache.amazonaws.com"
                     "CacheClusterId": "myreplgroup-002"
                  },
                     "CurrentRole": "replica",
                     "PreferredAvailabilityZone": "us-west-2c",
                     "CacheNodeId": "0001",
                     "ReadEndpoint": {
                        "Port": 6379,
                        "Address": "myreplgroup-003.1abc4d.0001.usw2.cache.amazonaws.com"
                     "CacheClusterId": "myreplgroup-003"
                  }
               ٦,
               "NodeGroupId": "0001",
               "PrimaryEndpoint": {
                  "Port": 6379,
                  "Address": "myreplgroup.1abc4d.ng.0001.usw2.cache.amazonaws.com"
               "ReaderEndpoint": {
                  "Port": 6379,
                  "Address": "myreplgroup-ro.1abc4d.ng.0001.usw2.cache.amazonaws.com"
         ],
         "ReplicationGroupId": "myreplgroup",
         "AutomaticFailover": "enabled",
         "SnapshottingClusterId": "myreplgroup-002",
         "MemberClusters": [
            "myreplgroup-001",
            "myreplgroup-002",
            "myreplgroup-003"
         "PendingModifiedValues": {}
      }
   ]
}
```

자세한 내용은 AWS CLI Command Reference의 describe-replication-groups 항목을 참조하십시오.

엔드포인트 찾기 (ElastiCache API)

Amazon ElastiCache API 사용하여 노드, 클러스터 및 복제 그룹의 엔드포인트를 찾을 수 있습니다.

주제

- 노드 및 클러스터의 엔드포인트 찾기 (ElastiCache API) (p. 187)
- 복제 그룹의 엔드포인트 찾기 (ElastiCache API) (p. 187)

노드 및 클러스터의 엔드포인트 찾기 (ElastiCache API)

ElastiCache API를 사용하여 DescribeCacheClusters 작업으로 클러스터 및 해당 노드의 엔드포인트를 찾을 수 있습니다. Redis 클러스터의 경우 명령이 클러스터 엔드포인트를 반환합니다. 또한 선택적 파라미 터 ShowCacheNodeInfo를 포함할 경우 작업이 클러스터에 있는 개별 노드의 엔드포인트를 반환합니다.

Example

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheClusters
&CacheClusterId=mycluster
&ShowCacheNodeInfo=true
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>

복제 그룹의 엔드포인트 찾기 (ElastiCache API)

ElastiCache API를 사용하여 DescribeReplicationGroups 작업으로 복제 그룹 및 해당 클러스터의 엔 드포인트를 찾을 수 있습니다. 이 작업은 리더 엔드포인트와 함께, 복제 그룹의 기본 엔드포인트와 복제 그룹 에 있는 모든 클러스터의 목록 및 해당 엔드포인트를 반환합니다.

다음 작업은 복제 그룹 myrep1group의 기본 엔드포인트(PrimaryEndpoint), 리더 엔드포인트 (ReaderEndpoint) 및 개별 노드 엔드포인트(ReadEndpoint)를 검색합니다. 모든 쓰기 작업에 기본 엔드포인트를 사용합니다.

자세한 정보는 DescribeReplicationGroups를 참조하십시오.

Redis용 ElastiCache 백업 및 복원

Redis를 실행하는 Amazon ElastiCache 클러스터는 데이터를 백업할 수 있습니다. 클러스터를 복원하거나 새 클러스터를 시드하기 위해 백업을 사용할 수 있습니다. 백업은 클러스터의 모든 데이터와 클러스터의 메타데이터로 구성됩니다. 모든 백업은 Amazon Simple Storage Service(Amazon S3)에 쓰여지므로 내구성 있는 스토리지가 확보됩니다. 언제든지 새로운 Redis 클러스터를 생성하고 백업의 데이터로 클러스터를 채워 데이터를 복원할 수 있습니다. ElastiCache를 통해 AWS Management 콘솔, AWS Command Line Interface(AWS CLI) 및 ElastiCache API를 사용하여 백업을 관리할 수 있습니다.

Redis 버전 2.8.22부터는 사용 가능한 메모리에 따라 백업 방법을 선택합니다. 사용 가능한 메모리가 충분하면 캐시가 백업되는 동안 캐시의 예약된 메모리에 모든 변경 사항을 쓰는 하위 프로세스가 생성됩니다. 백업

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Constraints

프로세스 중에 캐시에 쓰는 수에 따라 이 하위 프로세스가 모든 예약된 메모리를 소비하므로 백업이 실패할 수 있습니다.

사용 가능한 메모리가 부족하면 forkless 협조 백그라운드 프로세스가 적용됩니다. forkless 방법은 지연 시간과 처리량 모두에 영향을 줄 수 있습니다. 자세한 내용은 동기화 및 백업 구현 방법 (p. 143) 단원을 참조하십시오.

백업 프로세스의 성능 영향에 대한 자세한 내용은 백업의 성능 영향 (p. 189) 단원을 참조하십시오.

다음은 백업 및 복원 작업의 개요입니다.

Important

드물지만 백업 프로세스에서 최종 백업을 포함하여 백업 생성에 실패하는 경우가 있습니다. 예약된 메모리가 부족하여 백업이 실패하기도 합니다. 따라서 백업을 시도하기 전에 예약된 메모리를 충분히 확보해야 합니다. 메모리가 부족하면 일부 키를 제거하거나 reserved-memory-percent의 값을 늘릴 수 있습니다.

자세한 내용은 다음 자료를 참조하십시오.

- 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)
- 예약된 메모리 관리 (p. 405)

클러스터를 삭제하려고 하고 데이터를 보존하는 것이 중요한 경우 추가적인 예방 조치를 취할 수 있습니다. 이렇게 하려면 먼저 수동 백업을 생성하고 사용 가능한 상태인지 확인한 다음 클러스터를 삭제합니다. 이렇게 하면 백업에 실패하더라도 클러스터 데이터를 계속 사용할 수 있습니다. 앞서 설명한 모범 사례에 따라 백업을 다시 시도할 수 있습니다.

주제

- 백업 제약 조건 (p. 188)
- 백업 비용 (p. 189)
- 백업의 성능 영향 (p. 189)
- 자동 백업 예약 (p. 190)
- 수동 백업 만들기 (p. 190)
- 최종 백업 생성 (p. 195)
- 백업 설명 (p. 198)
- 백업 복사 (p. 199)
- 백업 내보내기 (p. 201)
- 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208)
- 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211)
- 백업 태그 지정 (p. 217)
- 백업 삭제 (p. 217)
- Redis용 ElastiCache의 AOF(Append Only Files) (p. 218)

백업 제약 조건

백업을 계획하거나 만들려는 경우 다음 제약 조건을 고려해야 합니다.

- 이때 Redis에서 실행되는 클러스터에 대해서만 백업 및 복원이 지원됩니다.
- Redis(클러스터 모드 비활성화됨) 클러스터의 경우 cache.t1.micro 노드에서 백업 및 복원이 지원되지 않습니다. 다른 모든 캐시 노드 유형은 지원됩니다.
- Redis(클러스터 모드 활성화됨) 클러스터의 경우 모든 노드 유형에 대해 백업 및 복원이 지원됩니다.
- 24시간 연속으로 클러스터의 노드당 20개 이내의 수동 백업을 만들 수 있습니다.

- Redis(클러스터 모드 활성화됨)는 클러스터 수준(API 또는 CLI의 경우 복제 그룹 수준)에서만 백업 생성을 지원합니다. Redis(클러스터 모드 활성화됨)는 샤드 수준(API 또는 CLI의 경우 노드 그룹 수준)에서는 백업 생성을 지원하지 않습니다.
- 백업 프로세스 중에는 클러스터에서 다른 API 또는 CLI 작업을 실행할 수 없습니다.

백업 비용

ElastiCache를 사용하여 활성 Redis 클러스터마다 백업 하나를 무료로 저장할 수 있습니다. 추가 백업을 위한 스토리지 공간은 모든 AWS 리전에 대해 매월 0.085 USD/GB의 요금이 청구됩니다. 백업을 만들거나 백업의 데이터를 Redis 클러스터에 복원하는 경우에는 데이터 전송 요금이 없습니다.

백업의 성능 영향

백업 프로세스는 실행 중인 Redis 버전에 따라 결정됩니다. Redis 2.8.22부터는 프로세스가 forkless입니다.

Redis 2.8.22 이상을 실행하는 경우 백업

버전 2.8.22 이상에서 Redis 백업은 두 가지 방법 중 하나를 선택합니다. forked 백업을 지원할 메모리가 부족할 경우 ElastiCache에서 협조 백그라운드 처리를 사용하는 forkless 방법을 사용합니다. forked 저장 프로세스를 지원할 메모리가 충분할 경우 이전의 Redis 버전과 동일한 프로세스가 사용됩니다.

forkless 백업 중에 쓰기 로드가 높으면 캐시에 대한 쓰기가 지연됩니다. 이렇게 지연되면 변경 사항이 너무 많이 누적되지 않았는지 확인하므로 백업이 성공적으로 수행되지 않습니다.

2.8.22 이전 Redis 버전을 실행하는 경우 백업

Redis의 기본 BGSAVE 작업을 사용하여 백업이 생성됩니다. 캐시 노드의 Redis 프로세스에서 하위 프로세스를 생성하여 캐시에서 Redis .rdb 파일로 모든 데이터를 씁니다. 하위 프로세스를 생성하는 데 최대 10초가 걸릴 수 있습니다. 이 시간 동안 상위 프로세스는 들어오는 애플리케이션 요청을 수락할 수 없습니다. 하위 프로세스가 독립적으로 실행된 후 상위 프로세스에서 일반적인 작업을 다시 시작합니다. 백업 작업이 완료되면 하위 프로세스가 생깁니다.

백업을 쓰는 동안 추가 캐시 노드 메모리가 새로운 쓰기에 사용됩니다. 이 추가 메모리 사용량이 노드의 사용 가능한 메모리를 초과하면 과도한 페이징으로 인해 처리가 느려지거나 실패할 수 있습니다.

백업 성능 개선

다음은 백업 성능을 개선하기 위한 지침입니다.

- reserved-memory-percent 파라미터 설정 과도한 페이징을 완화하려면 reserved-memory-percent 파라미터를 설정하는 것이 좋습니다. 이 파라미터를 사용하면 Redis가 노드의 모든 사용 가능한 메모리를 소비하고 페이징 양을 줄이는 데 도움이 됩니다. 더 큰 노드를 사용하기만 해도 성능 개선을 확인할 수 있습니다. reserved-memory 및 reserved-memory-percent 파라미터에 대한 자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.
- 읽기 전용 복제본으로 백업을 만듭니다. 노드가 2개 이상인 노드 그룹에서 Redis를 실행하는 경우 기본 노드 또는 읽기 전용 복제본 중 하나에서 백업을 만들 수 있습니다. BGSAVE 도중 필요한 시스템 리소스로 인해 읽기 전용 복제본 중 하나에서 백업을 생성하는 것이 좋습니다. 복제본으로 백업을 생성하는 동안 기 본 노드는 BGSAVE 리소스 요구 사항의 영향을 받지 않습니다. 기본 노드는 속도를 늦추지 않고 계속해서 요청을 처리할 수 있습니다.

복제 그룹을 삭제하고 최종 백업을 요청할 경우 ElastiCache에서는 언제나 기본 노드에서 백업을 만듭니다. 그러면 복제 그룹이 삭제되기 전에 가장 최신 Redis 데이터를 캡처할 수 있습니다.

자동 백업 예약

모든 Redis 클러스터에 자동 백업을 활성화할 수 있습니다. 자동 백업이 활성화되면 ElastiCache에서 매일 클러스터 백업을 생성합니다. 자동 백업은 데이터 손실을 막는 데 도움이 됩니다. 실패할 경우 새로운 클러스터를 생성해 최신 백업에서 모든 데이터를 복원할 수 있습니다. 그 결과 웜 부팅 클러스터가 되고 데이터가 사전 로드되어 사용할 수 있게 됩니다. 자세한 내용은 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.

자동 백업을 예약할 때는 다음 설정을 계획해야 합니다.

• 백업 기간 – ElastiCache에서 백업을 만들기 시작하는 하루 중 기간입니다. 백업 기간의 최소 길이는 60분입니다. 아무 때나 편리한 시간 또는 하루 중 백업을 피할 시간, 특히 사용량이 많은 기간의 백업 기간을 설정할 수 있습니다.

백업 기간을 지정하지 않으면 ElastiCache에서 자동으로 할당합니다.

• 백업 보존 제한 – Amazon S3에서 백업을 보존되는 일수입니다. 예를 들어, 보존 제한을 5로 설정하면 오늘 만든 백업이 5일간 보존됩니다. 보존 제한이 만료되면 백업이 자동으로 삭제됩니다.

최대 백업 보존 제한은 35일입니다. 백업 보존 제한을 0으로 설정하면 클러스터의 자동 백업이 비활성화됩니다.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용해 수정하여 기존의 Redis 클러스터나 보존 그룹에서 자동 백업을 활성화하거나 비활성화할 수 있습니다. 기존 클러스터나 복제 그룹에서 자동 백업을 활성화하거나 비활성화하는 방법에 대한 자세한 내용은 ElastiCache 클러스터 수정 (p. 87) 또는 복제 그룹 수정 (p. 165) 단원을 참조하십시오.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis 클러스터나 보존 그룹을 만들 때 자동 백업을 활성화하거나 비활성화할 수 있습니다. [Advanced Redis Settings] 단원에서 [Enable Automatic Backups] 상자를 선택하여 Redis 클러스터를 만들 때 자동 백업을 활성화할 수 있습니다. 자세한 내용은 Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69)의 2단계를 참조하십시오. 기존의 클러스터를 기본 클러스터로 사용하지 않는 경우 Redis 복제 그룹을 만들 때 자동 백업을 활성화할 수 있습니다. 자세한 내용은 Redis 복제 그룹을 처음부터 새로 생성 (p. 148) 단원을 참조하십시오.

수동 백업 만들기

자동 백업 외에도 언제든지 수동 백업을 만들 수 있습니다. 지정한 보존 기간 후에 자동으로 삭제되는 자동백업과 달리 수동 백업에는 나중에 자동으로 삭제되는 보존 기간이 없습니다. 수동 백업은 수동으로 삭제해야 합니다. 클러스터나 노드를 삭제하더라도 해당 클러스터나 노드의 수동 백업은 보존됩니다. 수동 백업을 더 이상 보존하지 않으려면 이 백업을 직접 명시적으로 삭제해야 합니다.

수동 백업은 테스트와 보관에 유용합니다. 예를 들어, 테스트 목적으로 기준 데이터 세트를 개발했다고 가정해 보겠습니다. 데이터의 수동 백업을 만들고 언제든지 복원할 수 있습니다. 데이터를 수정하는 애플리케이션을 테스트한 후 새로운 클러스터를 만들고 기준 백업에서 복원하여 데이터를 재설정할 수 있습니다. 클러스터가 준비되면 기준 데이터와 비교하여 애플리케이션을 다시 테스트하고— 필요한 만큼 이 프로세스를 반복할 수 있습니다.

수동 백업을 직접 생성할 뿐 아니라 다음 방법 중 하나로 수동 백업을 생성할 수 있습니다.

- 백업 복사 (p. 199) 소스 백업을 자동으로 생성했는지 수동으로 생성했는지는 중요하지 않습니다.
- 최종 백업 생성 (p. 195) 클러스터나 노드를 삭제하기 직전에 백업을 생성합니다.

기타 가져오기 항목

• 백업 제약 조건 (p. 188)

- 백업 비용 (p. 189)
- 백업의 성능 영향 (p. 189)

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 노드의 수동 백업을 생성할 수 있습니다.

수동 백업 생성 (콘솔)

클러스터 (console)의 백업을 생성하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.

Redis 클러스터 화면이 나타납니다.

- 3. 백업할 Redis 클러스터 이름 왼쪽의 상자를 선택합니다.
- 4. [Backup]을 선택합니다.
- 5. [Create Backup] 대화 상자의 [Backup Name] 상자에 백업 이름을 입력합니다. 이름은 백업된 클러스터 와 백업 날짜 및 시간을 나타내는 것이 좋습니다.

클러스터 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.
- 6. [Create Backup]을 선택합니다.

클러스터 상태가 snapshotting으로 바뀝니다. 상태가 다시 available로 바뀌면 백업이 완료됩니다.

수동 백업 생성 (AWS CLI)

AWS CLI를 사용하여 클러스터의 수동 백업을 생성하려면 다음 파라미터와 함께 create-snapshot AWS CLI 작업을 사용하십시오.

- --cache-cluster-id
 - 백업 중인 클러스터에 복제본 노드가 없으면 --cache-cluster-id는 백업 중인 클러스터의 이름입니다(예: mycluster).
 - 백업 중인 클러스터에 복제본 노드가 하나 이상 있으면 --cache-cluster-id는 백업에 사용하려는 클러스터의 노드 이름입니다. 예를 들면, 이름은 mycluster-002일 수 있습니다.

Redis(클러스터 모드 비활성화됨) 클러스터를 백업할 때에만 이 파라미터를 사용합니다.

- --replication-group-id 백업 원본으로 사용할 Redis(클러스터 모드 활성화됨) 클러스터(CLI/API 의 경우 복제 그룹)의 이름입니다. Redis(클러스터 모드 활성화됨) 클러스터를 백업할 때에만 이 파라미터 를 사용하십시오.
- --snapshot-name 생성할 스냅샷의 이름입니다.

클러스터 명명 제약 조건은 다음과 같습니다.

• 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.

- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

예제 1: 복제본 노드가 없는 Redis(클러스터 모드 비활성화됨) 클러스터 백업

다음 AWS CLI 작업에서 읽기 전용 복제본이 없는 Redis(클러스터 모드 비활성화됨) 클러스터 myNonClusteredRedis로 백업 bkup-20150515을 생성합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-snapshot \
    --cache-cluster-id myNonClusteredRedis \
    --snapshot-name bkup-20150515
```

Windows의 경우:

```
aws elasticache create-snapshot ^
--cache-cluster-id myNonClusteredRedis ^
--snapshot-name bkup-20150515
```

예제 2: 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터 백업

다음 AWS CLI 작업에서 Redis(클러스터 모드 비활성화됨) 클러스터 myNonClusteredRedis에서 백업 bkup-20150515을 생성합니다. 이 백업에는 하나 이상의 읽기 전용 복제본이 있습니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-snapshot \
--cache-cluster-id myNonClusteredRedis-001 \
--snapshot-name bkup-20150515
```

Windows의 경우:

```
aws elasticache create-snapshot ^
--cache-cluster-id myNonClusteredRedis-001 ^
--snapshot-name bkup-20150515
```

예제 출력: Redis(클러스터 모드 비활성화됨)복제본 노드가 있는 클러스터 백업

이 작업의 출력은 다음과 같습니다.

```
"Snapshot": {
    "Engine": "redis",
    "CacheParameterGroupName": "default.redis3.2",
    "VpcId": "vpc-91280df6",
    "CacheClusterId": "myNonClusteredRedis-001",
    "SnapshotRetentionLimit": 0,
    "NumCacheNodes": 1,
    "SnapshotName": "bkup-20150515",
    "CacheClusterCreateTime": "2017-01-12T18:59:48.048Z",
    "AutoMinorVersionUpgrade": true,
    "PreferredAvailabilityZone": "us-east-1c",
    "SnapshotStatus": "creating",
    "SnapshotSource": "manual",
```

예제 3: Redis(클러스터 모드 활성화됨)에 대한 클러스터 백업

다음 AWS CLI 작업에서 Redis(클러스터 모드 활성화됨)클러스터 myClusteredRedis에서 백업 bkup-20150515을 생성합니다. --cache-cluster-id 대신 --replication-group-id를 사용하여 원본을 식별하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-snapshot \
    --replication-group-id myClusteredRedis \
    --snapshot-name bkup-20150515
```

Windows의 경우:

```
aws elasticache create-snapshot ^
--replication-group-id myClusteredRedis ^
--snapshot-name bkup-20150515
```

예제 출력: Redis(클러스터 모드 활성화됨) 클러스터 백업

이 작업의 출력은 다음과 같습니다.

```
{
    "Snapshot": {
        "Engine": "redis",
        "CacheParameterGroupName": "default.redis3.2.cluster.on",
        "VpcId": "vpc-91280df6",
        "NodeSnapshots": [
                "CacheSize": "",
                "NodeGroupId": "0001"
                "CacheSize": "",
                "NodeGroupId": "0002"
        "NumNodeGroups": 2,
        "SnapshotName": "bkup-20150515",
        "ReplicationGroupId": "myClusteredRedis",
        "AutoMinorVersionUpgrade": true,
        "SnapshotRetentionLimit": 1,
        "AutomaticFailover": "enabled",
        "SnapshotStatus": "creating",
        "SnapshotSource": "manual",
```

```
"SnapshotWindow": "10:00-11:00",
    "EngineVersion": "3.2.4",
    "CacheSubnetGroupName": "default",
    "ReplicationGroupDescription": "2 shards 2 nodes each",
    "Port": 6379,
    "PreferredMaintenanceWindow": "sat:03:30-sat:04:30",
    "CacheNodeType": "cache.r3.large"
}
```

관련 주제

자세한 내용은 AWS CLI Command Reference의 create-snapshot을 참조하십시오.

수동 백업 생성 (ElastiCache API)

ElastiCache API를 사용하여 클러스터의 수동 백업을 생성하려면 다음 파라미터와 함께 CreateSnapshot ElastiCache API 작업을 사용하십시오.

- CacheClusterId
 - 백업 중인 클러스터에 복제본 노드가 없으면 CacheClusterId는 백업 중인 클러스터의 이름입니다(예: mycluster).
 - 백업 중인 클러스터에 복제본 노드가 하나 이상 있으면 CacheClusterId는 백업에 사용하려는 클러스터 의 노드 이름입니다(예: mycluster-002).

Redis(클러스터 모드 비활성화됨) 클러스터를 백업할 때에만 이 파라미터를 사용하십시오.

- ReplicationGroupId 백업 원본으로 사용할 Redis(클러스터 모드 활성화됨) 클러스터(CLI/API의 경우 복제 그룹)의 이름입니다. Redis(클러스터 모드 활성화됨) 클러스터를 백업할 때에만 이 파라미터를 사용하십시오.
- SnapshotName 생성할 스냅샷의 이름입니다.

클러스터 명명 제약 조건은 다음과 같습니다.

- 1-40자의 영숫자 또는 하이픈으로 구성되어야 합니다.
- 문자로 시작해야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

API 코드 예제

- 예제 1: 복제본 노드가 없는 Redis(클러스터 모드 비활성화됨) 클러스터 백업 (p. 194)
- 예제 2: 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터 백업 (p. 195)
- 예제 3: Redis(클러스터 모드 활성화됨) 클러스터 백업 (p. 195)
- 관련 주제 (p. 195)

예제 1: 복제본 노드가 없는 Redis(클러스터 모드 비활성화됨) 클러스터 백업

다음 ElastiCache API 작업에서 읽기 전용 복제본이 없는 Redis(클러스터 모드 비활성화됨) 클러스터 myNonClusteredRedis에서 백업 bkup-20150515을 생성합니다.

https://elasticache.us-west-2.amazonaws.com/

?Action=CreateSnapshot
&CacheClusterId=myNonClusteredRedis
&SnapshotName=bkup-20150515
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

예제 2: 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터 백업

다음 ElastiCache API 작업에서 읽기 전용 복제본이 1개 이상인 Redis(클러스터 모드 비활성화됨)클러스터 myNonClusteredRedis에서 백업 bkup-20150515을 생성합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=CreateSnapshot
&CacheClusterId=myNonClusteredRedis-001
&SnapshotName=bkup-20150515
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

예제 3: Redis(클러스터 모드 활성화됨) 클러스터 백업

다음 ElastiCache API 작업에서 Redis(클러스터 모드 활성화됨)클러스터 myClusteredRedis에서 백업 bkup-20150515을 생성합니다. CacheClusterId 대신 ReplicationGroupId를 사용하여 원본을 식별하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=CreateSnapshot
&ReplicationGroupId=myClusteredRedis
&SnapshotName=bkup-20150515
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API Reference의 CreateSnapshot을 참조하십시오.

관련 주제

자세한 내용은 Amazon ElastiCache API Reference의 CreateSnapshot을 참조하십시오.

최종 백업 생성

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 최종 백업을 생성할 수 있습니다.

최종 백업 생성 (콘솔)

ElastiCache 콘솔을 사용하여 Redis 클러스터(API 또는 CLI의 경우 복제 그룹)를 삭제할 때 최종 백업을 만들수 있습니다.

Redis 클러스터를 삭제할 때 최종 백업을 만들려면 대화 상자에서(5단계) [Yes]를 선택하고 백업 이름을 제공합니다.

관련 주제

- AWS Management 콘솔 사용 (p. 102)
- 복제 그룹 (콘솔) 삭제 (p. 167)

최종 백업 생성 (AWS CLI)

AWS CLI를 사용하여 Redis 클러스터(API 또는 CLI의 경우 복제 그룹)를 삭제할 때 최종 백업을 만들 수 있습니다.

주제

- 읽기 전용 복제본이 없는 Redis 클러스터를 삭제하는 경우 (p. 196)
- 읽기 전용 복제본이 있는 Redis 클러스터를 삭제하는 경우 (p. 196)

읽기 전용 복제본이 없는 Redis 클러스터를 삭제하는 경우

최종 백업을 만들려면 다음 파라미터와 함께 delete-cache-cluster AWS CLI 작업을 사용하십시오.

- --cache-cluster-id 삭제 중인 클러스터의 이름입니다.
- --final-snapshot-identifier 백업 이름입니다.

다음 코드는 클러스터 myRedisCluster를 삭제할 때 최종 백업 bkup-20150515-final을 생성합니다.

Linux. macOS. or Unix의 경우:

```
aws elasticache delete-cache-cluster \
--cache-cluster-id myRedisCluster \
--final-snapshot-identifier bkup-20150515-final
```

Windows의 경우:

```
aws elasticache delete-cache-cluster ^
--cache-cluster-id myRedisCluster ^
--final-snapshot-identifier bkup-20150515-final
```

자세한 내용은 AWS CLI Command Reference의 delete-cache-cluster 항목을 참조하십시오.

읽기 전용 복제본이 있는 Redis 클러스터를 삭제하는 경우

복제 그룹을 삭제할 때 최종 백업을 만들려면 다음 파라미터와 함께 delete-replication-group AWS CLI 작업을 사용하십시오.

- --replication-group-id 삭제 중인 복제 그룹의 이름입니다.
- --final-snapshot-identifier 최종 백업 이름입니다.

다음 코드는 복제 그룹 myRep1Group을 삭제할 때 최종 백업 bkup-20150515-final을 만듭니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache delete-replication-group \
    --replication-group-id myReplGroup \
```

--final-snapshot-identifier bkup-20150515-final

Windows의 경우:

```
aws elasticache delete-replication-group ^
--replication-group-id myReplGroup ^
--final-snapshot-identifier bkup-20150515-final
```

자세한 내용은 AWS CLI Command Reference의 delete-replication-group 항목을 참조하십시오.

최종 백업 생성 (ElastiCache API)

ElastiCache API를 사용하여 Redis 클러스터 또는 복제 그룹을 삭제할 때 최종 백업을 만들 수 있습니다.

주제

- Redis 클러스터를 삭제하는 경우 (p. 197)
- Redis 복제 그룹을 삭제하는 경우 (p. 197)

Redis 클러스터를 삭제하는 경우

최종 백업을 만들려면 다음 파라미터와 함께 DeleteCacheCluster ElastiCache API 작업을 사용하십시오.

- CacheClusterId 삭제 중인 클러스터의 이름입니다.
- FinalSnapshotIdentifier 백업 이름입니다.

다음 ElastiCache API 작업에서는 클러스터 myRedisCluster를 삭제할 때 백업 bkup-20150515-final을 생성합니다.

자세한 내용은 Amazon ElastiCache API Reference의 DeleteCacheCluster 항목을 참조하십시오.

Redis 복제 그룹을 삭제하는 경우

복제 그룹을 삭제할 때 최종 백업을 만들려면 다음 파라미터와 함께 DeleteReplicationGroup ElastiCache API 작업을 사용하십시오.

- ReplicationGroupId 삭제 중인 복제 그룹의 이름입니다.
- FinalSnapshotIdentifier 최종 백업 이름입니다.

다음 ElastiCache API 작업에서는 복제 그룹 myReplGroup을 삭제할 때 백업 bkup-20150515-final을 생성합니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DeleteReplicationGroup
```

&FinalSnapshotIdentifier=bkup-20150515-final &ReplicationGroupId=myReplGroup &Version=2015-02-02 &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20150202T192317Z &X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API Reference의 DeleteReplicationGroup 항목을 참조하십시오.

백업 설명

다음 절차는 백업 목록을 표시하는 방법을 보여줍니다. 원한다면 특정 백업의 세부 정보를 볼 수도 있습니다.

백업 설명 (콘솔)

AWS Management 콘솔을 사용하여 백업을 표시하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Backups]를 선택합니다.
- 3. [Filter] 목록을 사용하여 [manual], [automatic] 또는 [all] 백업을 표시합니다.
- 4. 특정 백업의 세부 정보를 보려면 백업 이름 왼쪽의 상자를 선택합니다.

백업 설명 (AWS CLI)

백업 목록을 표시하고 특정 백업에 대한 세부 정보를 선택적으로 표시하려면 describe-snapshots CLI 작업을 사용하십시오.

예시

다음 작업에서는 --max-records 파라미터를 사용하여 계정에 연결된 백업을 20개까지 나열합니다. --max-records 파라미터를 생략하면 백업이 50개까지 나열됩니다.

aws elasticache describe-snapshots --max-records 20

다음 작업에서는 ––cache–cluster–id 파라미터를 사용하여 클러스터 my–cluster에 연결된 백업만 나 열합니다.

aws elasticache describe-snapshots --cache-cluster-id my-cluster

다음 작업에서는 --snapshot-name 파라미터를 사용하여 백업 my-backup의 세부 정보를 표시합니다.

aws elasticache describe-snapshots --snapshot-name my-backup

자세한 내용은 AWS CLI Command Reference의 describe-snapshots를 참조하십시오.

백업 설명 (ElastiCache API)

백업 목록을 표시하려면 DescribeSnapshots 작업을 사용하십시오.

예시

다음 작업에서는 MaxRecords 파라미터를 사용하여 계정에 연결된 백업을 20개까지 나열합니다. MaxRecords 파라미터를 생략하면 백업이 50개까지 나열됩니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=DescribeSnapshots
    &MaxRecords=20
    &SignatureMethod=HmacSHA256
    &SignatureVersion=4
    &Timestamp=20141201T220302Z
    &Version=2014-12-01
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20141201T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Credential=<credential>
    &X-Amz-Signature=<signature>
```

다음 작업에서는 CacheClusterId 파라미터를 사용하여 클러스터 MyCluster에 연결된 모든 백업을 나열합니다.

다음 작업에서는 SnapshotName 파라미터를 사용하여 백업 MyBackup의 세부 정보를 표시합니다.

자세한 내용은 DescribeSnapshots를 참조하십시오.

백업 복사

백업을 자동으로 생성했건 수동으로 생성했건 백업 복사본을 만들 수 있습니다. 또한 백업을 내보낼 수 있으므로 ElastiCache 외부에서 백업에 액세스할 수 있습니다. 백업을 내보내기 위한 지침은 백업 내보내기 (p. 201) 단원을 참조하십시오.

다음 절차에서는 백업을 복사하는 방법을 보여줍니다.

백업 복사 (콘솔)

백업을 복사하려면 (console)

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 백업 목록을 표시하려면 왼쪽 탐색 창에서 [Backups]를 선택합니다.
- 3. 백업 목록에서 복사할 백업의 이름 왼쪽에 있는 상자를 선택합니다.
- 4. [Copy]를 선택합니다.
- 5. 이미지 생성 대화 상자에서 다음 설정을 지정합니다.
 - a. [New backup name] 상자에 새 백업의 이름을 입력합니다.
 - b. 선택적 [Target S3 Bucket] 상자를 비워 둡니다. 이 필드는 백업을 내보내기 위해서만 사용해야 하며 특수한 S3 권한을 필요로 합니다. 백업 내보내기에 대한 정보는 백업 내보내기 (p. 201) 단원을 참 조하십시오.
 - c. [Copy]를 선택합니다.

백업 복사 (AWS CLI)

백업을 복사하려면 copy-snapshot 작업을 사용하십시오.

파라미터

- --source-snapshot-name 복사할 백업의 이름입니다.
- --target-snapshot-name 백업 복사본의 이름입니다.
- --target-bucket 백업을 내보내기 위해 예약됩니다. 백업 복사본을 만들 때 이 파라미터를 사용하지 마십시오. 자세한 내용은 백업 내보내기 (p. 201) 단원을 참조하십시오.

다음 예제에서는 자동 백업 복사본을 만듭니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache copy-snapshot \
--source-snapshot-name automatic.my-redis-primary-2014-03-27-03-15 \
--target-snapshot-name my-backup-copy
```

Windows의 경우:

```
aws elasticache copy-snapshot ^
--source-snapshot-name automatic.my-redis-primary-2014-03-27-03-15 ^
--target-snapshot-name my-backup-copy
```

자세한 내용은 AWS CLI의 copy-snapshot 항목을 참조하십시오.

백업 복사 (ElastiCache API)

백업을 복사하려면 다음 파라미터와 함께 CopySnapshot 작업을 사용하십시오.

파라미터

- SourceSnapshotName 복사할 백업의 이름입니다.
- TargetSnapshotName 백업 복사본의 이름입니다.

• TargetBucket – 백업을 내보내기 위해 예약됩니다. 백업 복사본을 만들 때 이 파라미터를 사용하지 마십시오. 자세한 내용은 백업 내보내기 (p. 201) 단원을 참조하십시오.

다음 예제에서는 자동 백업 복사본을 만듭니다.

Example

https://elasticache.us-west-2.amazonaws.com/
 ?Action=CopySnapshot
 &SourceSnapshotName=automatic.my-redis-primary-2014-03-27-03-15
 &TargetSnapshotName=my-backup-copy
 &SignatureVersion=4
 &SignatureMethod=HmacSHA256
 &Timestamp=20141201T220302Z
 &Version=2014-12-01
 &X-Amz-Algorithm=AWS4-HMAC-SHA256
 &X-Amz-Date=20141201T220302Z
 &X-Amz-SignedHeaders=Host
 &X-Amz-Expires=20141201T220302Z
 &X-Amz-Expires=20141201T220302Z
 &X-Amz-Credential=<credential>
 &X-Amz-Signature=<signature>

자세한 내용은 Amazon ElastiCache API Reference의 CopySnapshot 항목을 참조하십시오.

백업 내보내기

Amazon ElastiCache에서는 Amazon Simple Storage Service(Amazon S3) 버킷에 ElastiCache 백업 내보 내기가 지원되므로 ElastiCache 외부에서 백업에 액세스할 수 있습니다. ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 백업을 내보낼 수 있습니다.

다른 AWS 리전에서 클러스터를 시작해야 할 경우 백업 내보내기가 유용할 수 있습니다. 한 AWS 리전에서 데이터를 내보내고 .rdb 파일을 새 AWS 리전으로 복사한 후 새 클러스터가 채워질 때까지 기다리는 대신 해당 .rdb 파일을 사용하여 새 클러스터를 시드할 수 있습니다. 새 클러스터 시드에 대한 자세한 내용은 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211) 단원을 참조하십시오. 오프라인 처리를 위해 .rdb 파일을 사용하기 위해 클러스터의 데이터를 내보낼 수도 있습니다.

Important

ElastiCache 백업과 백업을 복사할 Amazon S3 버킷이 같은 AWS 리전에 있어야 합니다. Amazon S3 버킷으로 복사된 백업이 암호화되긴 하지만 백업을 저장하려는 Amazon S3 버킷에 대한 액세스 권한을 다른 사람에게 부여하지 않는 것이 좋습니다.

Amazon S3 버킷으로 백업을 내보내려면 백업과 같은 AWS 리전에 Amazon S3 버킷에 있어야 합니다. 버킷에 ElastiCache 액세스 권한을 부여합니다. 처음 두 단계에서는 이 작업을 수행할 방법을 보여줍니다.

Warning

다음 시나리오에서는 사용자가 원하지 않을 수도 있는 방법으로 데이터를 노출합니다.

• 백업을 내보낸 Amazon S3 버킷에 대한 액세스 권한이 다른 사람에게 있는 경우.

백업에 대한 액세스를 제어하려면 데이터에 대한 액세스를 허용할 사람에게만 Amazon S3 버킷에 대한 액세스를 허용합니다. Amazon S3 버킷에 대한 액세스 관리에 대한 내용은 Amazon S3 개발자 안내서의 액세스 관리를 참조하십시오.

• 다른 사람이 CopySnapshot API 작업을 사용할 권한을 가진 경우

CopySnapshot API 작업을 사용할 권한을 가진 사용자나 그룹이 자체 Amazon S3 버킷을 생성 하고 백업을 이 버킷에 복사할 수 있습니다. 백업에 대한 액세스를 제어하려면 AWS Identity and Access Management 정책을 사용하여 CopySnapshot API를 사용할 수 있는 사람을 제어하십시

오. IAM을 사용하여 ElastiCache API 사용을 제어하는 데 대한 자세한 내용은 ElastiCache 사용 설명서의 Amazon ElastiCache의 Identity and Access Management(IAM) (p. 341) 단원을 참조 하십시오.

주제

- 2단계: Amazon S3 버킷 생성 (p. 202)
- 2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여 (p. 202)
- 3단계: ElastiCache 백업 내보내기 (p. 205)

2단계: Amazon S3 버킷 생성

다음 절차에서는 Amazon S3 콘솔을 사용하여 ElastiCache 백업을 내보내고 저장할 Amazon S3 버킷을 생성합니다.

Amazon S3 버킷을 생성하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
- 2. [Create Bucket]을 선택합니다.
- 3. [Create a Bucket Select a Bucket Name and Region]에서 다음을 수행합니다.
 - a. 버킷 이름에 Amazon S3 버킷 이름을 입력합니다.

Amazon S3 버킷의 이름은 DNS를 준수해야 합니다. 그렇지 않으면 ElastiCache가 백업 파일에 액세스할 수 없습니다. DNS 준수 규칙은 다음과 같습니다.

- 이름은 3자 이상, 63자 이하여야 합니다.
- 이름은 마침표(.)로 구분된 일련의 레이블(1개 이상)이어야 합니다. 여기서 각 레이블은 다음과 같 아야 합니다.
 - 소문자 또는 숫자로 시작합니다.
 - 소문자 또는 숫자로 끝납니다.
 - 소문자, 숫자 및 대시만 포함합니다.
- 이름에는 IP 주소 형식(예: 192.0.2.0)을 사용할 수 없습니다.
- b. 리전 목록에서 Amazon S3 버킷의 AWS 리전을 선택합니다. 이 AWS 리전은 내보낼 ElastiCache 백업과 동일한 AWS 리전이어야 합니다.
- c. Create를 선택합니다.

Amazon S3 버킷 생성에 대한 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서의 버킷 생성을 참조하십시오.

2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여

ElastiCache에서 스냅샷을 Amazon S3 버킷에 복사할 수 있게 하려면 버킷에 대한 액세스 권한을 부여합니다. 버킷이 기본 AWS 리전인지 아니면 옵트인 AWS 리전인지에 따라 ElastiCache에 다른 방식으로 Amazon S3 버킷에 대한 액세스 권한을 부여합니다.

2019년 3월 20일 이전에 도입된 AWS 리전은 기본적으로 활성 상태입니다. 이러한 AWS 리전에서는 즉시 작업을 시작할 수 있습니다. 아시아 태평양(홍콩) 및 중동(바레인)과 같이 2019년 3월 20일 이후에 도입된 리전은 기본적으로 비활성 상태입니다. AWS General Reference의 AWS 리전 관리에 설명된 대로 이러한 리전을 사용하려면 먼저 해당 리전을 활성화하거나 옵트인해야 합니다.

AWS 리전에 따라 접근 방식을 선택합니다.

- 기본 Region 의 경우 ElastiCache에 기본 리전의 S3 버킷에 대한 액세스 권한 부여 (p. 203)의 절차를 사용합니다.
- 옵트인 리전의 경우 ElastiCache에 옵트인 리전의 S3 버킷에 대한 액세스 권한 부여 (p. 204)의 절차를 사용합니다.

ElastiCache에 기본 리전의 S3 버킷에 대한 액세스 권한 부여

2019년 3월 20일 이전에 도입된 AWS 리전은 기본적으로 활성 상태입니다. 이러한 AWS 리전에서는 즉시 작업을 시작할 수 있습니다. 아시아 태평양(홍콩) 및 중동(바레인)과 같이 2019년 3월 20일 이후에 도입된 리전은 기본적으로 비활성 상태입니다. AWS General Reference의 AWS 리전 관리에 설명된 대로 이러한 리전을 사용하려면 먼저 해당 리전을 활성화하거나 옵트인해야 합니다.

기본적으로 활성화된 AWS 리전의 Amazon S3 버킷에 대한 적절한 권한을 생성하려면 다음 단계를 수행합니다.

Warning

Amazon S3 버킷으로 복사된 백업이 암호화되더라도 Amazon S3 버킷에 대한 액세스 권한이 있는 사람이 데이터에 액세스할 수 있습니다. 따라서 이 Amazon S3 버킷에 대한 무단 액세스를 방지하 도록 IAM 정책을 설정하는 것이 좋습니다. 자세한 내용은 Amazon S3 개발자 안내서의 액세스 관리를 참조하십시오.

ElastiCache에 기본 AWS 리전의 S3 버킷에 대한 액세스 권한을 부여하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
- 2. 백업을 복사할 Amazon S3 버킷 이름을 선택합니다. 2단계: Amazon S3 버킷 생성 (p. 202)에서 생성 한 S3 버킷이어야 합니다.
- 3. 버킷의 AWS 리전은 ElastiCache 백업의 AWS 리전과 동일해야 합니다. 그렇지 않을 경우 2단계: Amazon S3 버킷 생성 (p. 202)으로 돌아가 백업할 클러스터와 동일한 AWS 리전에 새 버킷을 만듭니다.
- 4. 권한 탭을 선택하고 액세스 제어 목록을 선택한 다음 다른 AWS 계정의 액세스에서 계정 추가를 선택합니다
- 5. 상자에서 다음 목록에 표시된 대로 AWS 리전의 정식 ID를 추가합니다.
 - 중국(베이징) 및 중국(닝샤) 리전:

b14d6a125bdf69854ed8ef2e71d8a20b7c490f252229b806e514966e490b8d83

• AWS GovCloud (US-West) 리전:

40fa568277ad703bd160f66ae4f83fc9dfdfd06c2f1b5060ca22442ac3ef8be6

Important

AWS GovCloud (US)의 S3 버킷에 백업을 내보내야 합니다.

• 다른 모든 AWS 리전은 기본적으로 활성화되어 있습니다.

540804c33a284a299d2547575ce1010f2312ef3da9b3a053c8bc45bf233e4353

- 6. 다음 옵션에 대해 예를 선택하여 버킷에 대한 권한을 설정합니다.
 - · List objects
 - · Write objects

- · Read bucket permissions
- 7. Save를 선택합니다.

이제 ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Amazon S3 버킷에 ElastiCache 백업을 내보낼 수 있습니다.

ElastiCache에 옵트인 리전의 S3 버킷에 대한 액세스 권한 부여

2019년 3월 20일 이전에 도입된 AWS 리전은 기본적으로 활성 상태입니다. 이러한 AWS 리전에서는 즉시 작업을 시작할 수 있습니다. 아시아 태평양(홍콩) 및 중동(바레인)과 같이 2019년 3월 20일 이후에 도입된 리전은 기본적으로 비활성 상태입니다. AWS General Reference의 AWS 리전 관리에 설명된 대로 이러한 리전을 사용하려면 먼저 해당 리전을 활성화하거나 옵트인해야 합니다.

옵트인 AWS 리전의 Amazon S3 버킷에 대한 적절한 권한을 생성하려면 다음 단계를 수행합니다.

ElastiCache에 옵트인 AWS 리전의 S3 버킷에 대한 액세스 권한을 부여하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
- 2. 백업을 복사할 Amazon S3 버킷 이름을 선택합니다. 2단계: Amazon S3 버킷 생성 (p. 202)에서 생성 한 S3 버킷이어야 합니다.
- 3. 권한 탭을 선택한 후 권한에서 버킷 정책을 선택합니다.
- 4. 정책을 업데이트하여 ElastiCache에 작업을 수행하는 데 필요한 권한을 부여합니다.
 - ["Service" : "region-full-name.elasticache-snapshot.amazonaws.com"]을 Principal에 추가합니다.
 - Amazon S3 버킷으로 스냅샷을 내보내는 데 필요한 다음 권한을 추가합니다.
 - "s3:PutObject"
 - "s3:GetObject"
 - "s3:ListBucket"
 - "s3:GetBucketAcl"
 - "s3:ListMultipartUploadParts"
 - "s3:ListBucketMultipartUploads"

다음은 업데이트된 정책의 예입니다.

```
"Version": "2012-10-17",
"Id": "Policy15397346",
"Statement": [
        "Sid": "Stmt15399483",
        "Effect": "Allow",
        "Principal": {
            "Service": "aws-opt-in-region.elasticache-snapshot.amazonaws.com"
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:ListBucket"
            "s3:GetBucketAcl",
            "s3:ListMultipartUploadParts",
            "s3:ListBucketMultipartUploads"
        "Resource": [
            "arn:aws:s3:::example-bucket",
```

3단계: ElastiCache 백업 내보내기

이제 S3 버킷을 생성하고 ElastiCache에 액세스할 수 있는 권한을 부여했습니다. 그런 다음 ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 스냅샷을 내보낼 수 있습니다. 다음 예제에서는 아래의 추가 S3 관련 IAM 권한이 있다고 가정합니다.

ElastiCache 백업 (콘솔) 내보내기

다음 프로세스에서는 ElastiCache 외부에서 액세스할 수 있도록 ElastiCache 콘솔을 사용하여 Amazon S3 버킷에 백업을 내보냅니다. Amazon S3 버킷은 ElastiCache 백업과 동일한 AWS 리전에 있어야 합니다.

ElastiCache 백업을 Amazon S3 버킷에 내보내려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 백업 목록을 표시하려면 왼쪽 탐색 창에서 [Backups]를 선택합니다.
- 3. 백업 목록에서 내보낼 백업의 이름 왼쪽에 있는 상자를 선택합니다.
- 4. [Copy]를 선택합니다.
- 5. Create a Copy of the Backup(백업 복사본을 생성)에서 다음 작업을 수행하십시오.
 - a. [New backup name] 상자에 새 백업의 이름을 입력합니다.

이름은 1~1,000자여야 하며 UTF-8 인코딩일 수 있습니다.

ElastiCache에서는 여기에 입력하는 값에 인스턴스 식별자 및 .rdb를 추가합니다. 예를 들어 my-exported-backup을 입력하면 ElastiCache에서 my-exported-backup-0001.rdb를 생성합니다.

b. Target S3 Location(대상 S3 위치) 목록에서 백업을 복사할 Amazon S3버킷(2단계: Amazon S3 버 킷 생성 (p. 202)에서 생성한 버킷)을 선택합니다.

Target S3 Location(대상 S3 위치)는 내보내기 프로세스가 성공할 수 있는 다음 권한이 있으며 백업의 AWS 리전에 있는 Amazon S3 버킷이어야 합니다.

• 객체 액세스 – 읽기 및 쓰기.

• 권한 액세스 – 읽기.

자세한 내용은 2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여 (p. 202) 단원을 참조하십시오.

c. [Copy]를 선택합니다.

Note

ElastiCache에서 백업을 내보내는 데 필요한 권한이 S3 버킷에 없으면 다음 오류 메시지 중 하나를 받습니다. 2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여 (p. 202)로 돌아가 지정한 권한을 추가하고 백업 내보내기를 다시 시도하십시오.

- S3 버킷에 대한 READ 권한이 ElastiCache에 부여되지 않았습니다.
 - 해결 방법: 버킷에 대한 읽기 권한을 추가합니다.
- S3 버킷에 대한 WRITE 권한이 ElastiCache에 부여되지 않았습니다.
 - 해결 방법: 버킷에 대한 쓰기 권한을 추가합니다.
- S3 버킷에 대한 READ_ACP 권한이 ElastiCache에 부여되지 않았습니다.
 - 해결 방법: 버킷에 대한 권한 액세스로 [Read]를 추가합니다.

다른 AWS 리전에 백업을 복사하려면 Amazon S3을 사용하여 백업을 복사합니다. 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서의 객체 복사를 참조하십시오.

ElastiCache 백업 (AWS CLI) 내보내기

다음 파라미터와 함께 copy-snapshot CLI 작업을 사용하여 Amazon S3 버킷에 백업을 내보냅니다.

파라미터

- --source-snapshot-name 복사할 백업의 이름입니다.
- --target-snapshot-name 백업 복사본의 이름입니다.
 - 이름은 1~1,000자여야 하며 UTF-8 인코딩일 수 있습니다.

ElastiCache에서는 여기에 입력하는 값에 인스턴스 식별자 및 .rdb를 추가합니다. 예를 들어 my-exported-backup을 입력하면 ElastiCache에서 my-exported-backup-0001.rdb를 생성합니다.

- --target-bucket 백업을 내보낼 Amazon S3 버킷의 이름입니다. 지정한 버킷에 백업 복사본이 생성 됩니다.
 - --target-bucket은 내보내기 프로세스가 성공할 수 있는 다음 권한이 있으며 백업의 AWS 리전에 있는 Amazon S3 버킷이어야 합니다.
 - 객체 액세스 읽기 및 쓰기.
 - 권한 액세스 읽기.

자세한 내용은 2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여 (p. 202) 단원을 참조하십시오.

다음 작업은 my-s3-bucket에 백업을 복사합니다.

Linux, macOS, or Unix의 경우:

aws elasticache copy-snapshot \
 --source-snapshot-name automatic.my-redis-primary-2016-06-27-03-15 \

--target-snapshot-name my-exported-backup \
--target-bucket my-s3-bucket

Windows의 경우:

aws elasticache copy-snapshot ^
--source-snapshot-name automatic.my-redis-primary-2016-06-27-03-15 ^
--target-snapshot-name my-exported-backup ^
--target-bucket my-s3-bucket

Note

ElastiCache에서 백업을 내보내는 데 필요한 권한이 S3 버킷에 없으면 다음 오류 메시지 중 하나를 받습니다. 2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여 (p. 202)로 돌아가 지정한 권한을 추가하고 백업 내보내기를 다시 시도하십시오.

• S3 버킷에 대한 READ 권한이 ElastiCache에 부여되지 않았습니다.

해결 방법: 버킷에 대한 읽기 권한을 추가합니다.

• S3 버킷에 대한 WRITE 권한이 ElastiCache에 부여되지 않았습니다.

해결 방법: 버킷에 대한 쓰기 권한을 추가합니다.

• S3 버킷에 대한 READ_ACP 권한이 ElastiCache에 부여되지 않았습니다.

해결 방법: 버킷에 대한 권한 액세스로 [Read]를 추가합니다.

자세한 내용은 AWS CLI Command Reference의 copy-snapshot 항목을 참조하십시오.

다른 AWS 리전에 백업을 복사하려면 Amazon S3 복사본을 사용합니다. 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서의 객체 복사를 참조하십시오.

ElastiCache 백업 (ElastiCache API) 내보내기

이러한 파라미터와 함께 CopySnapshot API 작업을 사용하여 Amazon S3 버킷에 백업을 내보냅니다.

파라미터

- SourceSnapshotName 복사할 백업의 이름입니다.
- TargetSnapshotName 백업 복사본의 이름입니다.

이름은 1~1.000자여야 하며 UTF-8 인코딩일 수 있습니다.

ElastiCache에서는 여기에 입력하는 값에 인스턴스 식별자 및 .rdb를 추가합니다. 예를 들어 my-exported-backup을 입력하면 my-exported-backup-0001.rdb가 생깁니다.

• TargetBucket – 백업을 내보낼 Amazon S3 버킷의 이름입니다. 지정한 버킷에 백업 복사본이 생성됩니다.

TargetBucket은 내보내기 프로세스가 성공할 수 있는 다음 권한이 있으며 백업의 AWS 리전에 있는 Amazon S3 버킷이어야 합니다.

- 객체 액세스 읽기 및 쓰기.
- 권한 액세스 읽기.

자세한 내용은 2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여 (p. 202) 단원을 참조하십시오.

다음 예제에서는 Amazon S3 버킷 my-s3-bucket에 자동 백업 복사본을 만듭니다.

Example

https://elasticache.us-west-2.amazonaws.com/
 ?Action=CopySnapshot
 &SourceSnapshotName=automatic.my-redis-primary-2016-06-27-03-15
 &TargetBucket=my-s3-bucket
 &TargetSnapshotName=my-backup-copy
 &SignatureVersion=4
 &SignatureMethod=HmacSHA256
 &Timestamp=20141201T220302Z
 &Version=2016-01-01
 &X-Amz-Algorithm=AWS4-HMAC-SHA256
 &X-Amz-Date=20141201T220302Z
 &X-Amz-SignedHeaders=Host
 &X-Amz-Expires=20141201T220302Z
 &X-Amz-Credential=<credential>
 &X-Amz-Signature=<signature>

Note

ElastiCache에서 백업을 내보내는 데 필요한 권한이 S3 버킷에 없으면 다음 오류 메시지 중 하나를 받습니다. 2단계: Amazon S3 버킷에 대한 ElastiCache 액세스 권한 부여 (p. 202)로 돌아가 지정한 권한을 추가하고 백업 내보내기를 다시 시도하십시오.

- S3 버킷에 대한 READ 권한이 ElastiCache에 부여되지 않았습니다.
 - 해결 방법: 버킷에 대한 읽기 권한을 추가합니다.
- S3 버킷에 대한 WRITE 권한이 ElastiCache에 부여되지 않았습니다.
 - 해결 방법: 버킷에 대한 쓰기 권한을 추가합니다.
- S3 버킷에 대한 READ_ACP 권한이 ElastiCache에 부여되지 않았습니다.
 - 해결 방법: 버킷에 대한 권한 액세스로 [Read]를 추가합니다.

자세한 내용은 Amazon ElastiCache API Reference의 CopySnapshot 항목을 참조하십시오.

다른 AWS 리전에 백업을 복사하려면 Amazon S3 복사본을 사용하여 내보낸 백업을 다른 AWS 리전의 Amazon S3 버킷에 복사합니다. 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서의 객체 복사를 참조하십시오.

선택적으로 클러스터 크기를 조정하여 백업에서 복원

언제든지 Redis .rdb 백업 파일에서 새 클러스터로 데이터를 복원할 수 있습니다.

Redis용 Amazon ElastiCache 복원 프로세스는 다음을 지원합니다.

- Redis(클러스터 모드 비활성화됨) 클러스터에서 Redis 버전 3.2.4를 실행하는 Redis(클러스터 모드 활성화됨) 클러스터로 업그레이드.
- 자체 관리형 Redis 클러스터로 생성한 하나 이상의 .rdb 백업 파일에서 Redis(클러스터 모드 활성화됨) 클 러스터용 단일 ElastiCache로 마이그레이션입니다.

.rdb 파일을 S3에 배치해야 복원할 수 있습니다.

- 백업 파일 생성에 사용한 클러스터의 샤드 수와 다른 새 클러스터의 샤드(API/CLI: 노드 그룹) 수 지정
- —더 크거나 작은 새 클러스터의 다른 노드 유형 지정. 더 작은 노드 유형으로 조정하는 경우 새로운 노드 유형에 충분한 메모리가 있어 데이터와 Redis 오버헤드를 수용할 수 있어야 합니다. 자세한 내용은 노드 크기 선택 (p. 67) 단원을 참조하십시오.
- 백업 파일 생성에 사용한 클러스터에서와 다른 새 Redis(클러스터 모드 활성화됨) 클러스터의 슬롯 구성입니다.

Important

- Redis(클러스터 모드 활성화됨)클러스터Redis(클러스터 모드 비활성화됨)를 사용해 생성된 백업을 클러스터로 복원할 수 없습니다.
- Redis(클러스터 모드 활성화됨) 클러스터는 여러 데이터베이스를 지원하지 않습니다. 따라서 Redis(클러스터 모드 활성화됨)로 복원하는 경우 .rdb 파일이 데이터베이스를 두 개 이상 참조하면 복원에 실패합니다.

백업에서 클러스터를 복원할 때 변경 여부를 선택할 수 있습니다. ElastiCache 콘솔을 사용하여 복원할 때 클러스터 복원 대화 상자에서 이러한 선택을 할 수 있습니다. AWS CLI 또는 ElastiCache API를 사용하여 복원할 때 파라미터 값을 설정하여 이러한 선택을 할 수 있습니다.

복원 작업 중에 ElastiCache가 새 클러스터를 만든 후 백업 파일의 데이터로 클러스터를 채웁니다. 이 프로세스가 완료되면 Redis 클러스터가 워밍업되어 요청을 수락할 준비가 됩니다.

Important

계속하기 전에 복원할 클러스터의 백업을 생성해야 합니다. 자세한 내용은 수동 백업 만들기 (p. 190) 단원을 참조하십시오.

외부에서 생성된 백업에서 복원하려면 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211) 단원을 참조하십시오.

다음 절차에서는 ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 백업을 새 클러스터로 복원하는 방법을 보여줍니다.

주제

- 백업에서 복원 (콘솔) (p. 209)
- 백업에서 복원 (AWS CLI) (p. 210)
- 백업에서 복원 (ElastiCache API) (p. 211)

백업에서 복원 (콘솔)

두 가지 방법으로 Redis 백업을 복원할 수 있습니다. 싱글 노드 Redis(클러스터 모드 비활성화됨) 클러스터로 복원할 수 있습니다. 또는 읽기 전용 복제본(복제 그룹)으로 Redis(클러스터 모드 비활성화됨) 또는 Redis(클러스터 모드 활성화됨) Redis 클러스트로 복원할 수 있습니다.

새 클러스터로 백업을 복원하는 방법 (console)

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 탐색 창에서 [Backups]를 선택합니다.
- 3. 백업 목록에서 복원할 백업의 이름 왼쪽에 있는 상자를 선택합니다.
- 4. [Restore]를 선택합니다.
- 5. [Restore Cluster] 대화 상자를 완료합니다. 모든 "필수" 필드와 기본값을 변경할 다른 필드를 완료해야 합니다.

Redis(클러스터 모드 비활성화됨)

- 1. 클러스터 ID 필수, 새 클러스터의 이름입니다.
- 2. Engine version compatibility(엔진 버전 호환성) 실행할 Redis 엔진 버전의 ElastiCache.
- 3. Cluster mode enabled(scale out)(클러스터 모드 활성화(스케일 아웃)) Redis(클러스터 모드 비활성화됨) 클러스터를 Redis(클러스터 모드 활성화됨)로 변환하려면 선택하십시오. 엔진 버전이 3.2.4.가됩니다.

[Cluster mode enabled (scale out)]를 선택하는 경우

- a. 새 클러스터(API/CLI: 노드 그룹)에 포함할 샤드 수를 선택합니다.
- b. 각 샤드에 포함할 읽기 전용 복제본 수를 선택합니다.
- c. 원하는 대로 슬롯에 키를 배포합니다.
- 4. 노드 유형 새 클러스터에 사용할 노드 유형을 지정합니다.
- 5. 가용 영역 클러스터의 가용 영역 선택 방법을 지정합니다.
- 6. 포트 이 클러스터에 다른 포트를 사용하려는 경우에만 변경합니다.
- 7. VPC 선택 이 클러스터를 생성할 VPC를 선택합니다.
- 8. 파라미터 그룹 선택한 노드 유형의 Redis 오버헤드를 수용할 만큼 충분한 메모리가 있는 파라미터 그룹을 선택합니다.

Redis(클러스터 모드 활성화됨)

- 1. 클러스터 ID 필수. 새 클러스터의 이름입니다.
- 2. Cluster mode enabled(scale out)(클러스터 모드 활성화(스케일 아웃)) Redis(클러스터 모드 활성화됨) 클러스터에서 선택합니다. Redis(클러스터 모드 비활성화됨) 클러스터에 대해서는 선택을 취소합니다.
- 3. 노드 유형 새 클러스터에 사용할 노드 유형을 지정합니다.
- 4. 샤드 수 새 클러스터(API/CLI: 노드 그룹)에 포함할 샤드 수를 선택합니다.
- 5. Replicas per Shard(샤드 당 복제본) 각 샤드에 포함할 읽기 전용 복제본 노드 수를 선택합니다.
- 6. Slots and keyspaces(슬롯 및 키스페이스) 샤드에 키를 배포할 방법을 선택합니다. 키 배포를 지정하도록 선택할 경우 각 샤드의 키 범위를 지정하는 표를 완료합니다.
- 7. 가용 영역 클러스터의 가용 영역 선택 방법을 지정합니다.
- 8. 포트 이 클러스터에 다른 포트를 사용하려는 경우에만 변경합니다.
- 9. VPC 선택 이 클러스터를 생성할 VPC를 선택합니다.
- 10파라미터 그룹 선택한 노드 유형의 Redis 오버헤드를 수용할 만큼 충분한 메모리가 있는 파라미터 그룹을 선택합니다.
- 6. 원하는 대로 설정되었으면 [Launch Cluster]를 선택합니다.

백업에서 복원 (AWS CLI)

두 가지 방법으로 Redis(클러스터 모드 비활성화됨) 백업을 복원할 수 있습니다. AWS CLI 작업 create-cache-cluster를 사용하여 싱글 노드 Redis(클러스터 모드 비활성화됨) 클러스터로 복원할 수 있습니다. 또는 읽기 전용 복제본(복제 그룹)으로 Redis 클러스터로 복원할 수 있습니다. 후자의 방법을 수행하려면 AWS CLI 작업 create-replication-group으로 Redis(클러스터 모드 비활성화됨) 또는 Redis(클러스터 모드 활성화됨) 중 하나를 사용할 수 있습니다. 이 경우 Redis .rdb 파일로 복원을 시드합니다.

create-cache-cluster 또는 create-replication-group 작업을 사용할 때는 --snapshot-name 또는 --snapshot-arns 파라미터를 포함하여 백업의 데이터로 새로운 클러스터나 복제 그룹을 시드해야 합니다.

자세한 내용은 다음을 참조하십시오.

- 클러스터 생성 (AWS CLI) (p. 76)(출처: ElastiCache 사용 설명서).
- AWS CLI Command Reference of create-cache-cluster.
- Redis 복제 그룹을 처음부터 새로 생성 (p. 148)(출처: ElastiCache 사용 설명서).

• AWS CLI Command Reference create-replication-group.

백업에서 복원 (ElastiCache API)

ElastiCache API 작업 CreateCacheCluster을 사용하는 단일 노드 Redis(클러스터 모드 비활성화됨) 클러스터 또는 읽기 전용 복제본이 있는 Redis 클러스터(복제 그룹)—, 즉 ElastiCache API 작업 CreateReplicationGrooup을 사용하고 Redis .rdb 파일로 이를 시드하는 Redis(클러스터 모드 비활성화됨) 또는 Redis(클러스터 모드 활성화됨)로 Redis 백업을 복원할 수 있습니다.

CreateCacheCluster 또는 CreateReplicationGroup 작업을 사용할 때는 SnapshotName 또는 SnapshotArns 파라미터를 포함하여 백업의 데이터로 새로운 클러스터나 복제 그룹을 시드해야 합니다.

자세한 내용은 다음을 참조하십시오.

- 클러스터 생성 (ElastiCache API) (p. 77)(출처: ElastiCache 사용 설명서).
- ElastiCache API 참조의 CreateCacheCluster.
- Redis 복제 그룹을 처음부터 새로 생성 (p. 148)(출처: ElastiCache 사용 설명서).
- ElastiCache API 참조의 CreateReplicationGroup.

외부에서 생성된 백업으로 새 클러스터 시드

새 Redis 클러스터를 만들 때 Redis .rdb 백업 파일의 데이터로 이 클러스터를 시드할 수 있습니다. 현재 ElastiCache 외부에서 Redis 인스턴스를 관리하면서 Redis 클러스터용 새 ElastiCache에 기존의 Redis 데이터를 채우려는 경우 클러스터 시드가 유용합니다.

Amazon ElastiCache에서 생성한 Redis 백업 파일에서 새 Redis 클러스터를 시드하려면 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208)단원을 참조하십시오.

Redis .rdb 파일을 사용하여 새 Redis 클러스터를 시드할 때 다음을 수행할 수 있습니다.

- 분할되지 않은 클러스터에서 Redis 버전 3.2.4를 실행하는 Redis(클러스터 모드 활성화됨) 클러스터로 업 그레이드합니다.
- 새 클러스터에서 여러 샤드(API 및 CLI에서 노드 그룹이라고 함)를 지정합니다. 이 숫자는 백업 파일을 생성하는 데 사용된 클러스터의 샤드 수와 다를 수 있습니다.
- 백업을 만든 클러스터에 사용된 것보다 —크거나 작은 새 클러스터의 다른 노드 유형을 지정합니다. 더 작은 노드 유형으로 조정하는 경우 새로운 노드 유형에 충분한 메모리가 있어 데이터와 Redis 오버헤드를 수용할 수 있어야 합니다. 자세한 내용은 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404) 단원을 참조하십시오.
- 백업 파일 생성에 사용한 클러스터에서와 다른 새 Redis(클러스터 모드 활성화됨) 클러스터의 슬롯에 키를 배포합니다.

Note

Redis(클러스터 모드 활성화됨) 클러스터에서 생성된 .rdb 파일에서 Redis(클러스터 모드 비활성화됨) 클러스터를 시드할 수 없습니다.

Important

• Redis 백업 데이터가 노드의 리소스를 초과하지 않아야 합니다. 예를 들어 Redis 데이터가 5GB 인 .rdb 파일을 메모리가 2.9GB인 cache.m3.medium 노드에 업로드할 수 없습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 백업으로 클러스터 시드

백업이 너무 크면 결과 클러스터의 상태는 restore-failed가 됩니다. 이 경우 클러스터를 삭제하고 다시 시작해야 합니다.

노드 유형 및 사양의 전체 목록은 Redis 노드 유형별 파라미터 (p. 295) 및 Amazon ElastiCache 제품 기능 및 세부 정보를 참조하십시오.

• Amazon S3 서버 측 암호화(SSE-S3)만으로 Redis .rdb 파일을 암호화할 수 있습니다. 자세한 내용은 서버 측 암호화를 사용하여 데이터 보호 단원을 참조하십시오.

아래에서는 Redis용 ElastiCache 외부에서 Redis용 ElastiCache로 Redis 클러스터를 마이그레이션하는 과 정을 안내하는 항목을 찾을 수 있습니다.

Redis용 ElastiCache로 마이그레이션

- 1단계: Redis 백업 생성 (p. 212)
- 2단계: Amazon S3 버킷 및 폴더 생성 (p. 212)
- 3단계: Amazon S3에 백업 업로드 (p. 213)
- 4단계: ElastiCache에 .rdb 파일에 대한 읽기 액세스 부여 (p. 214)
- 5단계: .rdb 파일 데이터로 ElastiCache 클러스터 시드 (p. 216)

1단계: Redis 백업 생성

Redis 인스턴스용 ElastiCache를 시드할 Redis 백업을 생성하려면

- 1. 기존의 Redis 인스턴스에 연결합니다.
- 2. Redis BGSAVE 또는 SAVE 작업을 실행하여 백업을 생성합니다. .rdb 파일의 위치를 메모합니다.

BGSAVE는 비동기식이며 처리하는 동안 다른 클라이언트를 차단하지 않습니다. 자세한 내용은 Redis 웹사이트의 BGSAVE를 참조하십시오.

SAVE는 동기식이며 마칠 때까지 다른 프로세스를 차단합니다. 자세한 내용은 Redis 웹 사이트의 SAVE를 참조하십시오.

백업 생성에 대한 추가 정보는 Redis 웹 사이트의 Redis 지속성을 참조하십시오.

2단계: Amazon S3 버킷 및 폴더 생성

백업 파일을 만들었으면 Amazon S3 버킷에 있는 폴더에 업로드해야 합니다. 그러려면 먼저 Amazon S3 버킷과 버킷 내의 폴더가 있어야 합니다. 적절한 권한을 가진 Amazon S3 버킷과 폴더가 이미 있으면 3단계: Amazon S3에 백업 업로드 (p. 213)로 건너뛸 수 있습니다.

Amazon S3 버킷을 생성하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 연니다.
- 2. Amazon Simple Storage Service 콘솔 사용 설명서에 나온 버킷 생성의 Amazon S3 버킷 생성에 대한 지침을 따르십시오.

Amazon S3 버킷의 이름은 DNS를 준수해야 합니다. 그렇지 않으면 ElastiCache가 백업 파일에 액세스할 수 없습니다. DNS 준수 규칙은 다음과 같습니다.

- 이름은 3자 이상, 63자 이하여야 합니다.
- 이름은 마침표(.)로 구분된 일련의 레이블(1개 이상)이어야 합니다. 여기서 각 레이블은 다음과 같아야 합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 백업으로 클러스터 시드

- 소문자 또는 숫자로 시작합니다.
- 소문자 또는 숫자로 끝납니다.
- 소문자, 숫자 및 대시만 포함합니다.
- 이름에는 IP 주소 형식(예: 192.0.2.0)을 사용할 수 없습니다.

Redis 클러스터용 새 ElastiCache과 동일한 AWS 리전에 Amazon S3 버킷을 생성하는 것이 좋습니다. 이 방법은 ElastiCache가 Amazon S3에서 .rdb 파일을 읽을 때 최고의 데이터 전송 속도를 경험할 수 있 도록 합니다.

Note

데이터를 최대한 안전하게 유지하려면 Amazon S3 S3 버킷에 대한 권한을 최대한 제한적으로 설정합니다. 또한 새 Redis 클러스터를 시드하는 데 버킷과 버킷의 콘텐츠를 사용하기 위해 권 한이 계속 필요합니다.

Amazon S3 버킷에 폴더를 추가하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
- 2. .rdb 파일을 업로드할 버킷 이름을 선택합니다.
- 3. [Create folder]를 선택합니다.
- 4. 새 폴더의 이름을 입력합니다.
- 5. 저장을 선택합니다.

버킷 이름과 폴더 이름을 모두 메모합니다.

3단계: Amazon S3에 백업 업로드

이제 1단계: Redis 백업 생성 (p. 212)에서 생성한 .rdb 파일을 업로드합니다. 2단계: Amazon S3 버킷 및 폴더 생성 (p. 212)에서 생성한 Amazon S3 버킷과 폴더로 업로드합니다. 이 작업에 대한 자세한 내용은 버킷에 객체 추가를 참조하십시오. 2단계와 3단계 사이에 생성된 폴더 이름을 선택합니다.

.rdb 파일을 Amazon S3 폴더에 업로드하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
- 2. 2단계에서 만든 Amazon S3 버킷 이름을 선택합니다.
- 3. 2단계에서 만든 폴더 이름을 선택합니다.
- 4. Upload를 선택합니다.
- 5. [Add Files]를 선택합니다.
- 6. 업로드할 파일을 찾아 선택합니다. 파일을 여러 개 선택하려면 Ctrl 키를 누른 상태로 각 파일 이름을 선택합니다.
- 7. [Open]을 선택합니다.
- 8. [Upload] 대화 상자에서 정확한 파일 이름이 표시되는지 확인하고 [Upload]를 선택합니다.

.rdb 파일에 대한 경로를 기록합니다. 예를 들어 버킷 이름이 myBucket이고 경로가 myFolder/redis.rdb이면 myBucket/myFolder/redis.rdb를 입력합니다. 이 백업의 데이터로 새 클러스터를 시드하려면 이 경로가 필요합니다.

추가 내용은 Amazon Simple Storage Service 개발자 가이드의 버킷 규제 및 제한을 참조하십시오.

4단계: ElastiCache에 .rdb 파일에 대한 읽기 액세스 부여

이제 ElastiCache에 .rdb 백업 파일에 대한 읽기 액세스 권한을 부여합니다. 버킷이 기본 AWS 리전인지 아니면 옵트인 AWS 리전인지에 따라 ElastiCache에 다른 방식으로 백업 파일에 대한 액세스 권한을 부여합니다.

2019년 3월 20일 이전에 도입된 AWS 리전은 기본적으로 활성 상태입니다. 이러한 AWS 리전에서는 즉시 작업을 시작할 수 있습니다. 아시아 태평양(홍콩) 및 중동(바레인)과 같이 2019년 3월 20일 이후에 도입된 리전은 기본적으로 비활성 상태입니다. AWS General Reference의 AWS 리전 관리에 설명된 대로 이러한 리전을 사용하려면 먼저 해당 리전을 활성화하거나 옵트인해야 합니다.

AWS 리전에 따라 접근 방식을 선택합니다.

- 기본 리전의 경우 ElastiCache에 기본 리전의 .rdb 파일에 대한 읽기 액세스 권한 부여 (p. 214)의 절차를 사용합니다.
- 옵트인 리전의 경우 ElastiCache에 옵트인 리전의 .rdb 파일에 대한 읽기 액세스 권한 부여 (p. 215)의 절 차를 사용합니다.

ElastiCache에 기본 리전의 .rdb 파일에 대한 읽기 액세스 권한 부여

2019년 3월 20일 이전에 도입된 AWS 리전은 기본적으로 활성 상태입니다. 이러한 AWS 리전에서는 즉시 작업을 시작할 수 있습니다. 아시아 태평양(홍콩) 및 중동(바레인)과 같이 2019년 3월 20일 이후에 도입된 리전은 기본적으로 비활성 상태입니다. AWS General Reference의 AWS 리전 관리에 설명된 대로 이러한 리전을 사용하려면 먼저 해당 리전을 활성화하거나 옵트인해야 합니다.

ElastiCache에 기본적으로 활성화된 AWS 리전의 백업 파일에 대한 읽기 액세스 권한을 부여하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
- 2. .rdb 파일이 포함된 S3 버킷 이름을 선택합니다.
- 3. .rdb 파일이 포함된 폴더 이름을 선택합니다.
- 4. .rdb 백업 파일 이름을 선택합니다. 선택한 파일 이름은 페이지 맨 위의 탭 위에 나타납니다.



- 5. Permissions를 선택합니다.
- 6. aws-scs-s3-readonly 또는 다음 목록에 있는 정식 ID 중 하나가 사용자로 나열되지 않으면 다음을 수행합니다.
 - a. 다른 AWS 계정에 대한 액세스에서 계정 추가를 선택합니다.
 - b. 상자에서 다음 목록에 표시된 대로 AWS 리전의 정식 ID를 추가합니다.
 - 중국(베이징) 및 중국(닝샤) 리전:

b14d6a125bdf69854ed8ef2e71d8a20b7c490f252229b806e514966e490b8d83

• AWS GovCloud (US-West) 리전:

40fa568277ad703bd160f66ae4f83fc9dfdfd06c2f1b5060ca22442ac3ef8be6

Important

백업이 AWS GovCloud (US)의 S3 버킷에 있어야 AWS GovCloud (US)의 Redis 클러스터로 백업을 다운로드할 수 있습니다.

• 다른 모든 AWS 리전은 기본적으로 활성화되어 있습니다.

540804c33a284a299d2547575ce1010f2312ef3da9b3a053c8bc45bf233e4353

- c. 다음에 대해 예를 선택하여 버킷에 대한 권한을 설정합니다.
 - · Read object
 - · Read object permissions
- d. 저장을 선택합니다.
- 7. 개요를 선택한 다음 다운로드를 선택합니다.

ElastiCache에 옵트인 리전의 .rdb 파일에 대한 읽기 액세스 권한 부여

2019년 3월 20일 이전에 도입된 AWS 리전은 기본적으로 활성 상태입니다. 이러한 AWS 리전에서는 즉시 작업을 시작할 수 있습니다. 아시아 태평양(홍콩) 및 중동(바레인)과 같이 2019년 3월 20일 이후에 도입된 리전은 기본적으로 비활성 상태입니다. AWS General Reference의 AWS 리전 관리에 설명된 대로 이러한 리전을 사용하려면 먼저 해당 리전을 활성화하거나 옵트인해야 합니다.

ElastiCache에 AWS 리전의 백업 파일에 대한 읽기 액세스 권한을 부여하려면

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
- 2. .rdb 파일이 포함된 S3 버킷 이름을 선택합니다.
- 3. .rdb 파일이 포함된 폴더 이름을 선택합니다.
- 4. .rdb 백업 파일 이름을 선택합니다. 선택한 파일 이름은 페이지 맨 위의 탭 위에 나타납니다.



- 5. [Permissions] 탭을 선택합니다.
- 6. 권한에서 버킷 정책을 선택합니다.
- 7. 정책을 업데이트하여 ElastiCache에 작업을 수행하는 데 필요한 권한을 부여합니다.
 - ["Service" : "region-full-name.elasticache-snapshot.amazonaws.com"]을 Principal에 추가합니다.
 - Amazon S3 버킷으로 스냅샷을 내보내는 데 필요한 다음 권한을 추가합니다.
 - "s3:GetObject"
 - "s3:ListBucket"
 - "s3:GetBucketAcl"

다음은 업데이트된 정책의 예입니다.

```
{
    "Version": "2012-10-17",
    "Id": "Policy15397346",
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 백업으로 클러스터 시드

```
"Statement": [
            "Sid": "Stmt15399483",
            "Effect": "Allow",
            "Principal": {
                "Service": "ap-east-1.elasticache-snapshot.amazonaws.com"
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
                "s3:GetBucketAcl"
            "Resource": [
                "arn:aws:s3:::example-bucket",
                "arn:aws:s3:::example-bucket/backup1.rdb",
                "arn:aws:s3:::example-bucket/backup2.rdb"
            ]
        }
    1
}
```

5단계: .rdb 파일 데이터로 ElastiCache 클러스터 시드

이제 ElastiCache 클러스터를 생성하고 .rdb 파일의 데이터로 클러스터를 시드할 수 있습니다. 클러스터를 생성하려면 클러스터 생성 (p. 65) 또는 Redis 복제 그룹을 처음부터 새로 생성 (p. 148)의 지시를 따릅니다. Redis를 클러스터 엔진으로 선택해야 합니다.

Amazon S3에 업로드한 Redis 백업의 위치를 ElastiCache에 알리는 데 사용하는 방법은 클러스터 생성에 사용한 방법에 따라 결정됩니다.

Redis용 ElastiCache 클러스터 또는 .rdb 파일 데이터 포함 복제 그룹 시드

• ElastiCache 콘솔 사용

Redis 엔진을 선택한 후 [Advanced Redis settings] 단원을 확장하고 [Import data to cluster]를 찾습니다. Seed RDB file S3 location(RDB 파일 S3 위치 시드) 상자에 파일의 Amazon S3 경로를 입력합니다. .rdb 파일이 여러 개 있으면 쉼표로 구분된 목록에 각 파일의 경로를 입력합니다. Amazon S3 경로는 myBucket/myFolder/myBackupFilename.rdb처럼 표시될 수 있습니다.

• AWS CLI 사용

create-cache-cluster 또는 create-replication-group 작업을 사용하는 경우 -- snapshot-arns 파라미터를 사용하여 각 .rdb 파일의 정규화된 ARN을 지정합니다. 예: arn:aws:s3:::myBucket/myFolder/myBackupFilename.rdb. ARN은 Amazon S3에 저장한 백업 파일로 확인되어야 합니다.

• ElastiCache API 사용

CreateCacheCluster 또는 CreateReplicationGroup ElastiCache API 작업을 사용하는 경우 SnapshotArns파라미터 를 사용하여 각 .rdb 파일의 정규화된 ARN을 지정합니다. 예: arn:aws:s3:::myBucket/myFolder/myBackupFilename.rdb. ARN은 Amazon S3에 저장한 백업 파일로 확인되어야 합니다.

Important

Redis(클러스터 모드 활성화됨) 클러스터 시드 시 새 클러스터 또는 복제 그룹에서 각 노드 그룹(샤드)을 구성해야 합니다. 이를 수행하려면 --node-group-configuration(API: NodeGroupConfiguration) 파라미터를 사용합니다. 자세한 내용은 다음 자료를 참조하십시오.

• CLI: AWS CLI 참조의 create-replication-group

• API: ElastiCache API 참조의 CreateReplicationGroup

클러스터 생성 프로세스 중에 Redis 백업의 데이터가 클러스터에 쓰여집니다. ElastiCache 이벤트 메시지를 보면서 프로세스를 모니터링할 수 있습니다. 그러려면 ElastiCache 콘솔 화면에서 Cache Events(캐시 이벤트)를 선택합니다. AWS ElastiCache 명령줄 인터페이스나 ElastiCache API를 사용하여 이벤트 메시지를 가져올 수도 있습니다. 자세한 내용은 ElastiCache 이벤트 보기 (p. 370) 단원을 참조하십시오.

백언 태그 지정

비용 할당 태그는 태그 값으로 인보이스의 비용을 그룹화하여 여러 AWS 제품에서 비용을 추적하는 수단입니다. 비용 할당 태그에 대해 자세히 알아보려면 비용 할당 태그 사용을 참조하십시오.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 백업에서 비용 할당 태그를 추가, 나열, 수정, 제거 또는 복사할 수 있습니다. 자세한 내용은 비용 할당 태그를 사용한 비용 모니터링 (p. 375) 단원을 참조하십시오.

백업 삭제

보존 기간 제한이 만료되면 자동 백업이 자동으로 삭제됩니다. 클러스터를 삭제하면 모든 자동 백업도 삭제 됩니다. 복제 그룹을 삭제하면 해당 그룹에 속한 클러스터의 모든 자동 백업도 삭제됩니다.

백업이 자동으로 생성되건 수동으로 생성되건 관계없이 ElastiCache에서는 언제든지 백업을 삭제할 수 있는 삭제 API를 제공합니다. 수동 백업에는 보존 제한이 없으므로 수동 삭제를 통해서만 수동 백업을 제거할 수 있습니다.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 백업을 삭제할 수 있습니다.

백업 삭제 (콘솔)

다음 절차에서는 ElastiCache 콘솔을 사용하여 백업을 삭제합니다.

백업을 삭제하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Backups]를 선택합니다.

백업 화면에 백업 목록이 나타납니다.

- 3. 삭제할 백업의 이름 왼쪽에 있는 상자를 선택합니다.
- 4. 삭제를 선택합니다.
- 5. 이 백업을 삭제하려면 [Delete Backup] 확인 화면에서 [Delete]를 선택합니다. 상태가 [deleting]으로 변경됩니다.

백업 삭제 (AWS CLI)

delete-snapshot AWS CLI 작업을 다음 파라미터와 함께 사용하여 백업을 삭제합니다.

• --snapshot-name - 삭제할 백업의 이름입니다.

다음 코드는 myBackup 백업을 삭제합니다.

aws elasticache delete-snapshot --snapshot-name myBackup

자세한 내용은 AWS CLI Command Reference의 delete-snapshot 항목을 참조하십시오.

백업 삭제 (ElastiCache API)

DeleteSnapshot API 작업을 다음 파라미터와 함께 사용하여 백업을 삭제합니다.

• SnapshotName - 삭제할 백업의 이름입니다.

다음 코드는 myBackup 백업을 삭제합니다.

https://elasticache.us-west-2.amazonaws.com/
?Action=DeleteSnapshot
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&SnapshotId=myBackup
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API Reference의 DeleteSnapshot 항목을 참조하십시오.

Redis용 ElastiCache의 AOF(Append Only Files)

기본적으로 ElastiCache의 Redis 노드에 있는 데이터는 메모리에만 상주하며 영구적이지 않습니다. 노드가 재부팅되거나 기본 물리적 서버에 하드웨어 장애가 발생하면 캐시의 데이터가 손실됩니다.

데이터 내구성이 필요할 경우 Redis AOF(append-only file) 기능을 사용할 수 있습니다. 이 기능이 활성화되면 노드가 캐시 데이터를 AOF로 변경하는 모든 명령을 씁니다. 노드가 재부팅되고 캐시 엔진이 시작되면, AOF가 "재생됩니다". 그렇게 되면, 모든 데이터가 손상되지 않은 웜 Redis 캐시가 생성됩니다.

AOF는 기본적으로 비활성화되어 있습니다. Redis를 실행하는 클러스터에 대해 AOF를 활성화하려면 appendonly 파라미터를 yes로 설정하여 파라미터 그룹을 만들어야 합니다. 그런 다음 해당 파라미터 그룹을 클러스터에 할당합니다. appendfsync 파라미터를 수정하여 Redis가 AOF 파일에 쓰는 빈도를 제어할수도 있습니다.

Important

AOF에서는 cache.t1.micro 및 cache.t2.* 노드가 지원되지 않습니다. 이러한 유형의 노드에서는 appendonly 파라미터 값이 무시됩니다.

다중 AZ 복제 그룹에는 AOF가 비활성화됩니다.

Redis 버전 2.8.22 이상에서는 AOF가 지원되지 않습니다.

Warning

AOF가 모든 실패 시나리오로부터 보호할 수 있는 것은 아닙니다. 예를 들어, 기본 물리적 서버에서 하드웨어 결함이 생겨 노드가 실패하면 ElastiCache가 다른 서버에 새로운 노드를 프로비저닝합니 다. 이 경우 AOF 파일을 더 이상 사용할 수 없으며 이 파일을 사용하여 데이터를 복구할 수 없습니 다. 따라서 Redis가 콜드 캐시로 다시 시작됩니다.

안정성과 복구 속도를 높이려면 클러스터의 여러 가용 영역에 읽기 전용 복제본을 하나 이상 생성하는 것이 좋습니다. AOF를 사용하는 대신 복제 그룹에 다중 AZ를 활성화합니다. 다중 AZ 복제 그룹에는 AOF가 비활성화됩니다.

결함 완화에 대한 자세한 내용은 Redis 실행 시 장애 완화 (p. 395) 단원을 참조하십시오.

자세한 내용은 다음 자료를 참조하십시오.

• Redis 특정 파라미터 (p. 274)

- 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132)
- 장애 완화 (p. 395)

Redis용 ElastiCache 클러스터 조정

애플리케이션에서 처리해야 하는 데이터의 양은 거의 정적이 아닙니다. 비즈니스가 성장하거나 수요에서 일 반적인 변동을 경험할 경우 데이터의 양이 증가하거나 감소합니다. 캐시를 자체적으로 관리할 경우 최고의 수요에 대해 충분한 하드웨어를 프로비저닝해야 하므로, 비용이 많이 들 수 있습니다. Amazon ElastiCache 를 사용하면 현재 수요에 맞게 조정할 수 있어, 사용한 만큼만 요금을 지불할 수 있습니다. ElastiCache를 통 해 수요에 맞게 캐시를 조정할 수 있습니다.

다음은 수행하려는 조정 작업에 대한 올바른 주제를 찾는 데 도움이 됩니다.

Redis 클러스터 조정

작업	Redis(클러스터 모드 비활성화됨)	Redis(클러스터 모드 활성화됨)
축소	클러스터에서 노드 제 거 (p. 96)	Redis(클러스터 모드 활성화됨)에 서 클러스터 조정 (p. 241)
노드 유형 변경	대형 노드 유형으로 • Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터확장 (p. 221) • 복제본이 있는 Redis 클러스터스케일 업 (p. 230)	
	소형 노드 유형으로 ■ 단일 노드 Redis 클러스터 스케일 다운 (p. 225) ■ 복제본이 있는 Redis 클러스터스케일 다운 (p. 236)	
노드 그룹의 수 변경	Redis(클러스터 모드 비활성화 됨) 클러스터가 지원되지 않음	Redis(클러스터 모드 활성화됨)에 서 클러스터 조정 (p. 241)

주제

- Redis(클러스터 모드 비활성화됨)에 대한 클러스터 조정 (p. 219)
- Redis(클러스터 모드 활성화됨)에서 클러스터 조정 (p. 241)

Redis(클러스터 모드 비활성화됨)에 대한 클러스터 조 정

Redis(클러스터 모드 비활성화됨) 클러스터는 샤드가 없는 단일 노드 클러스터이거나 샤드가 1개인 다중 노드 클러스터일 수 있습니다. 단일 노드 클러스터에서는 읽기와 쓰기에 모두 사용되는 노드 1개를 사용합니다. 다중 노드 클러스터에는 읽기/쓰기 기본 노드인 노드 1개와 0 ~ 5개의 읽기 전용 복제본 노드가 있습니다.

목차

- Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 조정 (p. 220)
 - Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 확장 (p. 221)
 - Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 확장 (콘솔) (p. 222)
 - 단일 노드 Redis 캐시 클러스터 확장 (AWS CLI) (p. 222)
 - 단일 노드 Redis 캐시 클러스터 확장 (ElastiCache API) (p. 224)
 - 단일 노드 Redis 클러스터 스케일 다운 (p. 225)
 - 단일 노드 Redis 클러스터 축소 (콘솔) (p. 226)
 - 단일 노드 Redis 캐시 클러스터 축소(AWS CLI) (p. 226)
 - 단일 노드 Redis 캐시 클러스터 축소(ElastiCache API) (p. 228)
- 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터 조정 (p. 230)
 - 복제본이 있는 Redis 클러스터 스케일 업 (p. 230)
 - 복제본이 있는 Redis 클러스터 확장 (콘솔) (p. 231)
 - Redis 복제 그룹 확장 (AWS CLI) (p. 232)
 - Redis 복제 그룹 확장 (ElastiCache API) (p. 234)
 - 복제본이 있는 Redis 클러스터 스케일 다운 (p. 236)
 - 복제본이 있는 Redis 클러스터 축소 (콘솔) (p. 236)
 - Redis 복제 그룹 축소 (AWS CLI) (p. 232)
 - Redis 복제 그룹 축소 (ElastiCache API) (p. 239)
 - 읽기 용량 늘리기 (p. 241)
 - 읽기 용량 줄이기 (p. 241)

Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 조정

Redis(클러스터 모드 비활성화됨) 노드는 모든 캐시의 데이터와 Redis 오버헤드를 포함할 만큼 충분히 커야합니다. Redis(클러스터 모드 비활성화됨) 클러스터의 데이터 용량을 변경하려면 수직으로 조정해야 합니다. 대형 노드 유형으로 확장하여 데이터 용량을 늘리거나 소형 노드 유형으로 축소하여 데이터 용량을 줄입니다.

Redis용 ElastiCache 확장 프로세스는 기존 데이터를 최대한 유지할 수 있도록 진행되며, 성공적인 Redis 복제가 필요합니다. Redis(클러스터 모드 비활성화됨) Redis 클러스터의 경우, Redis에 충분한 메모리를 사용할 수 있도록 하는 것이 좋습니다.

여러 Redis(클러스터 모드 비활성화됨) 클러스터로 데이터를 분할할 수 없습니다. 그러나 클러스터의 읽기 용량만 늘리거나 줄여야 하는 경우 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터를 생성하고 읽기 전용 복제본을 추가 또는 제거할 수 있습니다. 단일 노드 Redis 캐시 클러스터를 기본 클러스터로 사용하여 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터를 생성하려면 Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69) 단원을 참조하십시오.

복제본이 있는 클러스터를 생성하면 읽기 전용 복제본을 추가하여 읽기 용량을 늘릴 수 있습니다. 나중에 필요한 경우 읽기 전용 복제본을 제거하여 읽기 용량을 줄일 수 있습니다. 자세한 내용은 읽기 용량 늘리기 (p. 241) 또는 읽기 용량 줄이기 (p. 241) 단원을 참조하십시오.

읽기 용량을 조정할 수 있는 것 외에도 복제본이 있는 Redis(클러스터 모드 비활성화됨) 클러스터는 다른 비즈니스 혜택을 제공합니다. 자세한 내용은 고가용성을 위한 복제 그룹 사용 (p. 127) 단원을 참조하십시오.

Important

파라미터 그룹이 reserved-memory를 사용하여 Redis 오버헤드에 대한 메모리를 구분한 경우, 조정을 시작하기 전에 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹이 있어야 합니다. 또는 reserved-memory-percent를 사용하고 새 클러스터에 대해 해당 파라미터 그룹을 사용하도록 사용자 지정 파라미터 그룹을 수정할 수 있습니다.

reserved-memory-percent를 사용할 경우에는 이렇게 하지 않아도 됩니다.

자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

주제

- Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 확장 (p. 221)
- 단일 노드 Redis 클러스터 스케일 다운 (p. 225)

Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 확장

단일 노드 Redis 클러스터를 확장하면 ElastiCache에서는 ElastiCache 콘솔, AWS CLI 또는 ElastiCache API 사용 여부에 상관없이 다음 프로세스를 수행합니다.

- 1. 새 노드 유형이 있는 새 캐시 클러스터가 동일한 가용 영역에서 기존 캐시 클러스터로 실행됩니다.
- 2. 기존 캐시 클러스터의 캐시 데이터가 새 캐시 클러스터로 복사됩니다. 이 프로세스의 기간은 노드 유형 및 캐시 클러스터에 있는 데이터의 양에 따라 달라집니다.
- 3. 새 캐시 클러스터를 사용하여 읽기 및 쓰기를 수행합니다. 새 캐시 클러스터의 엔드포인트가 이전 캐시 클러스터의 엔드포인트와 동일하므로 애플리케이션에 있는 엔드포인트를 업데이트할 필요가 없습니다. DNS 항목이 업데이트되는 동안 기본 노드의 읽기 및 쓰기가 잠깐 중단될 수 있습니다
- 4. ElastiCache가 이전 캐시 클러스터를 삭제합니다.

다음 표에 표시된 대로 다음 유지 관리 기간에 대해 엔진 업그레이드가 예약된 경우 Redis 스케일 업 작업이 차단됩니다. 유지 관리 기간에 대한 자세한 내용은 유지 관리 관리 중 (p. 50) 단원을 참조하십시오.

차단된 Redis 작업

대기 중 작업	차단된 작업
스케일 업	즉시 엔진 업그레이드
엔진 업그레이드	즉시 스케일 업
스케일 업 및 엔진 업그레이드	즉시 스케일 업
	즉시 엔진 업그레이드

사용자를 차단하는 대기 중 작업이 있는 경우 다음 중 하나를 수행할 수 있습니다.

- [Apply immediately] 확인란을 선택 취소하여 다음 유지 관리 기간에 대해 Redis 스케일 업 작업을 예약합니다(CLI 사용: --no-apply-immediately, API 사용: ApplyImmediately=false).
- Redis 스케일 업 작업을 수행하기 위해 다음 유지 관리 기간(또는 그 이후)까지 기다립니다.
- [Apply Immediately] 확인란을 선택한 채로 이 캐시 클러스터 수정 사항에 Redis 엔진 업그레이드를 추가합니다(CLI 사용: --apply-immediately, API 사용: ApplyImmediately=true). 이렇게 하면 스케일 업 작업의 차단이 해제되어 엔진 업그레이드가 즉시 수행됩니다.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 단일 노드 Redis(클러스터 모드 비활성화됨) 클러스터를 확장할 수 있습니다.

Important

파라미터 그룹이 reserved-memory를 사용하여 Redis 오버헤드에 대한 메모리를 구분한 경우, 조정을 시작하기 전에 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹이 있어야 합니다. 또는 reserved-memory-percent를 사용하고 새 클러스터에 대해 해당 파라미터 그룹을 사용하도록 사용자 지정 파라미터 그룹을 수정할 수 있습니다. reserved-memory-percent를 사용할 경우에는 이렇게 하지 않아도 됩니다. 자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

Redis(클러스터 모드 비활성화됨)에 대한 단일 노드 클러스터 확장 (콘솔)

다음 절차에서는 ElastiCache Management Console을 사용하여 단일 노드 Redis 클러스터를 확장하는 방법에 대해 설명합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

단일 노드 Redis 클러스터를 확장하려면 (console)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
- 3. 클러스터 목록에서 스케일 업할 클러스터를 선택합니다(Clustered Redis 엔진이 아닌 Redis 엔진을 실행해야 함).
- 4. [Modify]를 선택합니다.
- 5. [Modify Cluster] 마법사에서 다음을 수행합니다.
 - a. [Node type] 목록에서 조정할 노드 유형을 선택합니다.
 - b. reserved-memory를 사용하여 메모리를 관리할 경우 [Parameter Group] 목록에서 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹을 선택합니다.
- 6. 스케일 업 프로세스를 즉시 수행하려면 [Apply immediately] 상자를 선택합니다. [Apply immediately] 상자를 선택하지 않으면 이 클러스터의 다음 유지 관리 기간 중 스케일 업 프로세스가 수행됩니다.
- 7. [Modify]를 선택합니다.

이전 단계에서 [Apply immediately]를 선택한 경우 클러스터의 상태가 수정 중으로 변경됩니다. 상태가 사용 가능으로 변경되면 수정이 완료되고 새 클러스터의 사용을 시작할 수 있습니다.

단일 노드 Redis 캐시 클러스터 확장 (AWS CLI)

다음 절차에서는 AWS CLI를 사용하여 단일 노드 Redis 캐시 클러스터를 확장하는 방법에 대해 설명합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

단일 노드 Redis 캐시 클러스터를 확장하려면 (AWS CLI)

- 1. 다음 파라미터와 함께 AWS CLI list-allowed-node-type-modifications명령을 실행하여 확장할 수 있는 노드 유형을 확인합니다.
 - --cache-cluster-id

Linux, macOS, or Unix의 경우:

```
aws elasticache list-allowed-node-type-modifications \
    --cache-cluster-id my-cache-cluster-id
```

Windows의 경우:

```
aws elasticache list-allowed-node-type-modifications ^
--cache-cluster-id my-cache-cluster-id
```

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
{
    "ScaleUpModifications": [
        "cache.m3.2xlarge",
        "cache.m3.large",
```

```
"cache.m3.xlarge",
        "cache.m4.10xlarge",
        "cache.m4.2xlarge",
        "cache.m4.4xlarge",
        "cache.m4.large",
        "cache.m4.xlarge"
        "cache.r3.2xlarge",
        "cache.r3.4xlarge",
        "cache.r3.8xlarge",
        "cache.r3.large",
        "cache.r3.xlarge"
       "ScaleDownModifications": [
        "cache.t2.micro",
        "cache.t2.small "
        "cache.t2.medium ",
           "cache.t1.small ",
    ٦,
}
```

자세한 내용은 AWS CLI 참조의 list-allowed-node-type-modifications 항목을 참조하십시오.

- 2. AWS CLI modify-cache-cluster 명령 및 다음 파라미터를 사용하여 확장할 캐시 클러스터 및 새로 운 대형 노드 유형을 지정하도록 기존 캐시 클러스터를 수정합니다.
 - --cache-cluster-id 확장 또는 축소할 캐시 클러스터의 이름입니다.
 - --cache-node-type 캐시 클러스터를 확장할 새 노드 유형입니다. 이 값은 1단계 의 list-allowed-node-type-modifications 명령에 의해 반환되는 노드 유형 중 하나여야 합니다.
 - --cache-parameter-group-name [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우이 파라미터를 생략할 수 있습니다.
 - --apply-immediately 스케일 업 프로세스가 즉시 적용되도록 합니다. 스케일 업 프로세스를 클 러스터의 다음 유지 관리 기간으로 연기하려면 --no-apply-immediately 파라미터를 사용하십시 오.

Linux. macOS. or Unix의 경우:

```
aws elasticache modify-cache-cluster \
--cache-cluster-id my-redis-cache-cluster \
--cache-node-type cache.m3.xlarge \
--cache-parameter-group-name redis32-m2-xl \
--apply-immediately
```

Windows의 경우:

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
{
    "CacheCluster": {
        "Engine": "redis",
        "CacheParameterGroup": {
```

```
"CacheNodeIdsToReboot": [],
             "CacheParameterGroupName": "default.redis3.2",
             "ParameterApplyStatus": "in-sync"
         },
         "SnapshotRetentionLimit": 1,
         "CacheClusterId": "my-redis-cache-cluster",
         "CacheSecurityGroups": [],
         "NumCacheNodes": 1,
         "SnapshotWindow": "00:00-01:00",
         "CacheClusterCreateTime": "2017-02-21T22:34:09.645Z",
         "AutoMinorVersionUpgrade": true,
         "CacheClusterStatus": "modifying",
         "PreferredAvailabilityZone": "us-west-2a",
         "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
         "CacheSubnetGroupName": "default",
         "EngineVersion": "3.2.4",
         "PendingModifiedValues": {
             "CacheNodeType": "cache.m3.2xlarge"
         "PreferredMaintenanceWindow": "tue:11:30-tue:12:30",
         "CacheNodeType": "cache.m3.medium"
    }
}
```

자세한 내용은 AWS CLI 참조의 modify-cache-cluster 항목을 참조하십시오.

- 3. --apply-immediately를 사용한 경우 AWS CLI describe-cache-clusters 명령을 다음 파라미터와 함께 사용하여 새 캐시 클러스터의 상태를 확인합니다. 상태가 사용 가능으로 변경되면 새로운 대형 캐시 클러스터의 사용을 시작할 수 있습니다.
 - --cache-cache cluster-id 단일 노드 Redis 캐시 클러스터의 이름입니다. 모든 캐시 클러스터 대신 특정 캐시 클러스터를 설명하려면 이 파라미터를 사용하십시오.

```
aws elasticache describe-cache-clusters --cache-cluster-id my-redis-cache-cluster
```

자세한 내용은 AWS CLI 참조의 describe-cache-clusters 항목을 참조하십시오.

단일 노드 Redis 캐시 클러스터 확장 (ElastiCache API)

다음 절차에서는 ElastiCache API를 사용하여 단일 노드 Redis 캐시 클러스터를 확장하는 방법에 대해 설명합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

단일 노드 Redis 캐시 클러스터를 확장하려면 (ElastiCache API)

- 1. 다음 파라미터와 함께 ElastiCache API ListAllowedNodeTypeModifications 작업을 실행하여 확장할 수 있는 노드 유형을 확인합니다.
 - CacheClusterId 확장할 단일 노드 Redis 캐시 클러스터의 이름입니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=ListAllowedNodeTypeModifications
    &CacheClusterId=MyRedisCacheCluster
    &Version=2015-02-02
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T192317Z
    &X-Amz-Credential=<credential>
```

자세한 내용은 Amazon ElastiCache API 참조의 ListAllowedNodeTypeModifications을 참조하십시오.

- 2. ModifyCacheCluster ElastiCache API 작업 및 다음 파라미터를 사용하여 확장할 캐시 클러스터 및 새로운 대형 노드 유형을 지정하도록 기존 캐시 클러스터를 수정합니다.
 - CacheClusterId 확장할 캐시 클러스터의 이름입니다.
 - CacheNodeType 캐시 클러스터를 확장할 새로운 대형 노드 유형입니다. 이 값은 1단계의 ListAllowedNodeTypeModifications 작업에 의해 반환되는 노드 유형 중 하나여야 합니다.
 - CacheParameterGroupName [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우 이 파라미터를 생략할 수 있습니다.
 - ApplyImmediately 스케일 업 프로세스가 즉시 수행되도록 하려면 true로 설정합니다. 스케일 업 프로세스를 클러스터의 다음 유지 관리 기간으로 연기하려면 ApplyImmediately=false를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=ModifyCacheCluster
&ApplyImmediately=true
&CacheClusterId=MyRedisCacheCluster
&CacheNodeType=cache.m3.xlarge
&CacheParameterGroupName redis32-m2-xl
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 ModifyCacheCluster을 참조하십시오.

- 3. ApplyImmediately=true를 사용한 경우 ElastiCache API DescribeCacheClusters 작업을 다음 파라미터와 함께 사용하여 새 캐시 클러스터의 상태를 확인합니다. 상태가 사용 가능으로 변경되면 새로운 대형 캐시 클러스터의 사용을 시작할 수 있습니다.
 - CacheClusterId 단일 노드 Redis 캐시 클러스터의 이름입니다. 모든 캐시 클러스터 대신 특정 캐 시 클러스터를 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheClusters
&CacheClusterId=MyRedisCacheCluster
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 DescribeCacheClusters을 참조하십시오.

단일 노드 Redis 클러스터 스케일 다운

다음 단원에서는 단일 노드 Redis 클러스터를 소형 노드 유형으로 축소하는 방법을 살펴봅니다. 새 Redis 클러스터의 장기적인 성공을 위해 새로운 소형 노드 유형이 모든 데이터와 Redis 오버헤드를 수용할 만큼 충분히 큰지 확인하는 것이 중요합니다. 자세한 내용은 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404) 단원을 참조하십시오.

주제

- 단일 노드 Redis 클러스터 축소 (콘솔) (p. 226)
- 단일 노드 Redis 캐시 클러스터 축소(AWS CLI) (p. 226)
- 단일 노드 Redis 캐시 클러스터 축소(ElastiCache API) (p. 228)

단일 노드 Redis 클러스터 축소 (콘솔)

다음 절차는 ElastiCache 콘솔을 사용하여 단일 노드 Redis 클러스터를 소형 노드 유형으로 축소하는 방법을 안내합니다.

Important

파라미터 그룹이 reserved-memory를 사용하여 Redis 오버헤드에 대한 메모리를 구분한 경우, 조정을 시작하기 전에 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹이 있어야 합니다. 또는 reserved-memory-percent를 사용하고 새 클러스터에 대해 해당 파라미터 그룹을 사용하도록 사용자 지정 파라미터 그룹을 수정할 수 있습니다. reserved-memory-percent를 사용할 경우에는 이렇게 하지 않아도 됩니다. 자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

단일 노드 Redis 클러스터를 축소하려면 (console)

- 1. 소형 노드 유형이 데이터 및 오버헤드 요구 사항에 적합한지 확인합니다.
- 2. 파라미터 그룹이 reserved-memory를 사용하여 Redis 오버헤드에 대한 메모리를 구분한 경우, 새 노드 유형에 대해 올바른 메모리 양을 구분하는 사용자 지정 파라미터 그룹이 있어야 합니다.

또는 reserved-memory-percent를 사용하여 사용자 지정 파라미터 그룹을 수정할 수 있습니다. 자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

- 3. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 4. 클러스터 목록에서 스케일 다운할 클러스터를 선택합니다. 이 클러스터는 Clustered Redis 엔진이 아닌 Redis 엔진을 실행해야 합니다.
- 5. [Modify]를 선택합니다.
- 6. [Modify Cluster] 마법사에서 다음을 수행합니다.
 - a. [Node type] 목록에서 스케일 다운할 노드 유형을 선택합니다.
 - b. reserved-memory를 사용하여 메모리를 관리할 경우 [Parameter Group] 목록에서 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹을 선택합니다.
- 스케일 다운 프로세스를 즉시 수행하려면 [Apply immediately] 확인란을 선택합니다. [Apply immediately] 확인란을 선택하지 않고 비워 두면 이 클러스터의 다음 유지 관리 기간 중 스케일 다운 프 로세스가 수행됩니다.
- 8. 수정을 선택합니다.
- 9. 클러스터의 상태가 수정 중에서 사용 가능으로 변경되면 클러스터가 새 노드 유형으로 조정된 것입니다. 애플리케이션에서 엔드포인트를 업데이트할 필요가 없습니다.

단일 노드 Redis 캐시 클러스터 축소(AWS CLI)

다음 절차에서는 AWS CLI를 사용하여 단일 노드 Redis 캐시 클러스터를 축소하는 방법에 대해 설명합니다.

단일 노드 Redis 캐시 클러스터를 축소하려면 (AWS CLI)

- 1. 다음 파라미터와 함께 AWS CLI list-allowed-node-type-modifications 명령을 실행하여 축소할 수 있는 노드 유형을 확인합니다.
 - --cache-cluster-id

Linux, macOS, or Unix의 경우:

```
aws elasticache list-allowed-node-type-modifications \
    --cache-cluster-id my-cache-cluster-id
```

Windows의 경우:

```
aws elasticache list-allowed-node-type-modifications ^
--cache-cluster-id my-cache-cluster-id
```

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
{
     "ScaleUpModifications": [
         "cache.m3.2xlarge",
         "cache.m3.large",
         "cache.m3.xlarge",
         "cache.m4.10xlarge".
         "cache.m4.2xlarge",
         "cache.m4.4xlarge",
         "cache.m4.large",
         "cache.m4.xlarge"
         "cache.r3.2xlarge",
         "cache.r3.4xlarge",
         "cache.r3.8xlarge",
         "cache.r3.large",
         "cache.r3.xlarge"
        "ScaleDownModifications": [
         "cache.t2.micro",
         "cache.t2.small "
         "cache.t2.medium ",
            "cache.t1.small ",
    ],
}
```

자세한 내용은 AWS CLI 참조의 list-allowed-node-type-modifications 항목을 참조하십시오.

- 2. AWS CLI modify-cache-cluster 명령 및 다음 파라미터를 사용하여 축소할 캐시 클러스터 및 새롭고 더 작은 노드 유형을 지정하도록 기존 캐시 클러스터를 수정합니다.
 - --cache-cluster-id 축소할 캐시 클러스터의 이름입니다.
 - --cache-node-type 캐시 클러스터를 확장할 새 노드 유형입니다. 이 값은 1단계 의 list-allowed-node-type-modifications 명령에 의해 반환되는 노드 유형 중 하나여야 합니다.
 - --cache-parameter-group-name [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우이 파라미터를 생략할 수 있습니다.
 - --apply-immediately 축소 프로세스가 즉시 적용되도록 합니다. 스케일 업 프로세스를 클러스 터의 다음 유지 관리 기간으로 연기하려면 --no-apply-immediately 파라미터를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-cluster \
--cache-cluster-id my-redis-cache-cluster \
--cache-node-type cache.m3.xlarge \
--cache-parameter-group-name redis32-m2-xl \
--apply-immediately
```

Windows의 경우:

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
{
     "CacheCluster": {
         "Engine": "redis",
         "CacheParameterGroup": {
             "CacheNodeIdsToReboot": [],
             "CacheParameterGroupName": "default.redis3.2",
             "ParameterApplyStatus": "in-sync"
         "SnapshotRetentionLimit": 1,
         "CacheClusterId": "my-redis-cache-cluster",
         "CacheSecurityGroups": [],
         "NumCacheNodes": 1,
         "SnapshotWindow": "00:00-01:00",
         "CacheClusterCreateTime": "2017-02-21T22:34:09.645Z",
         "AutoMinorVersionUpgrade": true,
         "CacheClusterStatus": "modifying",
         "PreferredAvailabilityZone": "us-west-2a",
         "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
         "CacheSubnetGroupName": "default",
         "EngineVersion": "3.2.4",
         "PendingModifiedValues": {
             "CacheNodeType": "cache.m3.2xlarge"
         "PreferredMaintenanceWindow": "tue:11:30-tue:12:30",
         "CacheNodeType": "cache.m3.medium"
    }
}
```

자세한 내용은 AWS CLI 참조의 modify-cache-cluster 항목을 참조하십시오.

- 3. --apply-immediately를 사용한 경우 AWS CLI describe-cache-clusters 명령을 다음 파라미터와 함께 사용하여 새 캐시 클러스터의 상태를 확인합니다. 상태가 사용 가능으로 변경되면 새로운 대형 캐시 클러스터의 사용을 시작할 수 있습니다.
 - --cache-cache cluster-id 단일 노드 Redis 캐시 클러스터의 이름입니다. 모든 캐시 클러스터 대신 특정 캐시 클러스터를 설명하려면 이 파라미터를 사용하십시오.

```
aws elasticache describe-cache-clusters --cache-cluster-id my-redis-cache-cluster
```

자세한 내용은 AWS CLI 참조의 describe-cache-clusters 항목을 참조하십시오.

단일 노드 Redis 캐시 클러스터 축소(ElastiCache API)

다음 절차에서는 ElastiCache API를 사용하여 단일 노드 Redis 캐시 클러스터를 축소하는 방법에 대해 설명합니다.

단일 노드 Redis 캐시 클러스터를 축소하려면 (ElastiCache API)

- 1. 다음 파라미터와 함께 ElastiCache API ListAllowedNodeTypeModifications 작업을 실행하여 축소할 수 있는 노드 유형을 확인합니다.
 - CacheClusterId 축소할 단일 노드 Redis 캐시 클러스터의 이름입니다.

https://elasticache.us-west-2.amazonaws.com/
 ?Action=ListAllowedNodeTypeModifications
 &CacheClusterId=MyRedisCacheCluster
 &Version=2015-02-02
 &SignatureVersion=4
 &SignatureMethod=HmacSHA256
 &Timestamp=20150202T192317Z
 &X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 ListAllowedNodeTypeModifications을 참조하십시오.

- 2. ModifyCacheCluster ElastiCache API 작업 및 다음 파라미터를 사용하여 확장할 캐시 클러스터 및 새로운 대형 노드 유형을 지정하도록 기존 캐시 클러스터를 수정합니다.
 - CacheClusterId 축소할 캐시 클러스터의 이름입니다.
 - CacheNodeType 캐시 클러스터를 축소할 새롭고 더 큰 노드 유형입니다. 이 값은 1단계의 ListAllowedNodeTypeModifications 작업에 의해 반환되는 노드 유형 중 하나여야 합니다.
 - CacheParameterGroupName [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우 이 파라미터를 생략할 수 있습니다.
 - ApplyImmediately 축소 프로세스가 즉시 수행되도록 하려면 true로 설정합니다. 스케일 업 프로세스를 클러스터의 다음 유지 관리 기간으로 연기하려면 ApplyImmediately=false를 사용하십시오

https://elasticache.us-west-2.amazonaws.com/
?Action=ModifyCacheCluster
&ApplyImmediately=true
&CacheClusterId=MyRedisCacheCluster
&CacheNodeType=cache.m3.xlarge
&CacheParameterGroupName redis32-m2-xl
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 ModifyCacheCluster을 참조하십시오.

- 3. ApplyImmediately=true를 사용한 경우 ElastiCache API DescribeCacheClusters 작업을 다음 파라미터와 함께 사용하여 새 캐시 클러스터의 상태를 확인합니다. 상태가 사용 가능으로 변경되면 새롭고 더 작은 캐시 클러스터를 사용할 수 있습니다.
 - CacheClusterId 단일 노드 Redis 캐시 클러스터의 이름입니다. 모든 캐시 클러스터 대신 특정 캐 시 클러스터를 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/ ?Action=DescribeCacheClusters &CacheClusterId=MyRedisCacheCluster &Version=2015-02-02

&SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20150202T192317Z &X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 DescribeCacheClusters을 참조하십시오.

복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 클러스터 조 정

복제본 노드가 있는 Redis 클러스터(API/CLI에서 복제 그룹이라고 함)는 자동 장애 조치가 활성화된 다중 AZ가 있는 복제를 통해 고가용성을 제공합니다. 복제본 노드가 있는 클러스터는 최대 6개의 Redis 클러스터의 논리적 모음으로, 이 중 기본 노드는 읽기 및 쓰기 요청을 모두 제공할 수 있습니다. 클러스터의 다른 모든 노드는 기본 노드의 읽기 전용 복제본입니다. 기본에 작성된 데이터는 클러스터의 모든 읽기 전용 복제본으로 비동기식으로 복제됩니다. Redis(클러스터 모드 비활성화됨)는 여러 클러스터로의 데이터 분할을 지원하지 않으므로 Redis(클러스터 모드 비활성화됨) 복제 그룹의 각 클러스터에는 전체 캐시 데이터 세트가 포함됩니다. Redis(클러스터 모드 활성화됨) 클러스터는 최대 90개 샤드로의 데이터 분할을 지원합니다.

클러스터의 데이터 용량을 변경하려면 대형 노드 유형으로 스케일 업하거나 소형 노드 유형으로 스케일 다운 해야 합니다.

클러스터의 읽기 용량을 변경하려면 최대 5개의 추가 읽기 전용 복제본을 추가하거나 읽기 전용 복제본을 제 거하십시오.

ElastiCache 확장 프로세스는 기존 데이터를 최대한 유지할 수 있도록 진행되며, 성공적인 Redis 복제가 필요합니다. 복제본이 있는 Redis 클러스터의 경우, Redis에 충분한 메모리를 사용할 수 있도록 하는 것이 좋습니다.

관련 주제

- 고가용성을 위한 복제 그룹 사용 (p. 127)
- 복제: Redis(클러스터 모드 비활성화됨) 대 Redis(클러스터 모드 활성화됨) (p. 130)
- 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132)
- 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)

주제

- 복제본이 있는 Redis 클러스터 스케일 업 (p. 230)
- 복제본이 있는 Redis 클러스터 스케일 다운 (p. 236)
- 읽기 용량 늘리기 (p. 241)
- 읽기 용량 줄이기 (p. 241)

복제본이 있는 Redis 클러스터 스케일 업

Amazon ElastiCache는 Redis(클러스터 모드 비활성화됨) 복제 그룹 확장에 대한 콘솔, CLI 및 API 지원을 제공합니다.

스케일 업 프로세스가 시작되면 ElastiCache는 다음을 수행합니다.

- 1. 새 노드 유형을 사용하여 복제 그룹을 시작합니다.
- 2. 현재 기본 노드에서 새 기본 노드로 모든 데이터를 복사합니다.
- 3. 새 읽기 전용 복제본을 새 기본 노드와 동기화합니다.
- 4. DNS 항목이 새 노드를 가리키도록 해당 항목을 업데이트합니다. 따라서 애플리케이션의 엔드포인트를 업데이트할 필요가 없습니다. Redis 5.0.5 이상의 경우, 클러스터가 온라인 상태에서 들어오는 요청을

계속 처리하는 동안 자동 장애 조치가 활성화된 클러스터를 조정할 수 있습니다. 버전 5.0.4 이하의 경우, DNS 항목이 업데이트되는 동안 기본 노드에서 이전 버전에 대한 읽기 및 쓰기가 잠깐 중단될 수 있습니다.

5. 이전 노드(CLI/API: 복제 그룹)를 삭제합니다.

이 프로세스의 기간은 노드 유형 및 클러스터에 있는 데이터의 양에 따라 달라집니다.

다음 표에 표시된 대로, 클러스터의 다음 유지 관리 기간에 대해 엔진 업그레이드가 예약된 경우 Redis 스케일 업 작업이 차단됩니다.

차단된 Redis 작업

대기 중 작업	차단된 작업
스케일 업	즉시 엔진 업그레이드
엔진 업그레이드	즉시 스케일 업
스케일 업 및 엔진 업그레이드	즉시 스케일 업
	즉시 엔진 업그레이드

사용자를 차단하는 대기 중 작업이 있는 경우 다음 중 하나를 수행할 수 있습니다.

- [Apply immediately] 확인란을 선택 취소하여 다음 유지 관리 기간에 대해 Redis 스케일 업 작업을 예약합니다(CLI 사용: --no-apply-immediately, API 사용: ApplyImmediately=false).
- Redis 스케일 업 작업을 수행하기 위해 다음 유지 관리 기간(또는 그 이후)까지 기다립니다.
- [Apply Immediately] 확인란을 선택한 채로 이 캐시 클러스터 수정 사항에 Redis 엔진 업그레이드를 추가합니다(CLI 사용: --apply-immediately, API 사용: ApplyImmediately=true). 이렇게 하면 스케일 업 작업의 차단이 해제되어 엔진 업그레이드가 즉시 수행됩니다.

다음 단원에서는 ElastiCache 콘솔, AWS CLI 및 ElastiCache API를 사용하여 복제본이 있는 Redis 클러스터를 확장하는 방법에 대해 설명합니다.

Important

파라미터 그룹이 reserved-memory를 사용하여 Redis 오버헤드에 대한 메모리를 구분한 경우, 조정을 시작하기 전에 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹이 있어야 합니다. 또는 reserved-memory-percent를 사용하고 새 클러스터에 대해 해당 파라미터 그룹을 사용하도록 사용자 지정 파라미터 그룹을 수정할 수 있습니다.

reserved-memory-percent를 사용할 경우에는 이렇게 하지 않아도 됩니다.

자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

복제본이 있는 Redis 클러스터 확장 (콘솔)

대형 노드 유형으로 스케일 업하는 데 걸리는 시간은 노드 유형 및 현재 클러스터에 있는 데이터의 양에 따라 달라집니다.

다음 절차는 ElastiCache 콘솔을 사용하여 복제본이 있는 클러스터를 현재 노드 유형에서 새로운 대형 노드유형으로 조정합니다. 이 프로세스 중 DNS 항목이 업데이트되는 동안 기본 노드에서 다른 버전에 대한 읽기 및 쓰기가 잠깐 중단될 수 있습니다. 5.0.5 버전에서 실행 중인 노드의 경우 1초 미만, 이전 버전의 경우 몇 초동안 가동 중지가 발생할 수 있습니다.

복제본이 있는 Redis 클러스터를 확장하려면 (console)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.

- 3. 클러스터 목록에서 스케일 업할 클러스터를 선택합니다. 이 클러스터는 Clustered Redis 엔진이 아닌 Redis 엔진을 실행해야 합니다.
- 4. [Modify]를 선택합니다.
- 5. [Modify Cluster] 마법사에서 다음을 수행합니다.
 - a. [Node type] 목록에서 조정할 노드 유형을 선택합니다. 모든 노드 유형을 축소할 수 있는 것은 아닙니다.
 - b. reserved-memory를 사용하여 메모리를 관리할 경우 [Parameter Group] 목록에서 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹을 선택합니다.
- 6. 스케일 업 프로세스를 즉시 수행하려면 [Apply immediately] 확인란을 선택합니다. [Apply immediately] 확인란을 선택하지 않고 비워 두면 이 클러스터의 다음 유지 관리 기간 중 스케일 업 프로세스가 수행됩니다.
- 7. 수정을 선택합니다.
- 8. 클러스터의 상태가 수정 중에서 사용 가능으로 변경되면 클러스터가 새 노드 유형으로 조정된 것입니다. 애플리케이션에서 엔드포인트를 업데이트할 필요가 없습니다.

Redis 복제 그룹 확장 (AWS CLI)

다음 절차는 AWS CLI를 사용하여 복제 그룹을 현재 노드 유형에서 새롭고 더 큰 노드 유형으로 조정합니다. 이 프로세스 중 상태가 수정 중에서 사용 가능으로 변경될 때까지 애플리케이션과 기본 캐시 클러스터 간의 모든 읽기 및 쓰기가 차단됩니다.

대형 노드 유형으로 스케일 업하는 데 걸리는 시간은 노드 유형 및 현재 캐시 클러스터에 있는 데이터의 양에 따라 달라집니다.

Redis 복제 그룹을 확장하려면 (AWS CLI)

- 1. 다음 파라미터와 함께 AWS CLllist-allowed-node-type-modifications 명령을 실행하여 확장할 수 있는 노드 유형을 확인합니다.
 - --replication-group-id 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명 하려면 이 파라미터를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache list-allowed-node-type-modifications \
    --replication-group-id my-repl-group
```

Windows의 경우:

이 작업의 출력은 다음과 같습니다(JSON 형식).

```
{
    "ScaleUpModifications": [
        "cache.m3.2xlarge",
        "cache.m3.large",
        "cache.m3.xlarge",
        "cache.m4.10xlarge",
        "cache.m4.2xlarge",
        "cache.m4.4xlarge",
        "cache.m4.4xlarge",
        "cache.m4.1arge",
        "cache.m4.xlarge",
        "cache.m4.xlarge",
        "cache.m4.xlarge",
        "cache.m4.xlarge",
        "cache.m4.xlarge",
```

```
"cache.r3.2xlarge",
    "cache.r3.4xlarge",
    "cache.r3.8xlarge",
    "cache.r3.large",
    "cache.r3.xlarge"
]
}
```

자세한 내용은 AWS CLI 참조의 list-allowed-node-type-modifications 항목을 참조하십시오.

- 2. 다음 파라미터와 함께 AWS CLI modify-replication-group 명령을 사용하여 현재 복제 그룹을 새 노드 유형으로 확장합니다.
 - --replication-group-id 복제 그룹의 이름입니다.
 - --cache-node-type 이 복제 그룹에 있는 캐시 클러스터의 새로운 대형 노드 유형입니다. 이 값은 1단계의 list-allowed-node-type-modifications 명령에 의해 반환되는 인스턴스 유형 중 하나여야 합니다.
 - --cache-parameter-group-name [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우이 파라미터를 생략할 수 있습니다.
 - --apply-immediately 스케일 업 프로세스가 즉시 적용되도록 합니다. 스케일 업 작업을 다음 유지 관리 기간으로 연기하려면 --no-apply-immediately를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group \
--replication-group-id my-repl-group \
--cache-node-type cache.m3.xlarge \
--cache-parameter-group-name redis32-m3-2xl \
--apply-immediately
```

Windows의 경우:

```
aws elasticache modify-replication-group ^
--replication-group-id my-repl-group ^
--cache-node-type cache.m3.xlarge ^
--cache-parameter-group-name redis32-m3-2xl \
--apply-immediately
```

이 명령의 출력은 다음과 같습니다(JSON 형식).

```
"CurrentRole": "replica",
    "PreferredAvailabilityZone": "us-west-2c",
     "CacheNodeId": "0001",
     "ReadEndpoint": {
     "Port": 6379,
      "Address": "my-repl-group-002.8fdx4s.0001.usw2.cache.amazonaws.com"
    "CacheClusterId": "my-repl-group-002"
   }
   ٦,
   "NodeGroupId": "0001",
   "PrimaryEndpoint": {
   "Port": 6379,
    "Address": "my-repl-group.8fdx4s.ng.0001.usw2.cache.amazonaws.com"
 }],
  "ReplicationGroupId": "my-repl-group",
 "SnapshotRetentionLimit": 1,
 "AutomaticFailover": "disabled",
 "SnapshotWindow": "12:00-13:00",
  "SnapshottingClusterId": "my-repl-group-002",
 "MemberClusters": [
  "my-repl-group-001",
  "my-repl-group-002"
 ],
 "PendingModifiedValues": {}
}
```

자세한 내용은 AWS CLI 참조의 modify-replication-group 항목을 참조하십시오.

- 3. --apply-immediately 파라미터를 사용한 경우 AWS CLI describe-replication-group 명령을 다음 파라미터와 함께 사용하여 복제 그룹의 상태를 모니터링합니다. 상태가 계속 수정 중이면 5.0.5 버전에서 실행 중인 노드의 경우 1초 미만의 가동 중지가 발생할 수 있고, DNS 항목이 업데이트되는 동안 기본 노드에서 이전 버전에 대한 읽기 및 쓰기가 잠깐 중단될 수 있습니다.
 - --replication-group-id 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명 하려면 이 파라미터를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-replication-group \
    --replication-group-id my-replication-group
```

Windows의 경우:

```
aws elasticache describe-replication-groups ^
--replication-group-id my-replication-group
```

자세한 내용은 AWS CLI 참조의 describe-replication-groups을 참조하십시오.

Redis 복제 그룹 확장 (ElastiCache API)

다음 절차는 ElastiCache API를 사용하여 복제 그룹을 현재 노드 유형에서 새로운 대형 노드 유형으로 조정합니다. 이 프로세스 중 상태가 수정 중에서 사용 가능으로 변경될 때까지 애플리케이션과 기본 캐시 클러스터 간의 모든 읽기 및 쓰기가 차단됩니다. 그러나 읽기 전용 복제본 캐시 클러스터에서 읽기는 계속 중단되지 않습니다.

대형 노드 유형으로 스케일 업하는 데 걸리는 시간은 노드 유형 및 현재 캐시 클러스터에 있는 데이터의 양에 따라 달라집니다.

Redis 복제 그룹을 확장하려면 (ElastiCache API)

- 1. 다음 파라미터와 함께 ElastiCache API ListAllowedNodeTypeModifications 작업을 사용하여 확장할 수 있는 노드 유형을 확인합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=ListAllowedNodeTypeModifications
&ReplicationGroupId=MyReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 ListAllowedNodeTypeModifications을 참조하십시오.

- 2. 다음 파라미터와 함께 ModifyRedplicationGroup ElastiCache API 작업을 사용하여 현재 복제 그룹을 새 노드 유형으로 확장합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다.
 - CacheNodeType 이 복제 그룹에 있는 캐시 클러스터의 새로운 대형 노드 유형입니다. 이 값은 1단 계의 ListAllowedNodeTypeModifications 작업에 의해 반환되는 인스턴스 유형 중 하나여야합니다.
 - CacheParameterGroupName [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우 이 파라미터를 생략할 수 있습니다.
 - ApplyImmediately 스케일 업 프로세스가 즉시 적용되도록 하려면 true로 설정합니다. 스케일 업 프로세스를 다음 유지 관리 기간으로 연기하려면 ApplyImmediately=false를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/ ?Action=ModifyReplicationGroup &ApplyImmediately=true &CacheNodeType=cache.m3.2xlarge &CacheParameterGroupName=redis32-m3-2xl &ReplicationGroupId=myReplGroup &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20141201T220302Z &Version=2014-12-01 &X-Amz-Algorithm=AWS4-HMAC-SHA256 &X-Amz-Date=20141201T220302Z &X-Amz-SignedHeaders=Host &X-Amz-Expires=20141201T220302Z &X-Amz-Credential = < credential > &X-Amz-Signature=<signature>

자세한 내용은 Amazon ElastiCache API 참조의 ModifyReplicationGroup을 참조하십시오.

3. ApplyImmediately=true를 사용한 경우 ElastiCache API DescribeReplicationGroups 작업을 다음 파라미터와 함께 사용하여 복제 그룹의 상태를 모니터링합니다. 상태가 수정 중에서 사용 가능으로 변경되면 스케일 업된 새 복제 그룹에 쓰기를 시작할 수 있습니다.

• ReplicationGroupId – 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReplicationGroups
&ReplicationGroupId=MyReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 DescribeReplicationGroups을 참조하십시오.

복제본이 있는 Redis 클러스터 스케일 다운

다음 단원에서는 복제본 노드가 있는 Redis(클러스터 모드 비활성화됨) 캐시 클러스터를 소형 노드 유형으로 축소하는 방법을 살펴봅니다. 성공을 위해 새로운 소형 노드 유형이 모든 데이터와 오버헤드를 수용할 만큼 충분히 큰지 확인하는 것이 매우 중요합니다. 자세한 내용은 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404) 단원을 참조하십시오.

Important

파라미터 그룹이 reserved-memory를 사용하여 Redis 오버헤드에 대한 메모리를 구분한 경우, 조정을 시작하기 전에 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹이 있어야 합니다. 또는 reserved-memory-percent를 사용하고 새 클러스터에 대해 해당 파라미터 그룹을 사용하도록 사용자 지정 파라미터 그룹을 수정할 수 있습니다. reserved-memory-percent를 사용할 경우에는 이렇게 하지 않아도 됩니다.

자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

주제

- 복제본이 있는 Redis 클러스터 축소 (콘솔) (p. 236)
- Redis 복제 그룹 축소 (AWS CLI) (p. 232)
- Redis 복제 그룹 축소 (ElastiCache API) (p. 239)

복제본이 있는 Redis 클러스터 축소 (콘솔)

다음 절차는 ElastiCache 콘솔을 사용하여 복제본 노드가 있는 Redis 클러스터를 소형 노드 유형으로 조정합니다.

복제본 노드가 있는 Redis 클러스터를 축소하려면 (console)

- 1. 소형 노드 유형이 데이터 및 오버헤드 요구 사항에 적합한지 확인합니다.
- 2. 파라미터 그룹이 reserved-memory를 사용하여 Redis 오버헤드에 대한 메모리를 구분한 경우, 새 노드 유형에 대해 올바른 메모리 양을 구분하는 사용자 지정 파라미터 그룹이 있어야 합니다.

또는 reserved-memory-percent를 사용하여 사용자 지정 파라미터 그룹을 수정할 수 있습니다. 자세한 내용은 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

- 3. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 4. 클러스터 목록에서 스케일 다운할 클러스터를 선택합니다. 이 클러스터는 Clustered Redis 엔진이 아닌 Redis 엔진을 실행해야 합니다.
- 5. [Modify]를 선택합니다.
- 6. [Modify Cluster] 마법사에서 다음을 수행합니다.

- a. [Node type] 목록에서 스케일 다운할 노드 유형을 선택합니다.
- b. reserved-memory를 사용하여 메모리를 관리할 경우 [Parameter Group] 목록에서 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 파라미터 그룹을 선택합니다.
- 7. 스케일 다운 프로세스를 즉시 수행하려면 [Apply immediately] 확인란을 선택합니다. [Apply immediately] 확인란을 선택하지 않고 비워 두면 이 클러스터의 다음 유지 관리 기간 중 스케일 다운 프로세스가 수행됩니다.
- 8. 수정을 선택합니다.
- 9. 클러스터의 상태가 수정 중에서 사용 가능으로 변경되면 클러스터가 새 노드 유형으로 조정된 것입니다. 애플리케이션에서 엔드포인트를 업데이트할 필요가 없습니다.

Redis 복제 그룹 축소 (AWS CLI)

다음 절차는 AWS CLI를 사용하여 복제 그룹을 현재 노드 유형에서 새롭고 더 작은 노드 유형으로 조정합니다. 이 프로세스 중 상태가 수정 중에서 사용 가능으로 변경될 때까지 애플리케이션과 기본 캐시 클러스터 간의 모든 읽기 및 쓰기가 차단됩니다.

더 작은 노드 유형으로 축소하는 데 걸리는 시간은 노드 유형 및 현재 캐시 클러스터에 있는 데이터의 양에 따라 달라집니다.

Redis 복제 그룹을 축소하려면(AWS CLI)

- 1. 다음 파라미터와 함께 AWS CLI list-allowed-node-type-modifications 명령을 실행하여 축소할 수 있는 노드 유형을 확인합니다.
 - --replication-group-id 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명 하려면 이 파라미터를 사용하십시오.

Linux. macOS. or Unix의 경우:

```
aws elasticache list-allowed-node-type-modifications \
    --replication-group-id my-repl-group
```

Windows의 경우:

이 작업의 출력은 다음과 같습니다(JSON 형식).

```
{
     "ScaleDownModifications": [
         "cache.m3.2xlarge",
         "cache.m3.large",
         "cache.m3.xlarge",
         "cache.m4.10xlarge",
         "cache.m4.2xlarge",
         "cache.m4.4xlarge",
         "cache.m4.large",
         "cache.m4.xlarge",
         "cache.r3.2xlarge",
         "cache.r3.4xlarge",
         "cache.r3.8xlarge",
         "cache.r3.large"
         "cache.r3.xlarge"
     ]
```

}

자세한 내용은 AWS CLI 참조의 list-allowed-node-type-modifications 항목을 참조하십시오.

- 2. 다음 파라미터와 함께 AWS CLI modify-replication-group 명령을 사용하여 현재 복제 그룹을 새 노드 유형으로 확장합니다.
 - --replication-group-id 복제 그룹의 이름입니다.
 - --cache-node-type 이 복제 그룹에 있는 캐시 클러스터의 새로운 대형 노드 유형입니다. 이 값은 1단계의 list-allowed-node-type-modifications 명령에 의해 반환되는 인스턴스 유형 중 하나여야 합니다.
 - --cache-parameter-group-name [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우이 파라미터를 생략할 수 있습니다.
 - --apply-immediately 스케일 업 프로세스가 즉시 적용되도록 합니다. 스케일 업 작업을 다음 유지 관리 기간으로 연기하려면 --no-apply-immediately를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group \
--replication-group-id my-repl-group \
--cache-node-type cache.t2.small \
--cache-parameter-group-name redis32-m3-2xl \
--apply-immediately
```

Windows의 경우:

```
aws elasticache modify-replication-group ^
--replication-group-id my-repl-group ^
--cache-node-type cache.t2.small ^
--cache-parameter-group-name redis32-m3-2xl \
--apply-immediately
```

이 명령의 출력은 다음과 같습니다(JSON 형식).

```
{"ReplicationGroup": {
         "Status": "available",
         "Description": "Some description",
         "NodeGroups": [
                 "Status": "available",
                 "NodeGroupMembers": [
                          "CurrentRole": "primary",
                         "PreferredAvailabilityZone": "us-west-2b",
                          "CacheNodeId": "0001",
                          "ReadEndpoint": {
                             "Port": 6379,
                             "Address": "my-repl-
group-001.8fdx4s.0001.usw2.cache.amazonaws.com"
                          "CacheClusterId": "my-repl-group-001"
                     },
                          "CurrentRole": "replica",
                          "PreferredAvailabilityZone": "us-west-2c",
                          "CacheNodeId": "0001",
```

```
"ReadEndpoint": {
                              "Port": 6379,
                              "Address": "my-repl-
group-002.8fdx4s.0001.usw2.cache.amazonaws.com"
                          "CacheClusterId": "my-repl-group-002"
                     }
                 ٦,
                 "NodeGroupId": "0001",
                 "PrimaryEndpoint": {
                      "Port": 6379,
                      "Address": "my-repl-group.8fdx4s.ng.0001.usw2.cache.amazonaws.com"
                 }
             }
         ٦,
         "ReplicationGroupId": "my-repl-group",
         "SnapshotRetentionLimit": 1,
         "AutomaticFailover": "disabled",
         "SnapshotWindow": "12:00-13:00",
         "SnapshottingClusterId": "my-repl-group-002",
         "MemberClusters": [
             "my-repl-group-001",
             "my-repl-group-002",
         "PendingModifiedValues": {}
     }
}
```

자세한 내용은 AWS CLI 참조의 modify-replication-group 항목을 참조하십시오.

- 3. --apply-immediately 파라미터를 사용한 경우 AWS CLI describe-replication-group 명령을 다음 파라미터와 함께 사용하여 복제 그룹의 상태를 모니터링합니다. 상태가 수정 중에서 사용 가능으로 변경되면 축소된 새 복제 그룹에 쓰기를 시작할 수 있습니다.
 - --replication-group-id 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명 하려면 이 파라미터를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-replication-group \
    --replication-group-id my-replication-group
```

Windows의 경우:

```
aws elasticache describe-replication-groups ^
--replication-group-id my-replication-group
```

자세한 내용은 AWS CLI 참조의 describe-replication-groups을 참조하십시오.

Redis 복제 그룹 축소 (ElastiCache API)

다음 절차는 ElastiCache API를 사용하여 복제 그룹을 현재 노드 유형에서 새롭고 더 작은 노드 유형으로 조정합니다. 이 프로세스 중 상태가 수정 중에서 사용 가능으로 변경될 때까지 애플리케이션과 기본 캐시 클러스터 간의 모든 읽기 및 쓰기가 차단됩니다. 그러나 읽기 전용 복제본 캐시 클러스터에서 읽기는 계속 중단되지 않습니다.

더 작은 노드 유형으로 축소하는 데 걸리는 시간은 노드 유형 및 현재 캐시 클러스터에 있는 데이터의 양에 따라 달라집니다.

Redis 복제 그룹을 축소하려면(ElastiCache API)

- 1. 다음 파라미터와 함께 ElastiCache API ListAllowedNodeTypeModifications 작업을 사용하여 축소할 수 있는 노드 유형을 확인합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=ListAllowedNodeTypeModifications
&ReplicationGroupId=MyReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 ListAllowedNodeTypeModifications을 참조하십시오.

- 2. 다음 파라미터와 함께 ModifyRedplicationGroup ElastiCache API 작업을 사용하여 현재 복제 그룹을 새 노드 유형으로 확장합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다.
 - CacheNodeType 이 복제 그룹에 있는 캐시 클러스터의 새로운 대형 노드 유형입니다. 이 값은 1단 계의 ListAllowedNodeTypeModifications 작업에 의해 반환되는 인스턴스 유형 중 하나여야 합니다.
 - CacheParameterGroupName [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우 이 파라미터를 생략할 수 있습니다.
 - ApplyImmediately 스케일 업 프로세스가 즉시 적용되도록 하려면 true로 설정합니다. 축소 프로세스를 다음 유지 관리 기간으로 연기하려면 ApplyImmediately=false를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/ ?Action=ModifyReplicationGroup &ApplyImmediately=true &CacheNodeType=cache.m3.2xlarge &CacheParameterGroupName=redis32-m3-2xl &ReplicationGroupId=myReplGroup &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20141201T220302Z &Version=2014-12-01 &X-Amz-Algorithm=AWS4-HMAC-SHA256 &X-Amz-Date=20141201T220302Z &X-Amz-SignedHeaders=Host &X-Amz-Expires=20141201T220302Z &X-Amz-Credential=<credential> &X-Amz-Signature=<signature>

자세한 내용은 Amazon ElastiCache API 참조의 ModifyReplicationGroup을 참조하십시오.

- 3. ApplyImmediately=true를 사용한 경우 ElastiCache API DescribeReplicationGroups 작업을 다음 파라미터와 함께 사용하여 복제 그룹의 상태를 모니터링합니다. 상태가 수정 중에서 사용 가능으로 변경되면 축소된 새 복제 그룹에 쓰기를 시작할 수 있습니다.
 - ReplicationGroupId 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명하려면 이 파라미터를 사용하십시오.

자세한 내용은 Amazon ElastiCache API 참조의 DescribeReplicationGroups을 참조하십시오.

읽기 용량 늘리기

읽기 용량을 늘리려면 읽기 전용 복제본(최대 5개)을 Redis 복제 그룹에 추가합니다.

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis 클러스터의 읽기 용량을 조정할 수 있습니다. 자세한 내용은 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본 추가 (p. 176) 단원을 참조하십시오.

읽기 용량 줄이기

읽기 용량을 줄이려면 복제본이 있는 Redis 클러스터(API/CLI에서 복제 그룹이라고 함)에서 하나 이상의 읽기 전용 복제본을 삭제합니다. 클러스터가 자동 장애 조치가 활성화된 다중 AZ인 경우 먼저 자동 장애 조치가 있는 다중 AZ를 비활성화해야 마지막 읽기 전용 복제본을 삭제할 수 없습니다. 자세한 내용은 복제 그룹수정 (p. 165) 단원을 참조하십시오.

자세한 내용은 Redis(클러스터 모드 비활성화됨) 복제 그룹에 대해 읽기 전용 복제본 삭제 (p. 178) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨)에서 클러스터 조정

클러스터에 대한 수요 변화에 따라 Redis(클러스터 모드 활성화됨) 클러스터 내 샤드 수를 변경해 성능을 향상시키거나 비용을 줄이도록 결정할 수 있습니다. 이와 같이 하려면 온라인 수평적 조정을 사용하는 것이 좋은데, 이 방법은 조정 프로세스 중에도 클러스터가 계속해서 요청을 처리하도록 하기 때문입니다.

클러스터를 다시 조정하도록 결정할 수 있는 조건은 다음과 같습니다.

• 메모리 부족:

클러스터의 노드에서 메모리가 부족하면 데이터를 저장 및 요청 처리에 더 많은 리소스를 사용하도록 확장을 결정할 수 있습니다.

FreeableMemory, SwapUsage 및 BytesUseForCache 측정치를 모니터링해 노드에서 메모리가 부족한지 확인할 수 있습니다.

• CPU 또는 네트워크 병목 현상:

클러스터에서 지연 시간/처리량 문제가 발생하면 문제를 해결하기 위해 확장이 필요할 수 있습니다.

CPUUtilization, NetworkBytesIn, NetworkBytesOut, CurrConnections 및 NewConnections 측정치를 모니터링해 지연 시간 및 처리량 수준을 모니터링할 수 있습니다.

• 클러스터가 과도하게 조정됨:

축소와 같은 클러스터에 대한 현재 수요는 성능을 저하시키지 않고 비용을 줄입니다.

다음 FreeableMemory, SwapUseage, BytesUseForCache, CPUUtilization, NetworkBytesIn, NetworkBytesOut, CurrConnections 및 NewConnections 측정치를 사용하여 클러스터의 사용을 모니터링해 안전하게 축소할 수 있는지 확인할 수 있습니다.

조정의 성능 영향

오프라인 프로세스를 사용해 조정하는 경우 프로세스 중 상당 부분에서 클러스터가 오프라인 상태가 되기 때문에 요청을 처리할 수 없습니다. 온라인 방법을 사용해 조정하는 경우, 클러스터가 조정 작업 전체에서 계속해서 요청을 처리할 수 있음에도 불구하고 조정은 컴퓨팅 집약적인 작업이기 때문에 성능 저하가 발생합니다. 저하 정도는 일반적인 CPU 사용률과 데이터에 따라 달라집니다.

Redis(클러스터 모드 활성화됨) 클러스터를 조정하는 방법에는 수평 확장과 수직 확장이라는 두 가지 방법이 있습니다.

• 수평 확장에서는 노드 그룹(샤드)을 추가 또는 제거하여 복제 그룹 내 노드 그룹(샤드) 수를 변경할 수 있습니다. 온라인 리샤딩 프로세스를 통해 클러스터가 들어오는 요청을 계속 처리하는 동안 확장/축소할 수 있습니다.

새 클러스터에서 이전 클러스터에서와 달리 슬롯을 구성합니다. 오프라인 방법에만 해당합니다.

수직 확장 - 노드 유형을 변경하여 클러스터의 크기를 조정합니다. 온라인 수직 확장을 통해 클러스터가 들어오는 요청을 계속 처리하는 동안 확장/축소할 수 있습니다.

클러스터의 크기 및 메모리 용량을 확장하거나 축소하여 줄이는 경우 새 구성에 데이터 및 Redis 오버헤드가 충분한 메모리가 있는지 확인합니다.

자세한 내용은 노드 크기 선택 (p. 67) 단원을 참조하십시오.

목차

- Redis(클러스터 모드 활성화됨)를 위한 오프라인 리샤딩 및 샤드 재분배 (p. 243)
- Redis(클러스터 모드 활성화됨)를 위한 온라인 리샤딩 및 샤드 재분배 (p. 243)
 - 온라인 리샤딩을 사용하여 샤드 추가 (p. 244)
 - 샤드 추가(콘솔) (p. 244)
 - 샤드 추가(AWS CLI) (p. 245)
 - 샤드 추가(ElastiCache API) (p. 246)
 - 온라인 리샤딩을 사용하여 샤드 제거 (p. 247)
 - 샤드 제거(콘솔) (p. 247)
 - 샤드 제거(AWS CLI) (p. 247)
 - 샤드 제거(ElastiCache API) (p. 248)
 - 온라인 샤드 재분배 (p. 249)
 - 온라인 샤드 재분배(콘솔) (p. 249)
 - 온라인 샤드 재분배(AWS CLI) (p. 250)
 - 온라인 샤드 재분배(ElastiCache API) (p. 250)
- 노드 유형 수정하여 온라인 수직 확장 (p. 251)
 - 온라인 확장 (p. 252)
 - Redis 캐시 클러스터 확장(콘솔) (p. 252)
 - Redis 캐시 클러스터 확장 (AWS CLI) (p. 222)
 - Redis 캐시 클러스터 확장 (ElastiCache API) (p. 254)
 - 온라인 축소 (p. 256)
 - Redis 캐시 클러스터 축소 (콘솔) (p. 256)
 - Redis 캐시 클러스터 축소 (AWS CLI) (p. 256)

• Redis 캐시 클러스터 축소(ElastiCache API) (p. 259)

Redis(클러스터 모드 활성화됨)를 위한 오프라인 리샤딩 및 샤드 재분배

오프라인 리샤딩 재구성의 주요 이점은 복제 그룹에서 단순히 샤드를 추가 또는 제거하는 것 이상을 할 수 있다는 점입니다. 오프라인 리샤딩 시 복제 그룹의 샤드 수를 변경하는 것 이외에 다음을 수행할 수 있습니다.

- 복제 그룹의 노드 유형을 변경합니다.
- 복제 그룹의 각 노드에 대한 가용 영역을 지정합니다.
- 최신 엔진 버전으로 업그레이드합니다.
- 각 샤드 내 복제 노드 수를 독립적으로 지정합니다.
- 각 샤드에 대한 키스페이스를 지정합니다.

오프라인 샤드 재구성의 주요 단점은 프로세스의 복원 부분에서 클러스터가 오프라인 상태가 되어 애플리케이션에서 엔드포인트를 업데이트할 때까지 이 상태가 지속된다는 점입니다. 클러스터가 오프라인 상태도 지속되는 기간은 클러스터 내 데이터의 양에 따라 달라집니다.

샤드 Redis(클러스터 모드 활성화됨) 클러스터를 오프라인 상태에서 재구성하려면

- 1. 기존 Redis 클러스터의 수동 백업을 생성합니다. 자세한 내용은 수동 백업 만들기 (p. 190) 단원을 참조하십시오.
- 2. 백업에서 복원해 새 클러스터를 생성합니다. 자세한 내용은 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.
- 3. 애플리케이션에서 엔드포인트를 새 클러스터의 엔드포인트로 업데이트합니다. 자세한 내용은 연결 엔 드포인트 찾기 (p. 180) 단원을 참조하십시오.

Redis(클러스터 모드 활성화됨)를 위한 온라인 리샤딩 및 샤드 재 분배

Redis용 Amazon ElastiCache 버전 3.2.10 이상에서 온라인 리샤딩 및 샤드 재분배를 사용하여 중단 시간 없이 동적으로 Redis(클러스터 모드 활성화됨)용 ElastiCache를 조정할 수 있습니다. 이러한 접근 방식은 조정 또는 재분배 진행 중에도 클러스터에서 계속해서 요청을 처리할 수 있음을 의미합니다.

다음을 수행할 수 있습니다.

• 확장 – Redis(클러스터 모드 활성화됨) 클러스터(복제 그룹)에 샤드(노드 그룹)를 추가해 읽기 및 쓰기 용량을 늘립니다.

복제 그룹에 샤드를 하나 이상 추가하는 경우 각 샤드의 노드 수는 기존의 가장 작은 샤드에 있는 노드 수 와 동일합니다.

- 축소 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 제거해 읽기 및 쓰기 용량을 줄여 비용을 절감 합니다.
- 재분배 Redis(클러스터 모드 활성화됨) 클러스터용 ElastiCache에서 샤드 간에 키스페이스를 이동해 샤드 간에 키스페이스가 가급적 균일하게 분배되도록 합니다.

다음을 수행할 수 없습니다.

- 독립적으로 샤드 구성:
 - 각 샤드 내 노드 수는 독립적으로 지정할 수 없습니다.

• 샤드의 키스페이스는 독립적으로 지정할 수 없습니다. 이렇게 하려면 오프라인 프로세스를 사용해야 합니다.

현재. Redis용 ElastiCache 온라인 리샤딩 및 재분배에는 다음 제한 사항이 적용됩니다.

- 이러한 프로세스를 수행하려면 Redis 엔진 버전 3.2.10 이상이 필요합니다. 엔진 버전 업그레이드에 대한 자세한 내용은 엔진 버전 업그레이드 (p. 45) 단원을 참조하십시오.
- 슬롯 또는 키스페이스와 대용량 항목에 대한 제한 사항이 있습니다.

샤드 내 키에 대용량 항목이 포함되어 있으면 확장 또는 재분배 시 해당 키가 새 샤드로 마이그레이션되지 않습니다. 이 기능으로 인해 불균형 샤드가 발생할 수 있습니다.

샤드 내 키에 대용량 항목(직렬화 후 256MB보다 큰 항목)이 포함되어 있으면 축소 시 해당 샤드는 삭제되지 않습니다. 이 기능으로 인해 일부 샤드가 삭제되지 않을 수 있습니다.

- 확장 시 새 샤드의 노드 수는 기존의 가장 작은 노드 수와 동일합니다.
- 확장 시 기존의 모든 샤드에 공통된 태그는 새 샤드로 복사됩니다.

자세한 내용은 모범 사례: 온라인 클러스터 크기 조정 (p. 410) 단원을 참조하십시오.

AWS Management 콘솔, AWS CLI 및 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터용 ElastiCache를 수평적으로 조정 또는 재분배할 수 있습니다.

온라인 리샤딩을 사용하여 샤드 추가

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터에 샤드를 추가할 수 있습니다. Redis(클러스터 모드 활성화됨) 클러스터에 샤드를 추가하면 기존 샤드의 모든 태그가 새 샤드로 복사됩니다.

주제

- 샤드 추가(콘솔) (p. 244)
- 샤드 추가(AWS CLI) (p. 245)
- 샤드 추가(ElastiCache API) (p. 246)

샤드 추가(콘솔)

AWS Management 콘솔을 사용하여 Redis(클러스터 모드 활성화됨) 클러스터에 샤드를 하나 이상 추가할 수 있습니다. 다음 절차에서는 이러한 프로세스를 설명합니다.

Redis(클러스터 모드 활성화됨) 클러스터에 샤드를 추가하려면

- 1. https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
- 3. 샤드를 추가하고자 하는 Redis(클러스터 모드 활성화됨) 클러스터의 이름 왼쪽에 있는 상자가 아닌 클러스터의 이름을 찾아 선택합니다.

Tip

Redis(클러스터 모드 활성화됨) 클러스터의 샤드 열에는 1보다 크거나 같은 값이 있습니다.

- 4. [Add shard]를 선택합니다.
 - a. [Number of shards to be added]에서 이 클러스터에 추가할 샤드 수를 선택합니다.
 - b. [Availability zone(s)]에서는 [No preference] 또는 [Specify availability zones]을 선택합니다.
 - c. [Specify availability zones]를 선택한 경우 각 샤드의 각 노드에 대해 [Availability Zones] 목록에서 노드의 가용 영역을 선택합니다.

d. [추가]를 선택합니다.

샤드 추가(AWS CLI)

다음 프로세스에서는 AWS CLI를 사용해 샤드를 추가하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 재구성하는 방법을 설명합니다.

modify-replication-group-shard-configuration에 다음 파라미터를 사용합니다.

파라미터

- --apply-immediately 필수 사항. 즉시 시작할 샤드 재구성 작업을 지정합니다.
- --replication-group-id 필수 사항. 샤드 재구성 작업을 수행할 복제 그룹(클러스터)을 지정합니다.
- --node-group-count 필수 사항. 작업 완료 시 존재할 샤드(노드 그룹) 수를 지정합니다. 샤드를 추가 하는 경우 --node-group-count의 값은 현재 샤드 수보다 커야 합니다.

경우에 따라 --resharding-configuration을 사용해 복제 그룹의 각 노드에 대한 가용 영역을 지정할 수 있습니다.

--resharding-configuration – 선택 사항. 복제 그룹 내에 있는 각 샤드의 개별 노드에 대한 기본 가용 영역 목록입니다. 이 파라미터는 --node-group-count의 값이 현재 샤드 수보다 큰 경우에만 사용합니다. 샤드 추가 시 이 파라미터를 사용하지 않으면 Amazon ElastiCache에서는 새 노드에 대해 가용 영역을 선택합니다.

다음 예에서는 Redis(클러스터 모드 활성화됨) 클러스터 my-cluster에서 샤드 4개에 대한 키스페이스를 재구성합니다. 또한 이 예에서는 각 샤드 내 개별 노드에 대한 가용 영역을 지정합니다. 작업이 즉시 시작됩니다.

Example - 샤드 추가

Linux, macOS, or Unix의 경우:

Windows의 경우:

자세한 내용은 AWS CLI 문서의 modify-replication-group-shard-configuration을 참조하십시오.

샤드 추가(ElastiCache API)

ElastiCache API에서 ModifyReplicationGroupShardConfiguration 작업을 사용해 Redis(클러스터 모드 활성화됨) 클러스터 내 샤드를 온라인으로 재구성할 수 있습니다.

ModifyReplicationGroupShardConfiguration에 다음 파라미터를 사용합니다.

파라미터

- ApplyImmediately=true 필수 사항. 즉시 시작할 샤드 재구성 작업을 지정합니다.
- ReplicationGroupId 필수 사항. 샤드 재구성 작업을 수행할 복제 그룹(클러스터)을 지정합니다.
- NodeGroupCount 필수 사항. 작업 완료 시 존재할 샤드(노드 그룹) 수를 지정합니다. 샤드를 추가하는 경우 NodeGroupCount의 값은 현재 샤드 수보다 커야 합니다.

경우에 따라 ReshardingConfiguration을 사용해 복제 그룹의 각 노드에 대한 가용 영역을 지정할 수 있습니다.

• ReshardingConfiguration – 선택 사항. 복제 그룹 내에 있는 각 샤드의 개별 노드에 대한 기본 가용 영역 목록입니다. 이 파라미터는 NodeGroupCount의 값이 현재 샤드 수보다 큰 경우에만 사용합니다. 샤 드 추가 시 이 파라미터를 사용하지 않으면 Amazon ElastiCache에서는 새 노드에 대해 가용 영역을 선택 합니다.

다음 프로세스에서는 ElastiCache API를 사용해 샤드를 추가하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 재구성하는 방법을 설명합니다.

Example - 샤드 추가

east-2a

&Version=2015-02-02

https://elasticache.us-east-2.amazonaws.com/

다음 예에서는 Redis(클러스터 모드 활성화됨) 클러스터 my-cluster에 노드 그룹을 추가해 작업을 완료하면 노드 그룹이 총 4개가 됩니다. 또한 이 예에서는 각 샤드 내 개별 노드에 대한 가용 영역을 지정합니다. 작업이 즉시 시작됩니다.

?Action=ModifyReplicationGroupShardConfiguration & ApplyImmediately=true & NodeGroupCount=4 & ReplicationGroupId=my-cluster

& ReshardingConfiguration.ReshardingConfiguration.1.PreferredAvailabilityZones.AvailabilityZone.1=us-east-2a & ReshardingConfiguration.ReshardingConfiguration.1.PreferredAvailabilityZones.AvailabilityZone.2=us-east-2c & ReshardingConfiguration.ReshardingConfiguration.2.PreferredAvailabilityZones.AvailabilityZone.1=us-east-2b & ReshardingConfiguration.ReshardingConfiguration.2.PreferredAvailabilityZones.AvailabilityZone.2=us-east-2a & ReshardingConfiguration.ReshardingConfiguration.3.PreferredAvailabilityZones.AvailabilityZone.1=us-east-2c & ReshardingConfiguration.ReshardingConfiguration.3.PreferredAvailabilityZones.AvailabilityZone.1=us-east-2d & ReshardingConfiguration.ReshardingConfiguration.3.PreferredAvailabilityZones.AvailabilityZone.2=us-east-2d & ReshardingConfiguration.ReshardingConfiguration.4.PreferredAvailabilityZones.AvailabilityZone.1=us-east-2d

&ReshardingConfiguration.ReshardingConfiguration.4.PreferredAvailabilityZones.AvailabilityZone.2=us-

&SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20171002T192317Z &X-Amz-Credential=<credential>

자세한 내용은 ElastiCache API 참조의 ModifyReplicationGroupShardConfiguration 항목을 참조하십시오.

온라인 리샤딩을 사용하여 샤드 제거

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 제거할 수 있습니다.

주제

- 샤드 제거(콘솔) (p. 247)
- 샤드 제거(AWS CLI) (p. 247)
- 샤드 제거(ElastiCache API) (p. 248)

샤드 제거(콘솔)

다음 프로세스에서는 AWS Management 콘솔를 사용해 샤드를 제거하여 Redis(클러스터 모드 활성화됨) 클 러스터에서 샤드를 재구성하는 방법을 설명합니다.

복제 그룹에서 노드 그룹(샤드)을 제거하기 전에 ElastiCache에서는 모든 데이터가 나머지 샤드에 맞는지 확 인합니다. 데이터가 맞으면 요청된 대로 지정된 샤드가 복제 그룹에서 삭제됩니다. 데이터가 나머지 노드 그 룹에 맞지 않으면 프로세스가 종료되고 복제 그룹은 요청이 작성되기 전과 동일한 노드 그룹 구성으로 남습 니다.

AWS Management 콘솔를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 하나 이상 제거할 수 있습니다. 복제 그룹에서 샤드를 모두 제거할 수는 없습니다. 대신 복제 그룹을 삭제해야 합니다. 자세한 내용은 복제 그룹 삭제 (p. 166) 단원을 참조하십시오. 다음 절차는 샤드를 하나 이상 삭제하는 프로세스를 설명합니다.

Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 제거하려면

- 1. https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
- 3. 샤드를 제거하고자 하는 Redis(클러스터 모드 활성화됨) 클러스터의 이름 왼쪽에 있는 상자가 아닌 클러스터의 이름을 찾아 선택합니다.

Tip

Redis(클러스터 모드 활성화됨) 클러스터의 샤드 열에는 1보다 크거나 같은 값이 있습니다.

- 4. 샤드 목록에서 삭제하고자 하는 각 샤드의 이름 왼쪽에 있는 상자를 선택합니다.
- 5. [Delete shard]를 선택합니다.

샤드 제거(AWS CLI)

다음 프로세스에서는 AWS CLI를 사용해 샤드를 제거하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 재구성하는 방법을 설명합니다.

Important

복제 그룹에서 노드 그룹(샤드)을 제거하기 전에 ElastiCache에서는 모든 데이터가 나머지 샤드에 맞는지 확인합니다. 데이터가 맞으면 요청된 대로 지정된 샤드(--node-groups-to-remove)가 복제 그룹에서 삭제되고 해당 샤드의 키스페이스가 나머지 샤드로 매핑됩니다. 데이터가 나머지 노드 그룹에 맞지 않으면 프로세스가 종료되고 복제 그룹은 요청이 작성되기 전과 동일한 노드 그룹 구성으로 남습니다.

AWS CLI를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 하나 이상 제거할 수 있습니다. 복제 그룹에서 샤드를 모두 제거할 수는 없습니다. 대신 복제 그룹을 삭제해야 합니다. 자세한 내용은 복제 그룹 삭제 (p. 166) 단원을 참조하십시오.

modify-replication-group-shard-configuration에 다음 파라미터를 사용합니다.

파라미터

- --apply-immediately 필수 사항. 즉시 시작할 샤드 재구성 작업을 지정합니다.
- --replication-group-id 필수 사항. 샤드 재구성 작업을 수행할 복제 그룹(클러스터)을 지정합니다.
- --node-group-count 필수 사항. 작업 완료 시 존재할 샤드(노드 그룹) 수를 지정합니다. 샤드를 제거 하는 경우 --node-group-count의 값은 현재 샤드 수보다 작아야 합니다.
- --node-groups-to-remove --node-group-count이 노드 그룹(샤드)의 현재 수보다 작은 경우에 만 필요합니다. 복제 그룹에서 제거할 샤드(노드 그룹) ID 목록입니다.

다음 절차는 샤드를 하나 이상 삭제하는 프로세스를 설명합니다.

Example - 샤드 제거

다음 예에서는 Redis(클러스터 모드 활성화됨) 클러스터 my-cluster에서 노드 그룹 두 개를 제거해 작업을 완료하면 노드 그룹이 총 2개가 됩니다. 제거된 샤드의 키스페이스는 나머지 샤드 간에 균일하게 분배됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group-shard-configuration \
    --replication-group-id my-cluster \
    --node-group-count 2 \
    --node-groups-to-remove "0002" "0003" \
    --apply-immediately
```

Windows의 경우:

```
aws elasticache modify-replication-group-shard-configuration ^
--replication-group-id my-cluster ^
--node-group-count 2 ^
--node-groups-to-remove "0002" "0003" ^
--apply-immediately
```

자세한 내용은 AWS CLI 문서의 modify-replication-group-shard-configuration을 참조하십시오.

샤드 제거(ElastiCache API)

ElastiCache API에서 ModifyReplicationGroupShardConfiguration 작업을 사용해 Redis(클러스터 모드 활성화됨) 클러스터 내 샤드를 온라인으로 재구성할 수 있습니다.

다음 프로세스에서는 ElastiCache API를 사용해 샤드를 제거하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 재구성하는 방법을 설명합니다.

Important

복제 그룹에서 노드 그룹(샤드)을 제거하기 전에 ElastiCache에서는 모든 데이터가 나머지 샤드에 맞는지 확인합니다. 데이터가 맞으면 요청된 대로 지정된 샤드(NodeGroupsToRemove)가 복제 그룹에서 삭제되고 해당 샤드의 키스페이스가 나머지 샤드로 매핑됩니다. 데이터가 나머지 노드 그룹에 맞지 않으면 프로세스가 종료되고 복제 그룹은 요청이 작성되기 전과 동일한 노드 그룹 구성으로 남습니다.

ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 하나 이상 제거할 수 있습니다. 복제 그룹에서 샤드를 모두 제거할 수는 없습니다. 대신 복제 그룹을 삭제해야 합니다. 자세한 내용은 복제 그룹 삭제 (p. 166) 단원을 참조하십시오.

ModifyReplicationGroupShardConfiguration에 다음 파라미터를 사용합니다.

파라미터

- ApplyImmediately=true 필수 사항. 즉시 시작할 샤드 재구성 작업을 지정합니다.
- ReplicationGroupId 필수 사항. 샤드 재구성 작업을 수행할 복제 그룹(클러스터)을 지정합니다.
- NodeGroupCount 필수 사항. 작업 완료 시 존재할 샤드(노드 그룹) 수를 지정합니다. 샤드를 제거하는 경우 NodeGroupCount의 값은 현재 샤드 수보다 작아야 합니다.
- NodeGroupsToRemove --node-group-count이 노드 그룹(샤드)의 현재 수보다 작은 경우에만 필요합니다. 복제 그룹에서 제거할 샤드(노드 그룹) ID 목록입니다.

다음 절차는 샤드를 하나 이상 삭제하는 프로세스를 설명합니다.

Example - 샤드 제거

다음 예에서는 Redis(클러스터 모드 활성화됨) 클러스터 my-cluster에서 노드 그룹 두 개를 제거해 작업을 완료하면 노드 그룹이 총 2개가 됩니다. 제거된 샤드의 키스페이스는 나머지 샤드 간에 균일하게 분배됩니다.

```
https://elasticache.us-east-2.amazonaws.com/
    ?Action=ModifyReplicationGroupShardConfiguration
&ApplyImmediately=true
&NodeGroupCount=2
&ReplicationGroupId=my-cluster
&NodeGroupsToRemove.member.1=0002
&NodeGroupsToRemove.member.2=0003
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20171002T192317Z
&X-Amz-Credential=<credential>
```

자세한 내용은 ElastiCache API 참조의 ModifyReplicationGroupShardConfiguration 항목을 참조하십시오.

온라인 샤드 재분배

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 재분배할 수 있습니다.

주제

- 온라인 샤드 재분배(콘솔) (p. 249)
- 온라인 샤드 재분배(AWS CLI) (p. 250)
- 온라인 샤드 재분배(ElastiCache API) (p. 250)

온라인 샤드 재분배(콘솔)

다음 프로세스에서는 AWS Management 콘솔를 사용해 샤드를 제거하여 Redis(클러스터 모드 활성화됨) 클 러스터에서 샤드를 재분배하는 방법을 설명합니다.

Redis(클러스터 모드 활성화됨) 클러스터에서 샤드 간에 키스페이스를 재분배하려면

1. https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.

- 2. 탐색 창에서 [Redis]를 선택합니다.
- 3. 재분배하고자 하는 Redis(클러스터 모드 활성화됨) 클러스터의 이름 왼쪽에 있는 상자가 아닌 클러스터의 이름을 선택합니다.

Tip

Redis(클러스터 모드 활성화됨) 클러스터의 샤드 열에는 1보다 크거나 같은 값이 있습니다.

- 4. [Rebalance]를 선택합니다.
- 5. 메시지가 나타나면 Rebalance를 선택합니다. 다음과 유사한 메시지가 표시될 수 있습니다. Slots in the replication group are uniformly distributed. Nothing to do. (Service: AmazonElastiCache; Status Code: 400; Error Code: InvalidReplicationGroupState; Request ID: 2246cebd-9721-11e7-8d5b-e1b0f086c8cf). 이러한 메시지가 표시되면 [Cancel]을 선택합니다.

온라인 샤드 재분배(AWS CLI)

modify-replication-group-shard-configuration에 다음 파라미터를 사용합니다.

파라미터

- -apply-immediately 필수 사항. 즉시 시작할 샤드 재구성 작업을 지정합니다.
- --replication-group-id 필수 사항. 샤드 재구성 작업을 수행할 복제 그룹(클러스터)을 지정합니다.
- --node-group-count 필수 사항. 클러스터 내 모드 샤드 간에 키스페이스를 재분배하려면 이 값이 현재 샤드 수와 동일해야 합니다.

다음 프로세스에서는 AWS CLI를 사용해 샤드를 제거하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 재분배하는 방법을 설명합니다.

Example - 클러스터에서 샤드 재분배

다음 예에서는 Redis(클러스터 모드 활성화됨) 클러스터 my-cluster에서 슬롯을 재분배하여 슬롯이 가급적 균일하게 분산되도록 합니다. --node-group-count(4)의 값은 클러스터 내 현재 샤드 수입니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group-shard-configuration \
    --replication-group-id my-cluster \
    --node-group-count 4 \
    --apply-immediately
```

Windows의 경우:

```
aws elasticache modify-replication-group-shard-configuration ^
    --replication-group-id my-cluster ^
    --node-group-count 4 ^
    --apply-immediately
```

자세한 내용은 AWS CLI 문서의 modify-replication-group-shard-configuration을 참조하십시오.

온라인 샤드 재분배(ElastiCache API)

ElastiCache API에서 ModifyReplicationGroupShardConfiguration 작업을 사용해 Redis(클러스터 모드 활성화됨) 클러스터 내 샤드를 온라인으로 재구성할 수 있습니다.

ModifyReplicationGroupShardConfiguration에 다음 파라미터를 사용합니다.

파라미터

- ApplyImmediately=true 필수 사항. 즉시 시작할 샤드 재구성 작업을 지정합니다.
- ReplicationGroupId 필수 사항, 샤드 재구성 작업을 수행할 복제 그룹(클러스터)을 지정합니다.
- NodeGroupCount 필수 사항. 클러스터 내 모드 샤드 간에 키스페이스를 재분배하려면 이 값이 현재 샤드 수와 동일해야 합니다.

다음 프로세스에서는 ElastiCache API를 사용해 샤드를 재분배하여 Redis(클러스터 모드 활성화됨) 클러스터에서 샤드를 재구성하는 방법을 설명합니다.

Example - 클러스터 재분배

다음 예에서는 Redis(클러스터 모드 활성화됨) 클러스터 my-cluster에서 슬롯을 재분배하여 슬롯이 가급적 균일하게 분산되도록 합니다. NodeGroupCount(4)의 값은 클러스터 내 현재 샤드 수입니다.

자세한 내용은 ElastiCache API 참조의 ModifyReplicationGroupShardConfiguration 항목을 참조하십시오.

노드 유형 수정하여 온라인 수직 확장

Redis용 Amazon ElastiCache 버전 3.2.10 이상에서 온라인 수직 확장을 사용하여 중단 시간 없이 동적으로 Redis 클러스터를 조정할 수 있습니다. 이를 통해 Redis 클러스터는 조정 중에도 요청을 처리할 수 있습니다.

다음을 수행할 수 있습니다.

• 확장 – Redis 클러스터의 노드 유형을 더 큰 노드 유형을 사용하도록 조정하여 읽기 및 쓰기 용량을 늘립니다.

ElastiCache에서는 온라인 상태로 요청을 처리하는 동안 클러스터 크기를 동적으로 조정합니다.

 축소 – 더 작은 노드를 사용하도록 노드 유형을 조정하여 읽기 및 쓰기 용량을 줄입니다. 마찬가지로, ElastiCache에서는 온라인 상태로 요청을 처리하는 동안 클러스터 크기를 동적으로 조정합니다. 이 경우 노드를 축소하여 비용을 절감할 수 있습니다.

Note

확장 및 축소 프로세스는 새로 선택한 노드 유형을 사용하여 클러스터를 생성하고 새 노드를 이전 노드와 동기화합니다. 원활한 확장/축소 흐름을 위해 다음을 수행합니다.

• ENI(탄력적 네트워크 인터페이스) 용량이 충분한지 확인합니다. 축소된 경우, 작은 노드에 예상 트래픽을 흡수할 수 있는 충분한 메모리가 있어야 합니다.

메모리 관리 모범 사례에 대해서는 예약된 메모리 관리 (p. 405)를 참조합니다.

- 수직 확장 프로세스는 온라인 상태를 유지하도록 설계되었지만 이전 노드와 새 노드 간의 데이터 동기화에 의존합니다. 데이터 트래픽이 최소 수준일 것으로 예상되는 시간 동안 확장/축소를 시작 하는 것이 좋습니다.
- 가능한 경우 준비 환경에서 조정 중 애플리케이션 동작을 테스트합니다.

목차

- 온라인 확장 (p. 252)
 - Redis 캐시 클러스터 확장(콘솔) (p. 252)
 - Redis 캐시 클러스터 확장 (AWS CLI) (p. 222)
 - Redis 캐시 클러스터 확장 (ElastiCache API) (p. 254)
- 온라인 축소 (p. 256)
 - Redis 캐시 클러스터 축소 (콘솔) (p. 256)
 - Redis 캐시 클러스터 축소 (AWS CLI) (p. 256)
 - Redis 캐시 클러스터 축소(ElastiCache API) (p. 259)

온라인 확장

주제

- Redis 캐시 클러스터 확장(콘솔) (p. 252)
- Redis 캐시 클러스터 확장 (AWS CLI) (p. 222)
- Redis 캐시 클러스터 확장 (ElastiCache API) (p. 254)

Redis 캐시 클러스터 확장(콘솔)

다음 절차에서는 ElastiCache Management Console을 사용하여 Redis 클러스터를 확장하는 방법에 대해 설명합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

Redis 클러스터를 확장하려면 (console)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
- 3. 클러스터 목록에서 클러스터를 선택합니다.
- 4. 수정을 선택합니다.
- 5. [Modify Cluster] 마법사에서 다음을 수행합니다.
 - [Node type] 목록에서 조정할 노드 유형을 선택합니다. 확장하려면, 기존 노드보다 큰 노드 유형을 선택합니다.
- 6. 확장 프로세스를 즉시 수행하려면 즉시 적용 상자를 선택합니다. [Apply immediately] 상자를 선택하지 않으면 이 클러스터의 다음 유지 관리 기간 중 스케일 업 프로세스가 수행됩니다.
- 7. [Modify]를 선택합니다.

이전 단계에서 [Apply immediately]를 선택한 경우 클러스터의 상태가 수정 중으로 변경됩니다. 상태가 사용 가능으로 변경되면 수정이 완료되고 새 클러스터의 사용을 시작할 수 있습니다.

Redis 캐시 클러스터 확장 (AWS CLI)

다음 절차에서는 AWS CLI를 사용하여 Redis 캐시 클러스터를 확장하는 방법에 대해 설명합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

Redis 캐시 클러스터를 확장하려면 (AWS CLI)

1. 다음 파라미터와 함께 AWS CLI list-allowed-node-type-modifications명령을 실행하여 확장할 수 있는 노드 유형을 확인합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache list-allowed-node-type-modifications \
    --replication-group-id my-replication-group-id
```

Windows의 경우:

```
aws elasticache list-allowed-node-type-modifications ^
--replication-group-id my-replication-group-id
```

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
{
     "ScaleUpModifications": [
         "cache.m3.2xlarge",
         "cache.m3.large".
         "cache.m3.xlarge",
         "cache.m4.10xlarge",
         "cache.m4.2xlarge",
         "cache.m4.4xlarge",
         "cache.m4.large",
         "cache.m4.xlarge",
         "cache.r3.2xlarge",
         "cache.r3.4xlarge",
         "cache.r3.8xlarge",
         "cache.r3.large",
         "cache.r3.xlarge"
        "ScaleDownModifications": [
         "cache.t2.micro",
         "cache.t2.small "
         "cache.t2.medium",
         "cache.t1.small "
     ],
}
```

자세한 내용은 AWS CLI 참조의 list-allowed-node-type-modifications 항목을 참조하십시오.

- 2. 다음 파라미터와 함께 AWS CLImodify-replication-group 명령을 사용하여 복제 그룹을 수정하여 새롭고 더 큰 노드 유형으로 확장합니다.
 - --replication-group-id-확장하는 복제 그룹의 이름입니다.
 - --cache-node-type 캐시 클러스터를 확장할 새 노드 유형입니다. 이 값은 1단계 의 list-allowed-node-type-modifications 명령에 의해 반환되는 노드 유형 중 하나여야 합니다.
 - --cache-parameter-group-name [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우이 파라미터를 생략할 수 있습니다.
 - --apply-immediately 스케일 업 프로세스가 즉시 적용되도록 합니다. 스케일 업 프로세스를 클 러스터의 다음 유지 관리 기간으로 연기하려면 --no-apply-immediately 파라미터를 사용하십시 오.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group \
    --replication-group-id my-redis-cluster \
    --cache-node-type cache.m3.xlarge \
    --apply-immediately
```

Windows의 경우:

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
{
 "ReplicationGroup": {
        "Status": "modifying",
        "Description": "my-redis-cluster",
        "NodeGroups": [
                "Status": "modifying",
                "Slots": "0-16383",
                "NodeGroupId": "0001",
                "NodeGroupMembers": [
                        "PreferredAvailabilityZone": "us-east-1f",
                        "CacheNodeId": "0001",
                        "CacheClusterId": "my-redis-cluster-0001-001"
                    },
                        "PreferredAvailabilityZone": "us-east-1d",
                        "CacheNodeId": "0001",
                        "CacheClusterId": "my-redis-cluster-0001-002"
                    }
                ]
            }
        "ConfigurationEndpoint": {
            "Port": 6379,
            "Address": "my-redis-cluster.r7gdfi.clustercfg.use1.cache.amazonaws.com"
        "ClusterEnabled": true,
        "ReplicationGroupId": "my-redis-cluster",
        "SnapshotRetentionLimit": 1,
        "AutomaticFailover": "enabled",
        "SnapshotWindow": "07:30-08:30",
        "MemberClusters": [
            "my-redis-cluster-0001-001",
            "my-redis-cluster-0001-002"
        "CacheNodeType": "cache.m3.xlarge",
        "PendingModifiedValues": {}
    }
}
```

자세한 내용은 AWS CLI 참조의 modify-replication-group 항목을 참조하십시오.

3. --apply-immediately를 사용한 경우 AWS CLI describe-cache-clusters 명령을 다음 파라미터와 함께 사용하여 캐시 클러스터의 상태를 확인합니다. 상태가 사용 가능으로 변경되면 새롭고 더 큰 캐시 클러스터 노드를 사용할 수 있습니다.

Redis 캐시 클러스터 확장 (ElastiCache API)

다음 절차는 ElastiCache API를 사용하여 캐시 클러스터를 현재 노드 유형에서 새롭고 더 큰 노드 유형으로 조정합니다. 이 프로세스 중 상태가 수정 중에서 사용 가능으로 변경될 때까지 애플리케이션과 기본 캐시 클

러스터 간의 모든 읽기 및 쓰기가 차단됩니다. 그러나 읽기 전용 복제본 캐시 클러스터에서 읽기는 계속 중단 되지 않습니다.

대형 노드 유형으로 스케일 업하는 데 걸리는 시간은 노드 유형 및 현재 캐시 클러스터에 있는 데이터의 양에 따라 달라집니다.

Redis 캐시 클러스터를 확장하려면 (ElastiCache API)

- 1. 다음 파라미터와 함께 ElastiCache API ListAllowedNodeTypeModifications 작업을 사용하여 확 장할 수 있는 노드 유형을 확인합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
 ?Action=ListAllowedNodeTypeModifications
 &ReplicationGroupId=MyReplGroup
 &Version=2015-02-02
 &SignatureVersion=4
 &SignatureMethod=HmacSHA256
 &Timestamp=20150202T192317Z
 &X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 ListAllowedNodeTypeModifications을 참조하십시오.

- 2. 다음 파라미터와 함께 ModifyReplicationGroup ElastiCache API 작업을 사용하여 현재 복제 그룹을 새 노드 유형으로 확장합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다.
 - CacheNodeType 이 복제 그룹에 있는 캐시 클러스터의 새로운 대형 노드 유형입니다. 이 값은 1단 계의 ListAllowedNodeTypeModifications 작업에 의해 반환되는 인스턴스 유형 중 하나여야 합니다.
 - CacheParameterGroupName [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우 이 파라미터를 생략할 수 있습니다.
 - ApplyImmediately 스케일 업 프로세스가 즉시 적용되도록 하려면 true로 설정합니다. 스케일 업 프로세스를 다음 유지 관리 기간으로 연기하려면 ApplyImmediately=false를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/ ?Action=ModifyReplicationGroup &ApplyImmediately=true &CacheNodeType=cache.m3.2xlarge &CacheParameterGroupName=redis32-m3-2xl &ReplicationGroupId=myReplGroup &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20141201T220302Z &Version=2014-12-01 &X-Amz-Algorithm=AWS4-HMAC-SHA256 &X-Amz-Date=20141201T220302Z &X-Amz-SignedHeaders=Host &X-Amz-Expires=20141201T220302Z &X-Amz-Credential = < credential > &X-Amz-Signature=<signature>

자세한 내용은 Amazon ElastiCache API 참조의 ModifyReplicationGroup을 참조하십시오.

- 3. ApplyImmediately=true를 사용한 경우 ElastiCache API DescribeReplicationGroups 작업을 다음 파라미터와 함께 사용하여 복제 그룹의 상태를 모니터링합니다. 상태가 수정 중에서 사용 가능으로 변경되면 스케일 업된 새 복제 그룹에 쓰기를 시작할 수 있습니다.
 - ReplicationGroupId 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReplicationGroups
&ReplicationGroupId=MyReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 DescribeReplicationGroups을 참조하십시오.

온라인 축소

주제

- Redis 캐시 클러스터 축소 (콘솔) (p. 256)
- Redis 캐시 클러스터 축소 (AWS CLI) (p. 256)
- Redis 캐시 클러스터 축소(ElastiCache API) (p. 259)

Redis 캐시 클러스터 축소 (콘솔)

다음 절차에서는 ElastiCache Management Console을 사용하여 Redis 클러스터를 축소하는 방법에 대해 설명합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

Redis 클러스터를 축소하려면 (console)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Redis]를 선택합니다.
- 3. 클러스터 목록에서 원하는 클러스터를 선택합니다.
- 4. 수정을 선택합니다.
- 5. [Modify Cluster] 마법사에서 다음을 수행합니다.
 - [Node type] 목록에서 조정할 노드 유형을 선택합니다. 축소하려면, 기존 노드보다 작은 노드 유형을 선택합니다. 모든 노드 유형을 축소할 수 있는 것은 아닙니다.
- 6. 축소 프로세스를 즉시 수행하려면 즉시 적용 상자를 선택합니다. 즉시 적용 상자를 선택하지 않으면 이 클러스터의 다음 유지 관리 기간 중 축소 프로세스가 수행됩니다.
- 7. 수정을 선택합니다.

이전 단계에서 [Apply immediately]를 선택한 경우 클러스터의 상태가 수정 중으로 변경됩니다. 상태가 사용 가능으로 변경되면 수정이 완료되고 새 클러스터의 사용을 시작할 수 있습니다.

Redis 캐시 클러스터 축소 (AWS CLI)

다음 절차에서는 AWS CLI를 사용하여 Redis 캐시 클러스터를 축소하는 방법에 대해 설명합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

Redis 캐시 클러스터를 축소하려면 (AWS CLI)

1. 다음 파라미터와 함께 AWS CLI list-allowed-node-type-modifications 명령을 실행하여 축소할 수 있는 노드 유형을 확인합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache list-allowed-node-type-modifications \
    --replication-group-id my-replication-group-id
```

Windows의 경우:

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
"ScaleUpModifications": [
         "cache.m3.2xlarge",
         "cache.m3.large"
         "cache.m3.xlarge",
         "cache.m4.10xlarge",
         "cache.m4.2xlarge",
         "cache.m4.4xlarge",
         "cache.m4.large",
         "cache.m4.xlarge",
         "cache.r3.2xlarge",
         "cache.r3.4xlarge",
         "cache.r3.8xlarge",
         "cache.r3.large",
         "cache.r3.xlarge"
        "ScaleDownModifications": [
         "cache.t2.micro".
         "cache.t2.small "
         "cache.t2.medium ",
         "cache.t1.small"
     ]
}
```

자세한 내용은 AWS CLI 참조의 list-allowed-node-type-modifications 항목을 참조하십시오.

- 2. 다음 파라미터와 함께 AWS CLImodify-replication-group 명령을 사용하여 복제 그룹을 수정하여 새롭고 더 작은 노드 유형으로 축소합니다.
 - --replication-group-id- 축소하는 복제 그룹의 이름입니다.
 - --cache-node-type 캐시 클러스터를 확장할 새 노드 유형입니다. 이 값은 1단계 의 list-allowed-node-type-modifications 명령에 의해 반환되는 노드 유형 중 하나여야 합니다.
 - --cache-parameter-group-name [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우이 파라미터를 생략할 수 있습니다.
 - --apply-immediately 스케일 업 프로세스가 즉시 적용되도록 합니다. 축소 프로세스를 클러스 터의 다음 유지 관리 기간으로 연기하려면 --no-apply-immediately 파라미터를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-replication-group \
    --replication-group-id my-redis-cluster \
    --cache-node-type cache.t2.micro \
    --apply-immediately
```

Windows의 경우:

위 명령의 출력은 다음과 같습니다(JSON 형식).

```
"ReplicationGroup": {
        "Status": "modifying",
        "Description": "my-redis-cluster",
        "NodeGroups": [
                "Status": "modifying",
                "Slots": "0-16383",
                "NodeGroupId": "0001",
                "NodeGroupMembers": [
                    {
                        "PreferredAvailabilityZone": "us-east-1f",
                        "CacheNodeId": "0001",
                        "CacheClusterId": "my-redis-cluster-0001-001"
                    },
                        "PreferredAvailabilityZone": "us-east-1d",
                        "CacheNodeId": "0001",
                        "CacheClusterId": "my-redis-cluster-0001-002"
                ]
            }
        "ConfigurationEndpoint": {
            "Port": 6379,
            "Address": "my-redis-cluster.r7gdfi.clustercfg.use1.cache.amazonaws.com"
        "ClusterEnabled": true,
        "ReplicationGroupId": "my-redis-cluster",
        "SnapshotRetentionLimit": 1,
        "AutomaticFailover": "enabled",
        "SnapshotWindow": "07:30-08:30",
        "MemberClusters": [
            "my-redis-cluster-0001-001",
            "my-redis-cluster-0001-002"
        "CacheNodeType": "cache.t2.micro",
        "PendingModifiedValues": {}
    }
}
```

자세한 내용은 AWS CLI 참조의 modify-replication-group 항목을 참조하십시오.

3. --apply-immediately를 사용한 경우 AWS CLI describe-cache-clusters 명령을 다음 파라미터와 함께 사용하여 캐시 클러스터의 상태를 확인합니다. 상태가 사용 가능으로 변경되면 새롭고 더 작은 캐시 클러스터를 사용할 수 있습니다.

Redis 캐시 클러스터 축소(ElastiCache API)

다음 절차는 ElastiCache API를 사용하여 복제 그룹을 현재 노드 유형에서 새롭고 더 작은 노드 유형으로 조 정합니다. 이 프로세스 동안 Redis 클러스터는 가동 중지 시간을 최소화하면서 요청을 계속 처리합니다.

더 작은 노드 유형으로 축소하는 데 걸리는 시간은 노드 유형 및 현재 캐시 클러스터에 있는 데이터의 양에 따라 달라집니다.

축소 (ElastiCache API)

- 1. 다음 파라미터와 함께 ElastiCache API ListAllowedNodeTypeModifications 작업을 사용하여 축소할 수 있는 노드 유형을 확인합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다. 모든 복제 그룹 대신 특정 복제 그룹을 설명하려면 이 파라미터를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/
?Action=ListAllowedNodeTypeModifications
&ReplicationGroupId=MyReplGroup
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API 참조의 ListAllowedNodeTypeModifications을 참조하십시오.

- 2. 다음 파라미터와 함께 ModifyReplicationGroup ElastiCache API 작업을 사용하여 현재 복제 그룹을 새 노드 유형으로 축소합니다.
 - ReplicationGroupId 복제 그룹의 이름입니다.
 - CacheNodeType 이 복제 그룹에 있는 캐시 클러스터의 새롭고 더 작은 노드 유형입니다. 이 값은 1 단계의 ListAllowedNodeTypeModifications 작업에 의해 반환되는 인스턴스 유형 중 하나여야 합니다.
 - CacheParameterGroupName [선택 사항] reserved-memory를 사용하여 클러스터의 예약된 메모리를 관리할 경우 이 파라미터를 사용합니다. 새 노드 유형에 대해 올바른 메모리 양을 예약하는 사용자 지정 캐시 파라미터 그룹을 지정합니다. reserved-memory-percent를 사용할 경우 이 파라미터를 생략할 수 있습니다.
 - ApplyImmediately 축소 프로세스가 즉시 적용되도록 하려면 true로 설정합니다. 축소 프로세스 를 다음 유지 관리 기간으로 연기하려면 ApplyImmediately=false를 사용하십시오.

https://elasticache.us-west-2.amazonaws.com/ ?Action=ModifyReplicationGroup &ApplyImmediately=true &CacheNodeType=cache.t2.micro &CacheParameterGroupName=redis32-m3-2x1 &ReplicationGroupId=myReplGroup &SignatureVersion=4 &SignatureMethod=HmacSHA256 &Timestamp=20141201T220302Z &Version=2014-12-01 &X-Amz-Algorithm=AWS4-HMAC-SHA256 &X-Amz-Date=20141201T220302Z &X-Amz-SignedHeaders=Host &X-Amz-Expires=20141201T220302Z &X-Amz-Credential = < credential > &X-Amz-Signature=<signature>

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 파라미터 그룹을 사용해 엔진 파라미터 구성

자세한 내용은 Amazon ElastiCache API 참조의 ModifyReplicationGroup을 참조하십시오.

파라미터 그룹을 사용해 엔진 파라미터 구성

Amazon ElastiCache는 파라미터를 사용하여 노드 및 클러스터의 런타임 속성을 제어합니다. 일반적으로 최신 엔진 버전에는 새로운 기능을 기원하는 추가 파라미터가 포함됩니다. 파라미터가 정리된 표는 Redis 특정파라미터 (p. 274) 단원을 참조하십시오.

max_cache_memory와 같은 일부 파라미터 값은 엔진 및 노드 유형에 의해 결정됩니다. 노드 유형별 파라미터 값의 표는 Redis 노드 유형별 파라미터 (p. 295) 단원을 참조하십시오.

Note

Memcached 특정 파라미터 목록은 Memcached 특정 파라미터를 참조하십시오.

주제

- 파라미터 관리 (p. 260)
- 캐시 파라미터 그룹 티어 (p. 261)
- 파라미터 그룹 생성 (p. 262)
- 이름별로 파라미터 그룹 목록 조회 (p. 264)
- 파라미터 그룹의 값 목록 조회 (p. 269)
- 파라미터 그룹 수정 (p. 271)
- 파라미터 그룹 삭제 (p. 272)
- Redis 특정 파라미터 (p. 274)

파라미터 관리

더욱 쉬운 파라미터 관리를 위해 파라미터를 명명된 파라미터 그룹으로 그룹화합니다. 파라미터 그룹은 시작하는 동안 엔진 소프트웨어에 전달되는 파라미터의 특정 값 조합을 나타냅니다. 이 값은 각 노드의 엔진 프로세서가 런타임에 작동하는 방식을 결정합니다. 특정 파라미터 그룹의 파라미터 값은 해당 파라미터가 속한 클러스터와 상관없이 그룹과 연결된 모든 노드에 적용됩니다.

클러스터 성능을 미세 조정하려면 일부 파라미터 값을 수정하거나 클러스터의 파라미터 그룹을 변경할 수 있 습니다.

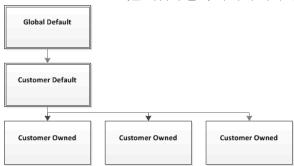
- 기본 파라미터 그룹을 수정하거나 삭제할 수 없습니다. 사용자 지정 파라미터 값이 필요하면 사용자 지정 파라미터 그룹을 생성해야 합니다.
- 파라미터 그룹 패밀리와 할당할 클러스터는 호환 가능해야 합니다. 예를 들어 클러스터에서 Redis 버전 3.2.10을 실행 중이라면 Redis3.2 패밀리의 기본 또는 사용자 지정 파라미터 그룹만 사용할 수 있습니다.
- 클러스터의 파라미터 그룹을 변경하면 조건부로 수정 가능한 파라미터의 값이 현재 및 새 파라미터 그룹에 서 동일해야 합니다.
- 클러스터의 파라미터를 변경하면 변경 사항은 또는 클러스터가 재시작된 즉시 또는 그 이후에 클러스터에 적용됩니다. 이는 클러스터의 파라미터 그룹 자체에서 변경하든 파라미터 값을 클러스터의 파라미터 그룹 내에서 변경하든 마찬가지입니다. 특정 파라미터의 변경 사항이 적용되는 시점을 확인하려면 Redis 특정 파라미터 (p. 274)에 대한 표에서 변경 사항 적용 열을 참조하십시오. 클러스터 재부팅에 관한 자세한 내용은 클러스터 재부팅 (p. 89) 단원을 참조하십시오.
- 파라미터 그룹을 Redis 글로벌 데이터 스토어와 연결할 수 있습니다. 글로벌 데이터 스토어는 AWS 리전에 걸쳐 있는 하나 이상의 클러스터 모음입니다. 이 경우 파라미터 그룹은 글로벌 데이터 스토어를 구성하는 모든 클러스터에서 공유됩니다. 기본 클러스터의 파라미터 그룹에 대한 모든 수정 사항은 글로벌 데이터 스토어의 나머지 모든 클러스터에 복제됩니다. 자세한 내용은 글로벌 데이터 스토어를 사용한 AWS 리전 간 복제 (p. 114) 단원을 참조하십시오.

다음 위치를 보면 파라미터 그룹이 글로벌 데이터 스토어의 일부인지 확인할 수 있습니다.

- 파라미터 그룹 페이지의 ElastiCache 콘솔에서 yes/no 글로벌 속성
- CacheParameterGroup API 작업의 yes/no IsGlobal 속성

캐시 파라미터 그룹 티어

Amazon ElastiCache에는 다음과 같이 세 가지 캐시 파라미터 그룹 티어가 있습니다.



Amazon ElastiCache 파라미터 그룹 티어

전역 기본값

해당 리전의 모든 Amazon ElastiCache 고객을 위한 최상위 루트 파라미터 그룹입니다.

전역 기본 캐시 파라미터 그룹:

• ElastiCache용으로 예약되어 있으며 고객이 사용할 수 없습니다.

고객 기본값

고객의 사용을 위해 생성된 전역 기본 캐시 파라미터 그룹의 사본입니다.

고객 기본 캐시 파라미터 그룹:

- ElastiCache가 생성하고 소유합니다.
- 고객이 이 캐시 파라미터 그룹에서 지원하는 엔진 버전을 실행 중인 모든 클러스터의 캐시 파라미터 그룹으로 사용할 수 있습니다.
- 고객이 편집할 수 없습니다.

고객 소유

고객 기본 캐시 파라미터 그룹의 사본입니다. 고객 소유 캐시 파라미터 그룹은 고객이 캐시 파라미터 그룹을 생성할 때마다 만들어집니다.

고객 소유 캐시 파라미터 그룹:

- 고객이 생성하고 소유합니다.
- 고객의 호환 가능한 모든 클러스터에 할당할 수 있습니다.
- 고객이 사용자 지정 캐시 파라미터 그룹을 생성하기 위해 수정할 수 있습니다.

모든 파라미터 값을 수정할 수 있는 것은 아닙니다. 자세한 내용은 Redis 특정 파라미터 (p. 274) 단원을 참조하십시오.

파라미터 그룹 생성

기본값에서 변경하려는 파라미터 값이 하나 이상이면 새 파라미터 그룹을 생성해야 합니다. ElastiCache 콘 솔, AWS CLI 또는 ElastiCache API를 사용하여 파라미터 그룹을 생성할 수 있습니다.

파라미터 그룹 생성 (콘솔)

다음 절차는 ElastiCache 콘솔을 사용하여 파라미터 그룹을 생성하는 방법을 보여줍니다.

ElastiCache 콘솔을 사용하여 파라미터 그룹을 생성하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 사용 가능한 모든 파라미터 목록을 표시하려면 왼쪽 탐색 창에서 [Parameter Groups]를 선택합니다.
- 3. 파라미터 그룹을 생성하려면 [Create Parameter Group]을 선택합니다.

[Create Parameter Group] 화면이 나타납니다.

4. [Family] 목록에서 파라미터 그룹의 템플릿이 될 파라미터 그룹 패밀리를 선택합니다.

redis3.2 같은 파라미터 그룹 패밀리는 파라미터 그룹의 실제 파라미터와 초기 값을 정의합니다. 파라미터 그룹 패밀리는 클러스터의 엔진 및 버전과 일치해야 합니다.

5. [Name] 상자에 파라미터 그룹의 고유 이름을 입력합니다.

클러스터를 생성하거나 클러스터의 파라미터 그룹을 수정할 때 그 이름으로 파라미터 그룹을 선택합니다. 그러므로 이름은 파라미터 그룹의 패밀리를 식별할 수 있고 정보를 알 수 있는 것이 좋습니다.

파라미터 그룹 명명 제약 조건은 다음과 같습니다.

- ASCII 문자로 시작해야 합니다.
- ASCII 문자, 숫자 및 하이픈만 포함할 수 있습니다.
- 1-255자여야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.
- 6. [Description] 상자에 파라미터 그룹에 대한 설명을 입력합니다.
- 7. 파라미터 그룹을 생성하려면 [Create]를 선택합니다.

파라미터 그룹을 생성하지 않고 프로세스를 종료하려면 [Cancel]을 선택합니다.

8. 파라미터 그룹을 생성하면 패밀리의 기본값이 부여됩니다. 기본값을 변경하려면 파라미터 그룹을 수정 해야 합니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

파라미터 그룹 생성 (AWS CLI)

AWS CLI를 사용하여 파라미터 그룹을 생성하려면 이 파라미터와 함께 create-cache-parameter-group 명령을 사용합니다.

• --cache-parameter-group-name — 파라미터 그룹의 이름입니다.

파라미터 그룹 명명 제약 조건은 다음과 같습니다.

- ASCII 문자로 시작해야 합니다.
- ASCII 문자, 숫자 및 하이픈만 포함할 수 있습니다.
- 1-255자여야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 파라미터 그룹 생성

- --cache-parameter-group-family 파라미터 그룹의 엔진 및 버전 패밀리입니다.
- --description 사용자가 정의한 파라미터 그룹에 대한 설명입니다.

Example

다음 예제에서는 redis2.8 패밀리를 템플릿으로 사용하여 myRed28이라는 파라미터 그룹을 생성합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-cache-parameter-group \
--cache-parameter-group-name myRed28 \
--cache-parameter-group-family redis2.8 \
--description "My first parameter group"
```

Windows의 경우:

```
aws elasticache create-cache-parameter-group ^
--cache-parameter-group-name myRed28 ^
--cache-parameter-group-family redis2.8 ^
--description "My first parameter group"
```

이 명령의 출력은 다음과 유사해야 합니다.

```
"CacheParameterGroup": {
    "CacheParameterGroupName": "myRed28",
    "CacheParameterGroupFamily": "redis2.8",
    "Description": "My first parameter group"
}
}
```

파라미터 그룹을 생성하면 패밀리의 기본값이 부여됩니다. 기본값을 변경하려면 파라미터 그룹을 수정해야 합니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

자세한 내용은 create-cache-parameter-group 단원을 참조하십시오.

파라미터 그룹 생성 (ElastiCache API)

ElastiCache API를 사용하여 파라미터 그룹을 생성하려면 이 파라미터와 함께 CreateCacheParameterGroup 작업을 사용합니다.

• ParameterGroupName — 파라미터 그룹의 이름입니다.

파라미터 그룹 명명 제약 조건은 다음과 같습니다.

- ASCII 문자로 시작해야 합니다.
- ASCII 문자, 숫자 및 하이픈만 포함할 수 있습니다.
- 1-255자여야 합니다.
- 하이픈 2개가 연속될 수 없습니다.
- 끝에 하이픈이 올 수 없습니다.
- CacheParameterGroupFamily 파라미터 그룹의 엔진 및 버전 패밀리입니다. 예: redis2.8.
- Description 사용자가 정의한 파라미터 그룹에 대한 설명입니다.

Example

다음 예제에서는 redis2.8 패밀리를 템플릿으로 사용하여 myRed28이라는 파라미터 그룹을 생성합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 이름별로 파라미터 그룹 목록 조회

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=CreateCacheParameterGroup
    &CacheParameterGroupFamily=redis2.8
    &CacheParameterGroupName=myRed28
    &Description=My%20first%20parameter%20group
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T192317Z
    &Version=2015-02-02
    &X-Amz-Credential=<credential>
```

이 작업의 응답은 다음과 유사해야 합니다.

파라미터 그룹을 생성하면 패밀리의 기본값이 부여됩니다. 기본값을 변경하려면 파라미터 그룹을 수정해야 합니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

자세한 내용은 CreateCacheParameterGroup 단원을 참조하십시오.

이름별로 파라미터 그룹 목록 조회

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 파라미터 그룹을 나열할 수 있습니다.

이름으로 파라미터 그룹 나열 (콘솔)

다음 절차에서는 ElastiCache 콘솔을 사용하여 파라미터 그룹 목록을 보는 방법을 설명합니다.

ElastiCache 콘솔을 사용하여 파라미터 그룹을 나열하려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 사용 가능한 모든 파라미터 목록을 표시하려면 왼쪽 탐색 창에서 [Parameter Groups]를 선택합니다.

이름으로 파라미터 그룹 나열 (AWS CLI)

AWS CLI를 사용하여 파라미터 그룹의 목록을 생성하려면 describe-cache-parameter-groups 명령을 사용합니다. 파라미터 그룹의 이름을 입력하면 그 파라미터 그룹만 나열됩니다. 파라미터 그룹의 이름을입력하지 않으면 최대 --max-records개의 파라미터 그룹이 나열됩니다. 두 경우 모두 파라미터 그룹의 이름, 패밀리 및 설명이 나열됩니다.

Example

다음 샘플 코드에는 myRed28 파라미터 그룹이 나열되어 있습니다.

Linux, macOS, or Unix의 경우:

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 이름별로 파라미터 그룹 목록 조회

```
aws elasticache describe-cache-parameter-groups \
--cache-parameter-group-name myRed28
```

Windows의 경우:

```
aws elasticache describe-cache-parameter-groups ^
--cache-parameter-group-name myRed28
```

이 명령의 출력은 파라미터 그룹의 이름, 패밀리 및 설명을 나열하는 것과 같습니다.

Example

다음 샘플 코드에는 Redis 엔진 버전 5.0.6 이상에서 실행되는 파라미터 그룹에 대한 파라미터 그룹 myRed56이 나열됩니다. 파라미터 그룹이 글로벌 데이터 스토어를 사용한 AWS 리전 간 복제 (p. 114)의 일부인 경우 출력에서 반환되는 IsGlobal 속성 값은 Yes가 됩니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-cache-parameter-groups \
--cache-parameter-group-name myRed56
```

Windows의 경우:

```
aws elasticache describe-cache-parameter-groups ^
--cache-parameter-group-name myRed56
```

이 명령의 출력은 파라미터 그룹의 이름, 패밀리, isGlobal 및 설명을 나열하는 것과 같습니다.

Example

다음 샘플 코드는 최대 10개의 파라미터 그룹을 나열합니다.

```
aws elasticache describe-cache-parameter-groups --max-records 20
```

이 명령의 JSON 출력은 각 파라미터 그룹의 이름, 패밀리, 설명 및 redis5.6의 경우 파라미터 그룹이 글로벌데이터 스토어(IsGlobal)의 일부인지 여부를 나열하는 것과 같습니다.

```
{
    "CacheParameterGroups": [
        {
            "CacheParameterGroupName": "custom-redis32",
            "CacheParameterGroupFamily": "redis3.2",
            "Description": "custom parameter group with reserved-memory > 0"
        },
            "CacheParameterGroupName": "default.memcached1.4",
            "CacheParameterGroupFamily": "memcached1.4",
            "Description": "Default parameter group for memcached1.4"
        },
            "CacheParameterGroupName": "default.redis2.6",
            "CacheParameterGroupFamily": "redis2.6",
            "Description": "Default parameter group for redis2.6"
        },
            "CacheParameterGroupName": "default.redis2.8",
            "CacheParameterGroupFamily": "redis2.8",
            "Description": "Default parameter group for redis2.8"
        },
            "CacheParameterGroupName": "default.redis3.2",
            "CacheParameterGroupFamily": "redis3.2",
            "Description": "Default parameter group for redis3.2"
        },
            "CacheParameterGroupName": "default.redis3.2.cluster.on",
            "CacheParameterGroupFamily": "redis3.2",
            "Description": "Customized default parameter group for redis3.2 with cluster
mode on"
       },
        {
            "CacheParameterGroupName": "default.redis5.6.cluster.on",
            "CacheParameterGroupFamily": "redis5.0",
            "Description": "Customized default parameter group for redis5.6 with cluster
mode on",
            "isGlobal": "yes"
        },
    ]
}
```

자세한 내용은 describe-cache-parameter-groups 단원을 참조하십시오.

이름으로 파라미터 그룹 나열 (ElastiCache API)

ElastiCache API를 사용하여 파라미터 그룹의 목록을 생성하려면 DescribeCacheParameterGroups 작업을 사용합니다. 파라미터 그룹의 이름을 입력하면 그 파라미터 그룹만 나열됩니다. 파라미터 그룹의 이름을 입력하지 않으면 최대 MaxRecords개의 파라미터 그룹이 나열됩니다. 두 경우 모두 파라미터 그룹의 이름, 패밀리 및 설명이 나열됩니다.

Example

다음 샘플 코드는 최대 10개의 파라미터 그룹을 나열합니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&MaxRecords=10
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
```

&Version=2015-02-02 &X-Amz-Credential=<credential>

이 작업의 응답은 각 파라미터 그룹의 이름, 패밀리, 설명 및 redis5.6의 경우 파라미터 그룹이 글로벌 데이터 스토어(IsGlobal)의 일부인지 여부를 나열하는 것과 같습니다.

```
<DescribeCacheParameterGroupsResponse xmlns="http://elasticache.amazonaws.com/</pre>
doc/2013-06-15/">
 <DescribeCacheParameterGroupsResult>
   <CacheParameterGroups>
      <CacheParameterGroup>
       <CacheParameterGroupName>myRedis28</CacheParameterGroupName>
        <CacheParameterGroupFamily>redis2.8</CacheParameterGroupFamily>
        <Description>My custom Redis 2.8 parameter group/Description>
      </CacheParameterGroup>
      <CacheParameterGroup>
        <CacheParameterGroupName>myMem14</CacheParameterGroupName>
        <CacheParameterGroupFamily>memcached1.4</CacheParameterGroupFamily>
        <Description>My custom Memcached 1.4 parameter group/Description>
      </CacheParameterGroup>
       <CacheParameterGroup>
       <CacheParameterGroupName>myRedis56</CacheParameterGroupName>
       <CacheParameterGroupFamily>redis5.0</CacheParameterGroupFamily>
        <Description>My custom redis 5.6 parameter group/Description>
        <isGlobal>yes</isGlobal>
      </CacheParameterGroup>
   </CacheParameterGroups>
 </DescribeCacheParameterGroupsResult>
  <ResponseMetadata>
    <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e</RequestId>
 </ResponseMetadata>
</DescribeCacheParameterGroupsResponse>
```

Example

다음 샘플 코드에는 myRed28 파라미터 그룹이 나열되어 있습니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&CacheParameterGroupName=myRed28
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential<
```

이 작업의 응답은 각 파라미터 그룹의 이름. 패밀리 및 설명을 나열하는 것과 같습니다.

</DescribeCacheParameterGroupsResponse>

Example

다음 샘플 코드에는 mvRed56 파라미터 그룹이 나열되어 있습니다.

이 작업의 응답은 각 파라미터 그룹의 이름, 패밀리, 설명 및 파라미터 그룹이 글로벌 데이터 스토어 (IsGlobal)의 일부인지 여부를 나열하는 것과 같습니다.

```
<DescribeCacheParameterGroupsResponse xmlns="http://elasticache.amazonaws.com/</pre>
doc/2013-06-15/">
 <DescribeCacheParameterGroupsResult>
    <CacheParameterGroups>
      <CacheParameterGroup>
        <CacheParameterGroupName>myRed56</CacheParameterGroupName>
        <CacheParameterGroupFamily>redis5.0</CacheParameterGroupFamily>
        <Description>My custom Redis 5.6 parameter group/Description>
        <isGlobal>yes</isGlobal>
      </CacheParameterGroup>
    </CacheParameterGroups>
 </DescribeCacheParameterGroupsResult>
  <ResponseMetadata>
    <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e/RequestId>
  </ResponseMetadata>
</DescribeCacheParameterGroupsResponse>
```

Example

다음 샘플 코드는 최대 10개의 파라미터 그룹을 나열합니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&MaxRecords=10
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

이 작업의 응답은 각 파라미터 그룹의 이름, 패밀리, 설명 및 redis5.6의 경우 파라미터 그룹이 글로벌 데이터 스토어(IsGlobal)의 일부인지 여부를 나열하는 것과 같습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 파라미터 그룹의 값 목록 조회

```
<CacheParameterGroupName>myMem14</CacheParameterGroupName>
       <CacheParameterGroupFamily>memcached1.4</CacheParameterGroupFamily>
       <Description>My custom Memcached 1.4 parameter group/Description>
      </CacheParameterGroup>
   </CacheParameterGroups>
     <CacheParameterGroup>
        <CacheParameterGroupName>myRedis56</CacheParameterGroupName>
       <CacheParameterGroupFamily>redis5.0</CacheParameterGroupFamily>
       <Description>My custom Redis 5.6 parameter group/Description>
       <isGlobal>yes</isGlobal>
      </CacheParameterGroup>
 </DescribeCacheParameterGroupsResult>
 <ResponseMetadata>
   <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e/RequestId>
 </ResponseMetadata>
</DescribeCacheParameterGroupsResponse>
```

자세한 내용은 DescribeCacheParameterGroups 단원을 참조하십시오.

파라미터 그룹의 값 목록 조회

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 파라미터 및 파라미터 그룹의 값을 나열할 수 있습니다.

파라미터 그룹 값 나열 (콘솔)

다음 절차에서는 ElastiCache 콘솔을 사용하여 파라미터 및 파라미터 그룹의 값을 나열하는 방법을 보여줍니다.

ElastiCache 콘솔을 사용하여 파라미터 그룹의 파라미터 및 그 값을 나열하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 사용 가능한 모든 파라미터 목록을 표시하려면 왼쪽 탐색 창에서 [Parameter Groups]를 선택합니다.
- 3. 파라미터 그룹의 이름 왼쪽에 있는 확인란을 선택하여 파라미터 및 값을 나열할 파라미터 그룹을 선택합니다.

파라미터 및 그 값이 화면 하단에 나열됩니다. 파라미터의 수가 많으면 위아래로 스크롤하여 원하는 파라미터를 찾습니다.

파라미터 그룹 값 나열 (AWS CLI)

AWS CLI를 사용하여 파라미터 그룹의 파라미터 및 그 값을 나열하려면 describe-cache-parameters 명령을 사용합니다.

Example

다음 샘플 코드는 myRedis28 파라미터 그룹의 모든 파라미터 및 그 값을 나열합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache describe-cache-parameters \
--cache-parameter-group-name myRedis28
```

Windows의 경우:

aws elasticache describe-cache-parameters ^

--cache-parameter-group-name myRed28

자세한 내용은 describe-cache-parameters 단원을 참조하십시오.

파라미터 그룹 값 나열 (ElastiCache API)

ElastiCache API를 사용하여 파라미터 그룹의 파라미터 및 그 값을 나열하려면 DescribeCacheParameters 작업을 사용합니다.

Example

다음 샘플 코드에는 myRed28 파라미터 그룹의 모든 파라미터가 나열되어 있습니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=DescribeCacheParameters
    &CacheParameterGroupName=myRed28
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T192317Z
    &Version=2015-02-02
    &X-Amz-Credential < credential >
```

이 작업의 응답은 다음과 유사합니다. 응답이 잘렸습니다.

```
<DescribeCacheParametersResponse xmlns="http://elasticache.amazonaws.com/doc/2013-06-15/">
 <DescribeCacheParametersResult>
   <CacheClusterClassSpecificParameters>
      <CacheNodeTypeSpecificParameter>
        <DataType>integer</DataType>
        <Source>system</Source>
        <IsModifiable>false</IsModifiable>
        <Description>The maximum configurable amount of memory to use to store items, in
megabytes.</Description>
        <CacheNodeTypeSpecificValues>
          <CacheNodeTypeSpecificValue>
            <Value>1000</Value>
            <CacheClusterClass>cache.cl.medium</CacheClusterClass>
          </CacheNodeTypeSpecificValue>
          <CacheNodeTypeSpecificValue>
            <Value>6000</Value>
            <CacheClusterClass>cache.c1.xlarge</CacheClusterClass>
          </CacheNodeTypeSpecificValue>
          <CacheNodeTypeSpecificValue>
            <Value>7100</Value>
            <CacheClusterClass>cache.ml.large</CacheClusterClass>
          </CacheNodeTypeSpecificValue>
          <CacheNodeTypeSpecificValue>
            <Value>1300</Value>
            <CacheClusterClass>cache.ml.small</CacheClusterClass>
          </CacheNodeTypeSpecificValue>
...output omitted...
   </CacheClusterClassSpecificParameters>
 </DescribeCacheParametersResult>
 <ResponseMetadata>
   <RequestId>6d355589-af49-11e0-97f9-279771c4477e</RequestId>
  </ResponseMetadata>
</DescribeCacheParametersResponse>
```

자세한 내용은 DescribeCacheParameters 단원을 참조하십시오.

파라미터 그룹 수정

Important

어떤 기본 파라미터 그룹도 수정할 수 없습니다.

파라미터 그룹의 일부 파라미터 값을 수정할 수 있습니다. 이 파라미터 값은 파라미터 그룹과 연결된 클러스터에 적용됩니다. 변경한 파라미터 값이 파라미터 그룹에 적용되는 시점에 관한 자세한 내용은 Redis 특정파라미터 (p. 274) 단원을 참조하십시오.

파라미터 그룹 수정 (콘솔)

다음 절차는 ElastiCache 콘솔을 사용하여 binding_protocol 파라미터 값을 변경하는 방법을 보여줍니다. 동일한 절차를 통해 모든 파라미터 값을 변경할 수 있습니다.

ElastiCache 콘솔을 사용하여 파라미터 값을 변경하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 사용 가능한 모든 파라미터 목록을 표시하려면 왼쪽 탐색 창에서 [Parameter Groups]를 선택합니다.
- 3. 파라미터 그룹의 이름 왼쪽에 있는 확인란을 선택하여 수정할 파라미터 그룹을 선택합니다.
 - 파라미터 그룹의 파라미터가 화면 하단에 나열됩니다. 모든 파라미터를 보려면 목록 페이지를 탐색해야할 수도 있습니다.
- 4. 파라미터를 하나 이상 수정하려면 [Edit Parameters]를 선택합니다.
- 5. [Edit Parameter Group:] 화면에서 왼쪽/오른쪽 화살표로 스크롤하여 binding_protocol 파라미터를 찾은 다음 [Value] 열에 [ascii]를 입력합니다.
- 6. [Save Changes]를 선택합니다.
- 7. 변경한 파라미터의 이름을 찾으려면 Redis 특정 파라미터 (p. 274) 단원을 참조하십시오. 다시 시작한 후 파라미터에 변경 사항이 발생하면 이 파라미터 그룹을 사용하는 모든 클러스터를 재부팅합니다. 자세한 내용은 클러스터 재부팅 (p. 89) 단원을 참조하십시오.

파라미터 그룹 수정 (AWS CLI)

AWS CLI를 사용하여 파라미터 값을 변경하려면 modify-cache-parameter-group 명령을 사용합니다.

Example

변경하려는 파라미터의 이름과 허용되는 값을 찾으려면 Redis 특정 파라미터 (p. 274) 단원을 참조하십시오.

다음 샘플 코드는 파라미터 그룹 myredis32-on-30에 대해 두 파라미터 reserved-memory-percent 및 cluster-enabled의 값을 설정합니다. reserved-memory-percent를 30(30%)로, cluster-enabled를 yes로 설정해 Redis(클러스터 모드 활성화됨) 클러스터(복제 그룹)에 해당 파라미터 그룹을 사용할 수 있도록 합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-parameter-group \
    --cache-parameter-group-name myredis32-on-30 \
    --parameter-name-values \
        ParameterName=reserved-memory-percent, ParameterValue=30 \
        ParameterName=cluster-enabled, ParameterValue=yes
```

Windows의 경우:

aws elasticache modify-cache-parameter-group ^

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 파라미터 그룹 삭제

```
--cache-parameter-group-name myredis32-on-30 ^
--parameter-name-values ^
ParameterName=reserved-memory-percent, ParameterValue=30 ^
ParameterName=cluster-enabled, ParameterValue=yes
```

이 명령의 출력은 다음과 같습니다.

```
{
    "CacheParameterGroupName": "my-redis32-on-30"
}
```

자세한 내용은 modify-cache-parameter-group 단원을 참조하십시오.

다시 시작한 후 파라미터에 변경 사항이 발생하면 이 파라미터 그룹을 사용하는 모든 클러스터를 재부팅합니다. 자세한 내용은 클러스터 재부팅 (p. 89) 단원을 참조하십시오.

파라미터 그룹 수정 (ElastiCache API)

ElastiCache API를 사용하여 파라미터 그룹의 파라미터 값을 변경하려면 ModifyCacheParameterGroup 작업을 사용합니다.

Example

변경하려는 파라미터의 이름과 허용되는 값을 찾으려면 Redis 특정 파라미터 (p. 274) 단원을 참조하십시 오.

다음 샘플 코드는 파라미터 그룹 myredis32-on-30에 대해 두 파라미터 reserved-memory-percent 및 cluster-enabled의 값을 설정합니다. reserved-memory-percent를 30(30%)로, cluster-enabled를 yes로 설정해 Redis(클러스터 모드 활성화됨) 클러스터(복제 그룹)에 해당 파라미터 그룹을 사용할 수 있도록 합니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=ModifyCacheParameterGroup
&CacheParameterGroupName=myredis32-on-30
&ParameterNameValues.member.1.ParameterName=reserved-memory-percent
&ParameterNameValues.member.1.ParameterValue=30
&ParameterNameValues.member.2.ParameterName=cluster-enabled
&ParameterNameValues.member.2.ParameterValue=yes
&SignatureVersion=4
&SignatureWersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

자세한 내용은 ModifyCacheParameterGroup 단원을 참조하십시오.

파라미터를 업데이트하고 저장한 후에 파라미터에 변경 사항이 발생하면 다시 시작한 후에 이 파라미터 그룹을 사용하는 각 클러스터를 재부팅합니다. 자세한 내용은 클러스터 재부팅 (p. 89) 단원을 참조하십시오.

파라미터 그룹 삭제

ElastiCache 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 사용자 지정 파라미터 그룹을 삭제할 수 있습니다.

파라미터 그룹이 클러스터와 연결된 경우 삭제할 수 없습니다. 또한 기본 파라미터 그룹도 삭제할 수 없습니 다

파라미터 그룹 삭제 (콘솔)

다음 절차는 ElastiCache 콘솔을 사용하여 파라미터 그룹을 삭제하는 방법을 보여줍니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 파라미터 그룹 삭제

ElastiCache 콘솔을 사용하여 파라미터 그룹을 삭제하려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 사용 가능한 모든 파라미터 목록을 표시하려면 왼쪽 탐색 창에서 [Parameter Groups]를 선택합니다.
- 파라미터 그룹의 이름 왼쪽에 있는 확인란을 선택하여 삭제할 파라미터 그룹을 선택합니다.

[Delete] 버튼이 활성화됩니다.

4. 삭제를 선택합니다.

[Delete Parameter Groups] 확인 화면이 나타납니다.

5. 파라미터 그룹을 삭제하려면 [Delete Parameter Groups] 확인 화면에서 [Delete]를 추가합니다.

파라미터 그룹을 유지하려면 [Cancel]을 선택합니다.

파라미터 그룹 삭제 (AWS CLI)

delete-cache-parameter-group를 사용하여 파라미터 그룹을 삭제하려면 AWS CLI 명령을 사용합니다. 삭제할 파라미터 그룹의 경우 --cache-parameter-group-name으로 지정된 파라미터 그룹에는 클러스터를 연결할 수 없으며 기본 파라미터 그룹이 될 수도 없습니다.

다음 샘플 코드에서는 myMem14 파라미터 그룹을 삭제합니다.

Example

Linux, macOS, or Unix의 경우:

```
aws elasticache delete-cache-parameter-group \
--cache-parameter-group-name myRed28
```

Windows의 경우:

```
aws elasticache delete-cache-parameter-group ^
--cache-parameter-group-name myRed28
```

자세한 내용은 delete-cache-parameter-group 단원을 참조하십시오.

파라미터 그룹 삭제 (ElastiCache API)

ElastiCache API를 사용하여 파라미터 그룹을 삭제하려면 DeleteCacheParameterGroup 작업을 사용합니다. 삭제할 파라미터 그룹의 경우 CacheParameterGroupName으로 지정된 파라미터 그룹에는 클러스터를 연결할 수 없으며 기본 파라미터 그룹이 될 수도 없습니다.

Example

다음 샘플 코드에서는 myRed28 파라미터 그룹을 삭제합니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DeleteCacheParameterGroup
&CacheParameterGroupName=myRed28
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
```

&X-Amz-Credential=<credential>

자세한 내용은 DeleteCacheParameterGroup 단원을 참조하십시오.

Redis 특정 파라미터

Redis 클러스터에 파라미터 그룹을 지정하지 않으면 엔진 버전에 적절한 기본 파라미터 그룹이 사용됩니다. 기본 파라미터 그룹에서는 어떤 파라미터 값도 변경할 수 없습니다. 그러나 조건부로 수정 가능한 파라미터 값이 두 파라미터 그룹에서 동일하면 언제든지 사용자 지정 파라미터 그룹을 생성하고 클러스터에 할당할 수 있습니다. 자세한 내용은 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

주제

- Redis 5.0.3 파라미터 변경 사항 (p. 275)
- Redis 5.0.0 파라미터 변경 사항 (p. 276)
- Redis 4.0.10 파라미터 변경 사항 (p. 279)
- Redis 3.2.10 파라미터 변경 사항 (p. 282)
- Redis 3.2.6 파라미터 변경 사항 (p. 282)
- Redis 3.2.4 파라미터 변경 사항 (p. 282)
- Redis 2.8.24(개선 됨) 추가 파라미터 (p. 285)
- Redis 2.8.23(개선 됨) 추가 파라미터 (p. 285)
- Redis 2.8.22(개선됨) 추가 파라미터 (p. 287)
- Redis 2.8.21 추가 파라미터 (p. 287)
- Redis 2.8.19 추가 파라미터 (p. 287)
- Redis 2.8.6 추가 파라미터 (p. 287)
- Redis 2.6.13 파라미터 (p. 290)
- Redis 노드 유형별 파라미터 (p. 295)

Note

새로운 Redis 버전은 보다 향상되고 안정적인 사용자 환경을 제공하므로 ElastiCache 콘솔을 사용할 경우 Redis 버전 2.6.13, 2.8.6 및 2.8.19는 더 이상 지원되지 않습니다. 이러한 Redis 버전은 사용하지 않을 것을 권장합니다. 이러한 버전 중 하나를 사용해야 하는 경우 AWS CLI 또는 ElastiCache API에서 작업하십시오.

자세한 정보는 다음 항목을 참조하십시오.

	AWS CLI	ElastiCache API
클러스터 생성	클러스터 생성 (AWS CLI) (p. 76)	클러스터 생성 (ElastiCache API) (p. 77)
	클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.	클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.
클러스터 수정	AWS CLI 사용 (p. 88) 클러스터 모드가 활성화된 상 태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니 다.	ElastiCache API 사용 (p. 89) 클러스터 모드가 활성화된 상태에서는 이 작업을 사용하여 복제 그룹을 생성할 수 없습니다.

	AWS CLI	ElastiCache API
복제 그룹 생성	Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 149) Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154)	Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성 (p. 151) Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158)
복제 그룹 수정	AWS CLI 사용 (p. 166)	ElastiCache API 사용 (p. 166)

Redis 5.0.3 파라미터 변경 사항

파라미터 그룹 Family: redis5.0

Redis 5.0 기본 파라미터 그룹

- default.redis5.0 이 파라미터 그룹 또는 Redis(클러스터 모드 비활성화됨) 클러스터 및 복제 그룹 으로부터 파생된 파라미터 그룹을 사용합니다.
- default.redis5.0.cluster.on 이 파라미터 그룹 또는 Redis(클러스터 모드 활성화됨) 클러스터 및 복제 그룹으로부터 파생된 파라미터 그룹을 사용합니다.

Redis 5.0.3에 추가된 파라미터

이름	세부 정보	설명
rename-	기본값: 없음	이름 변경된 Redis 명령의 목록(공백으로 구분)입니
commands	형식: 문자열	다. 다음은 이름 변경에 사용할 수 있는 제한된 명령 의 목록입니다.
	수정 가능 여부: 예	APPEND AUTH BITCOUNT BITFIELD BITOP
	변경 적용: 클러스터의 모든 노드에 즉시 적용됨	BITPOS BLPOP BRPOP BRPOPLPUSH BZPOPMIN BZPOPMAX CLIENT CLUSTER COMMAND DBSIZE DECR DECRBY DEL DISCARD DUMP ECHO EVAL EVALSHA EXEC EXISTS EXPIRE EXPIREAT FLUSHALL FLUSHDB GEOADD GEOHASH GEOPOS GEODIST GEORADIUS GEORADIUSBYMEMBER GET GETBIT GETRANGE GETSET HDEL HEXISTS HGET HGETALL HINCRBY HINCRBYFLOAT HKEYS HLEN HMGET HMSET HSET HSETNX HSTRLEN HVALS INCR INCRBY INCRBYFLOAT INFO KEYS LASTSAVE LINDEX LINSERT LLEN LPOP LPUSH LPUSHX LRANGE LREM LSET LTRIM MEMORY MGET MONITOR MOVE MSET MSETNX MULTI OBJECT PERSIST PEXPIRE PEXPIREAT PFADD PFCOUNT PFMERGE PING PSETEX PSUBSCRIBE PUBSUB PTTL PUBLISH PUNSUBSCRIBE RANDOMKEY READONLY READWRITE RENAME RENAMENX RESTORE ROLE RPOP RPOPLPUSH RPUSH RPUSHX SADD SCARD SCRIPT SDIFF SDIFFSTORE SELECT SET SETBIT SETEX

이름	세부 정보	설명
		SETNX SETRANGE SINTER SINTERSTORE SISMEMBER SLOWLOG SMEMBERS SMOVE SORT SPOP SRANDMEMBER SREM STRLEN SUBSCRIBE SUNION SUNIONSTORE SWAPDB TIME TOUCH TTL TYPE UNSUBSCRIBE UNLINK UNWATCH WAIT WATCH ZADD ZCARD ZCOUNT ZINCRBY ZINTERSTORE ZLEXCOUNT ZPOPMAX ZPOPMIN ZRANGE ZRANGEBYLEX ZREVRANGEBYLEX ZRANGEBYSCORE ZRANK ZREM ZREMRANGEBYLEX ZREMRANGEBYRANK ZREMRANGEBYSCORE ZREVRANGE ZREVRANGEBYSCORE ZREVRANK ZSCORE ZUNIONSTORE SCAN SSCAN HSCAN ZSCAN XINFO XADD XTRIM XDEL XRANGE XREVRANGE XLEN XREAD XGROUP XREADGROUP XACK XCLAIM XPENDING GEORADIUS_RO GEORADIUSBYMEMBER_RO LOLWUT XSETID SUBSTR

자세한 내용은 Redis용 ElastiCache 버전 5.0.3(개선됨) (p. 39) 단원을 참조하십시오.

Redis 5.0.0 파라미터 변경 사항

파라미터 그룹 Family: redis5.0

Redis 5.0 기본 파라미터 그룹

- default.redis5.0 이 파라미터 그룹 또는 Redis(클러스터 모드 비활성화됨) 클러스터 및 복제 그룹 으로부터 파생된 파라미터 그룹을 사용합니다.
- default.redis5.0.cluster.on 이 파라미터 그룹 또는 Redis(클러스터 모드 활성화됨) 클러스터 및 복제 그룹으로부터 파생된 파라미터 그룹을 사용합니다.

Redis 5.0에서 추가된 파라미터

이름	세부 정보	설명
stream- node-max- bytes	허용되는 값: 0+ 기본값: 4096 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	스트림 데이터 구조는 내부에 여러 항목을 인코딩하는 노드의 기수 트리입니다. 이 구성을 사용하여 기수 트리에서 단일 노드의 최대 크기를 바이트로 지정합니다. 0으로 설정하면 트리 노드의 크기는 무제한입니다.
stream- node-max- entries	허용되는 값: 0+ 기본값: 100 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	스트림 데이터 구조는 내부에 여러 항목을 인코딩하는 노드의 기수 트리입니다. 이 구성을 사용하여 새노드 항목을 추가할 때 새 노드로 전환하기 전에 단일 노드에 포함할 수 있는 최대 항목 수를 지정합니다. 0으로 설정하면 트리 노드의 항목 수는 무제한입니다.

이름	세부 정보	설명
active-	허용되는 값: 1-1000000	기본 사전 스캔에서 처리될 최대 set/hash/zset/list 필드 수
defrag-max- scan-fields	기본값: 1000	글느 ㅜ
	유형: 정수	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
lua-	허용되는 값: yes/no	Lua 스크립트에서 항상 Lua 효과 복제를 활성화하
replicate- commands	기본값: yes	거나 활성화하지 않음
	유형: 부울	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
replica-	기본값: yes	복제본이 마스터와 독립적인 항목을 유지하여
ignore- maxmemory	유형: 부울	maxmemory 설정을 무시하는지 결정합니다.
	수정 가능 여부: 아니요	

Redis는 커뮤니티 피드백에 따라 엔진 버전 5.0에서 여러 파라미터의 이름을 변경했습니다. 자세한 내용은 What's New in Redis 5?를 참조하십시오. 다음 표에는 새 이름 및 이러한 이름이 이전 버전과 매핑되는 방법이 나와 있습니다.

Redis 5.0에서 이름이 변경된 파라미터

이름	세부 정보	설명
replica-	기본값: 아니요	복제본 동기화 동안 비동기식 flushDB를 수행합니
lazy-flush	유형: 부울	다.
	수정 가능 여부: 아니요	
	이전 이름: slave-lazy-flush	
client- output- buffer- limit- replica- hard-limit	기본값: 값은 Redis 노드 유형별 파라미터 (p. 295)를 참조하십시 오.	Redis 읽기 전용 복제본: 클라이언트의 출력 버퍼가 특정 바이트 수에 도달하면 클라이언트가 연결 해제 됩니다.
	유형: 정수	
	수정 가능 여부: 아니요	
	이전 이름: client-output-buffer- limit-slave-hard-limit	
client- output- buffer- limit-	기본값: 값은 Redis 노드 유형별 파라미터 (p. 295)를 참조하십시 오. 유형: 정수	Redis 읽기 전용 복제본: 클라이언트의 출력 버퍼가 특정 바이트 수에 도달하면 클라이언트가 연결 해제 됩니다. 그러나 이러한 조건은 client-output- buffer-limit-replica-soft-seconds의 경 우에만 지속됩니다.

이름	세부 정보	설명
replica- soft-limit	수정 가능 여부: 아니요 이전 이름: client-output-buffer- limit-slave-soft-limit	
client- output- buffer- limit- replica- soft- seconds	기본값: 60 유형: 정수 수정 가능 여부: 아니요 이전 이름: client-output-buffer- limit-slave-soft-seconds	Redis 읽기 전용 복제본: 클라이언트의 출력 버퍼가 client-output-buffer-limit-replica-soft-limit 바이트에 해당 시간(초)보다 오래 유지되면 클라이언트가 연결 해제됩니다.
replica- allow- chaining	기본값: 아니요 유형: 문자열 수정 가능 여부: 아니요 이전 이름: slave-allow-chaining	Redis가 자체 읽기 전용 복제본을 가질 수 있는지를 결정합니다.
min- replicas- to-write	기본값: 0 유형: 정수 수정 가능 여부: 예 이전 이름: min-slaves-to-write 변경 사항 적용: 즉시	기본 노드가 클러스터에서 쓰기를 허용하기 위해 사용 가능해야 하는 최소 읽기 전용 복제본 수입니다. 사용 가능한 복제본 수가 이 수보다 떨어지면 기본 노드는 더 이상 쓰기 요청을 허용하지 않습니다. 이 파라미터 또는 min-replicas-max-lag가 0이면 사용 가능한 복제본이 없어도 기본 노드가 항상 쓰기 요청을 허용합니다.
min- replicas- max-lag	기본값: 10 유형: 정수 수정 가능 여부: 예 이전 이름: min-slaves-max-lag 변경 사항 적용: 즉시	기본 노드가 읽기 전용 복제본에서 핑 요청을 수신 해야 하는 시간(초)입니다. 이 시간까지 기본 노드가 핑을 수신하지 않으면 복제본을 더 이상 사용할 수 없는 것으로 간주합니다. 사용 가능한 복제본 수가 min-replicas-to-write 아래로 떨어지면 기본 복제본 이 해당 시점에서 쓰기 허용을 중지합니다. 이 파라미터 또는 min-replicas-to-write가 0이면 사 용 가능한 복제본이 없어도 기본 노드가 항상 쓰기 요청을 허용합니다.
close-on- replica- write	기본값: yes 유형: 부울 수정 가능 여부: 예 이전 이름: close-on-slave-write 변경 사항 적용: 즉시	활성화하면 읽기 전용 복제본에 작성을 시도하는 클라이언트의 연결이 끊어집니다.

Redis 5.0에서 제거된 파라미터

이름	세부 정보	설명
repl- timeout	기본값: 60	이 버전에서는 파라미터를 사용할 수 없습니다.
timeout	수정 가능 여부: 아니요	

Redis 4.0.10 파라미터 변경 사항

파라미터 그룹 패밀리: redis4.0 Redis 4.0.x 기본 파라미터 그룹

- default.redis4.0 이 파라미터 그룹 또는 Redis(클러스터 모드 비활성화됨) 클러스터 및 복제 그룹 으로부터 파생된 파라미터 그룹을 사용합니다.
- default.redis4.0.cluster.on 이 파라미터 그룹 또는 Redis(클러스터 모드 활성화됨) 클러스터 및 복제 그룹으로부터 파생된 파라미터 그룹을 사용합니다.

Redis 4.0.10에서 변경된 파라미터

이름	세부 정보	설명
maxmemory- policy	허용되는 값: allkeys-lru, volatile-lru, allkeys-lfu, volatile-lfu, allkeys-random, volatile-random, volatile-ttl, noeviction 기본값: volatile-lru 유형: 문자열 수정 가능 여부: 예 변경 사항 적용: 즉시	버전 2.6.13에는 maxmemory-policy 단원이 추가되었습니다. 4.0.10에는 새로 허용된 두 개의 값이추가되었습니다. allkeys-lfu는 근사 LFU를 사용하여 모든 키를 제거하고 volatile-lfu는 근사LFU를 사용하여 키 중 만료 설정이 있는 키를 제거합니다.

Redis 4.0.10에서 추가된 파라미터

이름	세부 정보	설명
비동기 삭제 피	i 라마터	
lazyfree-	허용되는 값: yes/no	제거 시 비동기식 삭제를 수행합니다.
lazy- eviction	기본값: 아니요	
	유형: 부울	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
lazyfree-	허용되는 값: yes/no	키 만료 시 비동기식 삭제를 수행합니다.
lazy- expire	기본값: 아니요	
	유형: 부울	

이름	세부 정보	설명
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
lazyfree-	허용되는 값: yes/no	값을 업데이트하는 명령에 대해 비동기식 삭
lazy- server-	기본값: 아니요	제를 수행합니다.
del	유형: 부울	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
slave-	허용되는 값: N/A	slave sync 동안 비동기식 flushDB를 수행합
lazy- flush	기본값: 아니요	니다.
	유형: 부울	
	수정 가능 여부: 아니요 변경 사항 적용: N/A	
LFU 파라미터		
lfu-log-	허용되는 values: any integer > 0	키 카운터를 포화시키는 키 히트 수를 결정
factor	기본값: 10	하는 로그 팩터를 설정합니다.
	유형: 정수	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
lfu-	허용되는 값: 모든 정수	키 카운터 감소에 소요된 시간입니다(분).
decay- time	기본값: 1	
	유형: 정수	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
활성 조각 모음 파라미터		
activedefra	a화용되는 값: yes/no	활성 조각 모음 파라미터가 활성화됩니다.
	기본값: 아니요	
	유형: 부울	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	

이름	세부 정보	설명	
active- defrag- ignore- bytes	허용되는 값: 10485760-104857600 기본값: 104857600	활성 조각 모음을 시작하는 조각의 최소 수 입니다.	
	유형: 정수		
	수정 가능 여부: 예		
	변경 사항 적용: 즉시		
active- defrag- threshold- lower	허용되는 값: 1-100	활성 조각 모음을 시작하는 조각의 최소 비 율입니다.	
	기본값: 10		
	유형: 정수		
	수정 가능 여부: 예		
	변경 사항 적용: 즉시		
active- defrag- threshold- upper	허용되는 값: 1-100	최대 작업을 사용하는 조각의 최대 비율입니다.	
	기본값: 100		
	유형: 정수		
	수정 가능 여부: 예		
	변경 사항 적용: 즉시		
active- defrag- cycle-min	허용되는 값: 1-75	CPU 비율의 조각 모음에 대한 최소 작업입니다.	
	기본값: 25		
	유형: 정수		
	수정 가능 여부: 예		
	변경 사항 적용: 즉시		
active- defrag- cycle-max	허용되는 값: 1-75	CPU 비율의 조각 모음에 대한 최대 작업입니다.	
	기본값: 75		
	유형: 정수		
	수정 가능 여부: 예		
	변경 사항 적용: 즉시		
클라이언트 출력 버퍼 파라미터			

이름	세부 정보	설명
client- query- buffer- limit	허용되는 값: 1048576-1073741824	단일 클라이언트 쿼리 버퍼의 최대 크기입니 다.
	기본값: 1073741824	J.
	유형: 정수	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	
proto- max-bulk- len	허용되는 값: 1048576-536870912	단일 요소 요청의 최대 크기입니다.
	기본값: 536870912	
	유형: 정수	
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	

Redis 3.2.10 파라미터 변경 사항

파라미터 그룹 패밀리: redis3.2

Redis용 ElastiCache 3.2.10에는 지원되는 추가 파라미터가 없습니다.

Redis 3.2.6 파라미터 변경 사항

파라미터 그룹 패밀리: redis3.2

Redis 3.2.6의 경우 지원되는 추가 파라미터가 없습니다.

Redis 3.2.4 파라미터 변경 사항

파라미터 그룹 패밀리: redis3.2

Redis 3.2.4부터 기본 파라미터 그룹이 2개 있습니다.

- default.redis3.2 Redis(클러스터 모드 비활성화됨) 복제 그룹을 생성하고 Redis 3.2.4.의 추가 기능을 계속 사용하려면 Redis 3.2.4를 실행할 때 이 파라미터 그룹 또는 그로부터 파생된 파라미터 그룹을 지정합니다.
- default.redis3.2.cluster.on Redis(클러스터 모드 활성화됨) 복제 그룹을 생성하려면 파라미터 그룹 또는 그로부터 파생된 파라미터 그룹을 지정합니다.

주제

- Redis 3.2.4의 새 파라미터 (p. 282)
- Redis 3.2.4(개선 됨)에서 파라미터 변경 (p. 284)

Redis 3.2.4의 새 파라미터

파라미터 그룹 패밀리: redis3.2

Redis 3.2.4의 경우 다음과 같은 추가 파라미터를 지원합니다.

이름	세부 정보	설명
list-max- ziplist- size	기본값: -2 유형: 정수 수정 가능 여부: 아니요	목록은 공간을 절약하기 위해 특별한 방법으로 인코 딩됩니다. 내부 목록 노드 당 허용되는 항목 수는 요 소의 최대 수 또는 최대 고정 크기로 지정할 수 있습 니다. 최대 고정 크기의 경우 -5 ~ -1을 사용합니다. • -5: 최대 크기: 64Kb - 일반 워크로드에 권장되지 않음 • -4: 최대 크기: 32Kb - 권장되지 않음 • -3: 최대 크기: 16Kb - 권장되지 않음 • -2: 최대 크기: 8Kb - 권장 • -1: 최대 크기: 4Kb - 권장 • 양수는 목록 노드당 정확히 그 수 만큼의 요소를 저장합니다.
list- compress- depth	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	목록을 압축할 수도 있습니다. 압축 깊이는 압축에서 제외할 목록 각 측면의 퀵리스트 집리스트 노드수입니다. 목록의 헤드와 테일은 빠른 푸시 및 팝 작업을 위해 항상 압축하지 않습니다. 설정: • 0: 모든 압축을 해제합니다. • 1: 헤드와 테일에서 첫 번째 노드로 압축을 시작합니다. [헤드]->노드->노드->>노드->[테일] [헤드]와 [테일]을 제외한 모든 노드를 압축합니다. • 2: 헤드와 테일에서 두 번째 노드로 압축을 시작합니다. [헤드]->[다음]->노드->노드->>노드->[이전]->[테일] [헤드]->[다음], [이전], [테일]은 압축하지 않습니다.다른 모든 노드를 압축합니다.
cluster- enabled	기본값: 아니요/예 * 유형: 문자열 수정 가능 여부: 예	클러스터 모드(yes)의 Redis(클러스터 모드 활성화됨) 복제 그룹인지 비클러스터 모드(no)의 Redis(클러스터 모드 활성화됨) 복제 그룹인지를 나타냅니다. 클러스터 모드의 Redis(클러스터 모드 활성화됨) 복제 그룹은 최대 90개의 노드 그룹에 데이터를 분할할 수 있습니다. * Redis 3.2.x에는 기본 파라미터 그룹 2개가 있습니다. • default.redis3.2 - 기본 값 no. • default.redis3.2.cluster.on - 기본 값 yes.

이름	세부 정보	설명
cluster- require- full- coverage	기본값: 아니요 유형: 부울 수정 가능 여부: 예 변경 사항 적용: 즉시	yes로 설정했을 때 클러스터 모드의 Redis(클러스터 모드 활성화됨) 노드는 확인되지 않은 해시 슬롯을 하나 이상 감지하면 더 이상 쿼리를 허용하지 않습니다(사용할 수 있는 노드가 없는 경우). 클러스터가 부분적으로 가동 중지되면 클러스터를 사용할 수 없게 됩니다. 모든 슬롯이 다시 확인되는 즉시 자동으로 사용할 수 있게 됩니다. 그러나 때로는 확인된 일부 키스페이스의 쿼리를 계속해서 허용하는 작업 중인 클러스터의 하위 세트가필요합니다. 이렇게 하려면 cluster-requirefull-coverage 옵션을 no로 설정합니다.
hll-sparse- max-bytes	기본값: 3000 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	HyperLogLog 스파스 표현 바이트 제한입니다. 제한은 16바이트 헤더를 포함합니다. 스파스 표현을 사용하는 HyperLogLog가 이 제한을 초과하면 밀도가높은 표현으로 변환됩니다. 그 시점에서는 밀도가 높은 표현이 메모리 효율을높이기 때문에 16000보다 큰 값은 권장하지 않습니다. 스파스 인코딩이 너무 많은 O(N)인 PFADD를 너무느리게 하지 않고 공간 효율적인 인코딩의 이점을얻으려면 값을 약 3000까지로 하는 것이 좋습니다. CPU의 문제가 아니라 공백이 있고 데이터 세트가 0~15000 범위의 카디널리티(cardinality)를 가진 많은 HyperLogLog로 구성되어 있으면 값을 10000까지 올릴 수 있습니다.
reserved- memory- percent	기본값: 25 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	비데이터 사용을 위해 예약된 노드의 메모리 비율입니다. 기본적으로 Redis 데이터 공간은 노드의 메모리를 모두 소진할 때까지 증가합니다. 이 경우 과도한 메모리 페이징으로 인해 노드 성능이 저하될 수 있습니다. 메모리를 예약하면 페이징 양을 줄일 수 있도록 Redis가 아닌 용도로 사용 가능한 메모리 일부를 구분하여 설정할 수 있습니다. 이 파라미터는 ElastiCache에 고유하며 표준 Redis배포의 일부가 아닙니다. 자세한 내용은 reserved-memory 및 예약된 메모리 관리 (p. 405) 단원을 참조하십시오.

Redis 3.2.4(개선 됨)에서 파라미터 변경

파라미터 그룹 패밀리: redis3.2

Redis 3.2.4에서는 다음 파라미터가 변경되었습니다.

이름	세부 정보	변경 사항
activerehash	ing정 가능 여부: 예	수정 가능 여부는 '아니요'였습니다.

이름	세부 정보	변경 사항
databases	수정 가능 여부: 파라미터 그룹이 캐시 클러스터와 연결되어 있지 않은 경우 예이고, 그렇지 않으면 아니오입니다.	수정 가능 여부는 '아니요'였습니다.
appendonly	기본값: 꺼짐 수정 가능 여부: 불가능	이전 Redis 버전에서 업그레이드하려면 먼저 appendonly를 해제해야 합니다.
appendfsync	기본값: 꺼짐 수정 가능 여부: 불가능	이전 Redis 버전에서 업그레이드하려면 먼저 appendfsync를 해제해야 합니다.
repl- timeout	기본값: 60 수정 가능 여부: 불가능	현재 기본값을 60으로 수정할 수 없습니다.
tcp- keepalive	기본값: 300	기본값은 0입니다.
list-max- ziplist- entries		파라미터를 더 이상 사용할 수 없습니다.
list-max- ziplist- value		파라미터를 더 이상 사용할 수 없습니다.

Redis 2.8.24(개선 됨) 추가 파라미터

파라미터 그룹 패밀리: redis2.8

Redis 2.8.24의 경우 지원되는 추가 파라미터가 없습니다.

Redis 2.8.23(개선 됨) 추가 파라미터

파라미터 그룹 패밀리: redis2.8

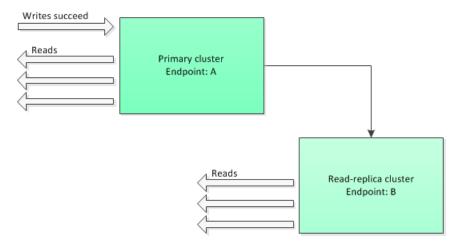
Redis 2.8.23의 경우 다음과 같은 추가 파라미터가 지원됩니다.

이름	세부 정보	설명
close-on-	기본값: yes	활성화하면 읽기 전용 복제본에 작성을 시도하는 클 라이언트의 연결이 끊어집니다.
slave-write	유형: 문자열(yes/no)	다이언트의 언결이 끊어겁니다.
	수정 가능 여부: 예	
	변경 사항 적용: 즉시	

close-on-slave-write의 작동 방식

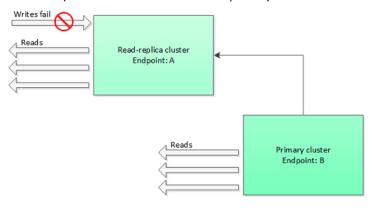
Amazon ElastiCache가 close-on-slave-write 파라미터를 도입하여 읽기 복제본을 기본으로 승격시켜 기본 노드 및 읽기 복제본 노드의 역할을 바꾸면 클러스터가 응답하는 방법을 보다 세부적으로 제어할 수 있도록 합니다.

Before read-replica promotion



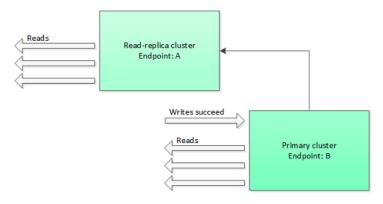
다중 AZ가 활성화된 복제 그룹 장애 조치 이외의 다른 이유로 읽기 전용 복제본 클러스터가 기본으로 승격되면 클라이언트는 엔드포인트 A에 계속 쓰려고 시도합니다. 그러나 엔드포인트 A가 읽기 전용 복제본의 엔드포인트이기 때문에 쓰기가 실패합니다. close-on-replica-write가 ElastiCache를 도입하기 전에 Redis의 동작이며 close-on-replica-write를 비활성화한 경우의 동작입니다.

Read-replica Promoted - writes to old primary fail



close-on-replica-write를 활성화하면 클라이언트가 읽기 전용 복제본에 쓰기를 시도할 때마다 클러스터와의 클라이언트 연결이 종료됩니다. 애플리케이션 논리가 연결 해제를 감지하고 DNS 테이블을 확인한다음 이제 기본 엔드포인트(엔드포인트 B)에 다시 연결합니다.

Client reconnected to new primary



close-on-replica-write를 비활성화할 때

close-on-replica-write를 비활성화했는데 장애가 발생한 클러스터에 쓴 경우 close-on-replica-write를 비활성화하는 이유는 무엇일까요?

앞서 언급했듯이 close-on-replica-write를 활성화하면 클라이언트가 읽기 전용 복제본에 쓰기를 시도할 때마다 클러스터와의 클라이언트 연결이 종료됩니다. 노드에 새로운 연결을 설정하는 것은 시간이 소요됩니다. 따라서 복제본에 대한 쓰기 요청의 결과로 연결을 끊고 다시 연결하면 동일한 연결을 통해 제공되는 읽기 요청의 지연 시간에도 영향을 미칩니다. 새로운 연결이 설정될 때까지 이 영향이 그대로 유지됩니다. 애플리케이션이 특별히 읽기 중심이거나 지연 시간에 매우 민감한 경우, 읽기 성능이 저하되지 않도록 클라이언트 연결을 유지하는 것이 좋습니다.

Redis 2.8.22(개선됨) 추가 파라미터

파라미터 그룹 패밀리: redis2.8

Redis 2.8.22의 경우 지원되는 추가 파라미터가 없습니다.

Important

- Redis 버전 2.8.22부터 repl-backlog-size가 기본 클러스터와 복제본 클러스터에 적용됩니다.
- Redis 버전 2.8.22부터 repl-timeout 파라미터를 지원하지 않습니다. 변경된 경우 appendonly와 같이 ElastiCache는 기본값(60초)로 덮어씁니다.

다음 파라미터는 더 이상 지원되지 않습니다.

- appendonly
- appendfsync
- · repl-timeout

Redis 2.8.21 추가 파라미터

파라미터 그룹 패밀리: redis2.8

Redis 2.8.21은 추가 파라미터를 지원하지 않습니다.

Redis 2.8.19 추가 파라미터

파라미터 그룹 패밀리: redis2.8

Redis 2.8.19의 경우 지원되는 추가 파라미터가 없습니다.

Redis 2.8.6 추가 파라미터

파라미터 그룹 패밀리: redis2.8

Redis 2.8.6의 경우 다음과 같은 추가 파라미터가 지원됩니다.

이름	세부 정보	설명
min-slaves-max-lag	기본값: 10 유형: 정수	기본 노드가 읽기 전용 복제본에서 핑 요청을 수신해야 하는 시간(초)입니다.
	수정 가능 여부: 예	이 시간까지 기본 노드가 핑을 수신하 지 않으면 복제본을 더 이상 사용할 수 없는 것으로 간주합니다. 사용 가능한

이름	세부 정보	설명
	변경 사항 적용: 즉시	복제본 수가 min-slaves-to-write 아래로 떨어지면 기본 복제본이 해당 시점에서 쓰기 허용을 중지합니다.
		이 파라미터 또는 min-slaves-to-write 가 0이면 사용 가능한 복제본이 없어 도 기본 노드가 항상 쓰기 요청을 허용 합니다.
min-slaves-to-write	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	기본 노드가 클러스터에서 쓰기를 허용하기 위해 사용 가능해야 하는 최소 위기 전용 복제본 수입니다. 사용 가능 한 복제본 수가 이 수보다 떨어지면 기 본 노드는 더 이상 쓰기 요청을 허용하 지 않습니다.
		이 파라미터 또는 min-slaves-max-lag 가 0이면 사용 가능한 복제본이 없어 도 기본 노드가 항상 쓰기 요청을 허용 합니다.

이름	세부 정보	설명
notify-keyspace-events	기본값: (빈 문자열) 유형: 문자열 수정 가능 여부: 예 변경 사항 적용: 즉시	Redis가 클라이언트에 알릴 수 있는 키스페이스 이벤트 유형입니다. 각 이벤트 유형은 한 글자로 표현됩니다. • K —keyspace@ <db>으로 게시된 키스페이스 이벤트 • E — 접두사가keyevent@<db>로 게시된 키-이벤트 이벤트 • g — DEL, EXPIRE, RENAME 등과 같은 일반적인 명령 • 1 — 나열 명령 • 1 — 다열 명령 • 2 — 정렬된 세트 명령 • 1 — 다열된 이벤트(키가 만료될 때마다 생성되는 이벤트) • 0 — 제거된 이벤트(최대 메모리로키가 제거될 때 생성되는 이벤트) • 1 — 대한 이벤트(최대 메모리로키가 제거될 때 생성되는 이벤트) • 1 — 대한 이벤트(최대 메모리로키가 제거될 때 생성되는 이벤트) • 2 — 제가된 이벤트(최대 메모리로키가 제거될 때 생성되는 이벤트) • 1 — 대한 이벤트 유형을 가유롭게 조합할 수 있습니다. 예를 들어, AKE는 Redis가 모든 이벤트 유형의 알림을게시할 수 있음을 의미합니다. 오류 메시지가 발생할 수 있으므로 위에 나열된 문자 이외의 다른 문자로 시도하지마십시오. 기본적으로 이 파라미터는 빈 문자열로 설정되어 있습니다. 즉, 키스페이스이벤트 알림이 비활성화되어 있습니다. 즉, 키스페이스이벤트 알림이 비활성화되어 있습니다.</db></db>
repl-backlog-size	기본값: 1048576 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	기본 노드 백로그 버퍼의 크기(바이트) 입니다. 백로그는 기본 노드의 데이터 에 대한 업데이트를 레코딩하는 데 사 용됩니다. 읽기 전용 복제본이 기본에 연결되면 기본 노드를 따라잡기 위해 백로그에서 데이터를 적용하는 부분적 동기화(psync)를 수행하려고 시도합 니다. psync가 실패하면 전체 동기화 가 필요합니다. 이 파라미터의 최소값은 16384입니다. Note Redis 2.8.22부터 이 파라미 터는 기본 클러스터와 읽기 전용 복제본에 적용됩니다.

이름	세부 정보	설명
repl-backlog-ttl	기본값: 3600 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	기본 노드가 백로그 버퍼를 보관할 시간(초)입 니다. 마지막 복제본 노드가연결 해제된 시점부터 백로그의 데이터는 repl-backlog-ttl이 만료될때까지 그대로 유지됩니다. 이 시간안에 복제본이 기본 노드에 연결되지않으면 기본이 백로그 버퍼를 해제합니다. 결국 복제본이 다시 연결되면 기본과 전체 동기화를 수행해야 합니다. 파라미터를 0으로 설정하면 백로그 버퍼가 절대 해제되지 않습니다.
repl-timeout	기본값: 60 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	다음에 대한 제한 시간(초)을 나타냅니다. • 읽기 전용 복제본의 관점에서 동기화 중 벌크 데이터 전송 • 복제본의 관점에서 기본 노드 제한시간 • 기본 노드의 관점에서 복제본 제한시간

Redis 2.6.13 파라미터

파라미터 그룹 패밀리: redis2.6

Redis 2.6.13은 ElastiCache가 지원하는 초기 버전이었습니다. 다음은 ElastiCache가 지원하는 Redis 2.6.13 파라미터를 보여주는 표입니다.

이름	세부 정보	설명
유형: 문자열(yes/no)		Redis의 활성 재해싱 기능을 사용할지 결정합니다. 기본 해시 테이블은 초당 10회 재해시되며 각 해시 작업은 1밀리초의 CPU 시간을 소비합니다.
	수정 가능 여부: 예 변경 사항 적용: 생성 시	이 값은 파라미터 그룹을 생성할 때 설정됩니다. 새 파라미터 그룹을 클러스터에 할당할 때 이전 파라미 터 그룹과 새 파라미터 그룹에서 값이 동일해야 합 니다.
appendonly	기본값: 아니요 유형: 문자열 수정 가능 여부: 예	Redis의 AOF(append only file)를 활성화하거나 비 활성화합니다. AOF는 캐시에 있는 데이터를 변경하 는 Redis 명령을 캡처하고 특정 노드 오류를 복구하 는 데 사용됩니다.
	변경 사항 적용: 즉시	기본값이 [no]이면 AOF가 해제되어 있다는 의미입 니다. AOF를 활성화하려면 이 파라미터를 [yes]로 설정합니다.
		자세한 내용은 장애 완화 (p. 395) 단원을 참조하 십시오.

이름	세부 정보	설명
		Note AOF(append-only file)는 cache.t1.micro 및 cache.t2.* 노드가 지원되지 않습니다. 이 유형의 노드에서는 appendonly 파라미터 값이 무시됩니다. Note 다중 AZ 복제 그룹은 AOF를 허용하지 않습니다.
appendfsync	기본값: everysec 유형: 문자열 수정 가능 여부: 예 변경 사항 적용: 즉시	appendonly가 '예'로 설정된 경우 AOF 출력 버퍼가 디스크에 기록되는 빈도를 제어합니다. • no — 버퍼가 필요에 따라 디스크로 플러시됩니다. • everysec — 버퍼가 1초에 한번씩 플러시됩니다.이 값이 기본값입니다. • always — 클러스터의 데이터가 수정될 때마다 버퍼가 플러시됩니다. • 버전 2.8.22 이상에서는 appendfsync가 지원되지 않습니다.
client- output- buffer- limit- normal- hard-limit	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	클라이언트의 출력 버퍼가 특정 바이트 수에 도달하면 클라이언트가 연결 해제됩니다. 기본값은 0입니다(하드 제한 없음).
client- output- buffer- limit- normal- soft-limit	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	클라이언트의 출력 버퍼가 특정 바이트 수에 도달 하면 클라이언트가 연결 해제됩니다. 그러나 이러 한 조건은 client-output-buffer-limit- normal-soft-seconds의 경우에만 지속됩니다. 기본값은 0입니다(소프트 제한 없음).
client- output- buffer- limit- normal- soft- seconds	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	클라이언트의 출력 버퍼가 client-output- buffer-limit-normal-soft-limit 바이트에 해당 시간(초)보다 오래 유지되면 클라이언트가 연 결 해제됩니다. 기본값은 0입니다(시간 제한 없음).
client- output- buffer- limit- pubsub- hard-limit	기본값: 33554432 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	Redis 게시/구독 클라이언트: 클라이언트의 출력 버퍼가 특정 바이트 수에 도달하면 클라이언트가 연결해제됩니다.

이름	세부 정보	설명
client- output- buffer- limit- pubsub- soft-limit	기본값: 8388608 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	Redis 게시/구독 클라이언트: 클라이언트의 출력 버퍼가 특정 바이트 수에 도달하면 클라이언트가 연결 해제됩니다. 그러나 이러한 조건은 client-output-buffer-limit-pubsub-soft-seconds의 경우에만 지속됩니다.
client- output- buffer- limit- pubsub- soft- seconds	기본값: 60 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	Redis 게시/구독 클라이언트: 클라이언트의 출력 버퍼가 client-output-buffer-limit-pubsub-soft-limit 바이트에 해당 시간(초)보다 오래 유지되면 클라이언트가 연결 해제됩니다
client- output- buffer- limit- slave-hard- limit	기본값: 값은 Redis 노드 유형별 파라미터 (p. 295)를 참조하십시 오. 유형: 정수 수정 가능 여부: 아니요	Redis 읽기 전용 복제본: 클라이언트의 출력 버퍼가 특정 바이트 수에 도달하면 클라이언트가 연결 해제 됩니다.
client- output- buffer- limit- slave-soft- limit	기본값: 값은 Redis 노드 유형별 파라미터 (p. 295)를 참조하십시 오. 유형: 정수 수정 가능 여부: 아니요	Redis 읽기 전용 복제본: 클라이언트의 출력 버퍼가 특정 바이트 수에 도달하면 클라이언트가 연결 해제 됩니다. 그러나 이러한 조건은 client-output- buffer-limit-slave-soft-seconds의 경우에 만 지속됩니다.
client- output- buffer- limit- slave-soft- seconds	기본값: 60 유형: 정수 수정 가능 여부: 아니요	Redis 읽기 전용 복제본: 클라이언트의 출력 버퍼가 client-output-buffer-limit-slave-soft-limit 바이트에 해당 시간(초)보다 오래 유지되면클라이언트가 연결 해제됩니다.
databases	기본값: 16 유형: 정수 수정 가능 여부: 아니요 변경 사항 적용: 생성 시	데이터베이스가 분할되는 논리적 피티션의 수입니다. 이 값을 낮게 유지하는 것이 좋습니다. 이 값은 파라미터 그룹을 생성할 때 설정됩니다. 새파라미터 그룹을 클러스터에 할당할 때 이전 파라미터 그룹과 새 파라미터 그룹에서 값이 동일해야 합니다.
hash-max- ziplist- entries	기본값: 512 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	해시에 사용되는 메모리 양을 결정합니다. 지정된 수보다 적은 수의 항목이 있는 해시는 공간을 절약 하는 특수 인코딩을 사용하여 저장됩니다.

이름	세부 정보	설명
hash-max- ziplist- value	기본값: 64 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	해시에 사용되는 메모리 양을 결정합니다. 지정된 바이트 수보다 작은 항목이 있는 해시는 공간을 절 약하는 특수 인코딩을 사용하여 저장됩니다.
list-max- ziplist- entries	기본값: 512 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	목록에 사용되는 메모리 양을 결정합니다. 지정된 수보다 적은 수의 항목이 있는 목록은 공간을 절약 하는 특수 인코딩을 사용하여 저장됩니다.
list-max- ziplist- value	기본값: 64 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	목록에 사용되는 메모리 양을 결정합니다. 지정된 바이트 수보다 작은 항목이 있는 목록은 공간을 절 약하는 특수 인코딩을 사용하여 저장됩니다.
lua-time- limit	기본값: 5000 유형: 정수 수정 가능 여부: 아니요	ElastiCache가 스크립트를 중지하기 전 Lua 스크립트의 최대 실행 시간(밀리초)입니다. lua-time-limit을 초과하면 모든 Redis 명령은BUSY 형식의 오류를 반환합니다. 이런 상태는 여러 가지 필수적인 Redis 작업에 방해가 될 수 있으므로 ElastiCache는 먼저 SCRIPT KILL 명령을 실행합니다. 실패할 경우 ElastiCache는 Redis를 강제로다시 시작합니다.
maxclients	기본값: 65000 유형: 정수 수정 가능 여부: 아니요	한 번에 연결할 수 있는 최대 클라이언트 수입니다.
maxmemory-policy	기본값: volatile-lru 유형: 문자열 수정 가능 여부: 예 변경 사항 적용: 즉시	최대 메모리 사용량에 도달했을 때 키에 대한 제거 정책입니다. 유효한 값은 volatile-lru allkeys-lru volatile-random allkeys-random volatile-ttl noeviction입니다. 자세한 내용은 LRU 캐시로 Redis 사용을 참조하십 시오.
maxmemory- samples	기본값: 3 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	LRU(가장 오랫동안 사용되지 않음) 및 TTL(Time To Live) 계산의 경우 파라미터는 확인할 키의 샘플 크기를 나타냅니다. 기본적으로 Redis는 키 3개를 선택하고 가장 최근에 사용한 키를 사용합니다.

이름	세부 정보	설명
reserved- memory	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	비데이터 사용을 위해 예약된 총 메모리(바이트)입니다. 기본적으로 Redis 노드는 노드의 maxmemory를 소진할 때까지 증가합니다(Redis 노드 유형별 파라미터 (p. 295) 참조). 이 경우 과도한 메모리 페이징으로 인해 노드 성능이 저하될 수 있습니다. 메모리를 예약하면 페이징 양을 줄일 수 있도록 Redis가 아닌 용도로 사용 가능한 메모리 일부를 구분하여 설정할 수 있습니다. 이 파라미터는 ElastiCache에 고유하며 표준 Redis 배포의 일부가 아닙니다. 자세한 내용은 reserved-memory-percent 및예약된 메모리 관리 (p. 405) 단원을 참조하십시오.
set-max- intset- entries	기본값: 512 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	특정 종류의 세트(64비트 부호가 있는 정수의 범위에서 기수 10의 정수 문자열)에 사용되는 메모리의 양을 결정합니다. 지정된 수보다 적은 수의 항목이 있는 세트는 공간을 절약하는 특수 인코딩을 사용하여 저장됩니다.
slave- allow- chaining	기본값: 아니요 유형: 문자열 수정 가능 여부: 아니요	Redis가 자체 읽기 전용 복제본을 가질 수 있는지를 결정합니다.
slowlog- log-slower- than	기본값: 10000 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	Redis 느린 로그 기능으로 기록할 명령의 최대 실행 시간(마이크로초)입니다.
slowlog- max-len	기본값: 128 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	Redis 느린 로그의 최대 길이입니다.
tcp- keepalive	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	이 값을 0이 아닌 값(N)으로 설정하면 연결이 되어 있는지 확인하기 위해 노드 클라이언트가 N초마다 폴링됩니다. 기본 설정인 0을 사용하면 폴링이 발생 하지 않습니다. Important Redis 버전 3.2.4에서 변경된 파라미터 의 일부 측면에 대해서는 을 참조하십시 오. Redis 3.2.4(개선 됨)에서 파라미터 변 경 (p. 284) 단원을 참조하십시오.

이름	세부 정보	설명
timeout	기본값: 0 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	제한 시간이 지나기 전에 노드가 대기하는 시간(초)입니다. 유효한 값은 입니다. • 0 – 유휴 클라이언트 연결을 절대로 끊지 마십시오. • 1-19 – 잘못된 값입니다. • >=20 – 유휴 클라이언트 연결을 끊기 전에 노드가대기하는 시간(초)입니다.
zset-max- ziplist- entries	기본값: 128 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	정렬된 세트에 사용할 메모리 양을 결정합니다. 지 정된 수보다 적은 수의 정렬된 세트는 공간을 절약 하는 특수 인코딩을 사용하여 저장됩니다.
zset-max- ziplist- value	기본값: 64 유형: 정수 수정 가능 여부: 예 변경 사항 적용: 즉시	정렬된 세트에 사용할 메모리 양을 결정합니다. 지 정된 바이트 수보다 작은 항목이 있는 정렬된 세트 는 공간을 절약하는 특수 인코딩을 사용하여 저장됩 니다.

Note

Redis 2.6.13 클러스터의 파라미터 그룹을 지정하지 않으면 기본 파라미터 그룹 (default.redis2.6)이 사용됩니다. 기본 파라미터 그룹의 파라미터 값은 변경할 수 없지만 언제든지 사용자 지정 파라미터 그룹을 생성하고 클러스터에 할당할 수 있습니다.

Redis 노드 유형별 파라미터

대부분의 파라미터는 단일 값을 갖지만 일부 파라미터는 사용하는 노드 유형에 따라 다양한 값을 갖습니다. 다음 표에는 각 노드 유형에 대한 maxmemory, client-output-buffer-limit-slave-hard-limit 및 client-output-buffer-limit-slave-soft-limit 파라미터의 기본값이 나와 있습니다. maxmemory의 값은 노드에서 데이터 및 다른 용도에 사용할 수 있는 최대 바이트 수입니다.

Note

maxmemory 파라미터는 수정할 수 없습니다.

노드 유형	maxmemory	client-output-buffer- limit-slave-hard-limit	client-output-buffer- limit-slave-soft-limit
cache.t1.micro	142606336	14260633	14260633
cache.t2.micro	581959680	58195968	58195968
cache.t2.small	1665138688	166513868	166513868
cache.t2.medium	3461349376	346134937	346134937
cache.t3.micro	536870912	53687091	53687091
cache.t3.small	1471026299	147102629	147102629

노드 유형	maxmemory	client-output-buffer- limit-slave-hard-limit	client-output-buffer- limit-slave-soft-limit
cache.t3.medium	3317862236	331786223	331786223
cache.m1.small	943718400	94371840	94371840
cache.m1.medium	3093299200	309329920	309329920
cache.m1.large	7025459200	702545920	702545920
cache.m1.xlarge	14889779200	1488977920	1488977920
cache.m2.xlarge	17091788800	1709178880	1709178880
cache.m2.2xlarge	35022438400	3502243840	3502243840
cache.m2.4xlarge	70883737600	7088373760	7088373760
cache.m3.medium	2988441600	309329920	309329920
cache.m3.large	6501171200	650117120	650117120
cache.m3.xlarge	14260633600	1426063360	1426063360
cache.m3.2xlarge	29989273600	2998927360	2998927360
cache.m4.large	6892593152	689259315	689259315
cache.m4.xlarge	15328501760	1532850176	1532850176
cache.m4.2xlarge	31889126359	3188912636	3188912636
cache.m4.4xlarge	65257290629	6525729063	6525729063
cache.m4.10xlarge	166047614239	16604761424	16604761424
cache.m5.large	6854542746	685454275	685454275
cache.m5.xlarge	13891921715	1389192172	1389192172
cache.m5.2xlarge	27966669210	2796666921	2796666921
cache.m5.4xlarge	56116178125	5611617812	5611617812
cache.m5.12xlarge	168715971994	16871597199	16871597199
cache.m5.24xlarge	337500562842	33750056284	33750056284
cache.c1.xlarge	6501171200	650117120	650117120
cache.r3.large	14470348800	1468006400	1468006400
cache.r3.xlarge	30513561600	3040870400	3040870400
cache.r3.2xlarge	62495129600	6081740800	6081740800
cache.r3.4xlarge	126458265600	12268339200	12268339200
cache.r3.8xlarge	254384537600	24536678400	24536678400
cache.r4.large	13201781556	1320178155	1320178155

노드 유형	maxmemory	client-output-buffer- limit-slave-hard-limit	client-output-buffer- limit-slave-soft-limit
cache.r4.xlarge	26898228839	2689822883	2689822883
cache.r4.2xlarge	54197537997	5419753799	5419753799
cache.r4.4xlarge	108858546586	10885854658	10885854658
cache.r4.8xlarge	218255432090	21825543209	21825543209
cache.r4.16xlarge	437021573120	43702157312	43702157312
cache.r5.large	14037181030	1403718103	1403718103
cache.r5.xlarge	28261849702	2826184970	2826184970
cache.r5.2xlarge	56711183565	5671118356	5671118356
cache.r5.4xlarge	113609865216	11360986522	11360986522
cache.r5.12xlarge	341206346547	34120634655	34120634655
cache.r5.24xlarge	682485973811	68248597381	68248597381

Note

현재 세대의 모든 인스턴스 유형은 기본적으로 Amazon Virtual Private Cloud VPC에서 생성됩니다. T1 인스턴스는 자동 장애 조치가 포함된 다중 AZ를 지원하지 않습니다.

T1 및 T2 인스턴스는 Redis AOF를 지원하지 않습니다.

Redis 구성 변수 appendonly 및 appendfsync는 Redis 버전 2.8.22 이상에서 지원되지 않습니다.

Amazon ElastiCache의 보안

AWS에서는 클라우드 보안을 가장 중요하게 생각합니다. AWS 고객은 보안에 매우 보안에 민감한 조직의 요구 사항에 부합하도록 구축된 데이터 센터 및 네트워크 아키텍처의 혜택을 누릴 수 있습니다.

보안은 AWS와 귀하의 공동 책임입니다. 책임 분담 모델은 이 사항을 클라우드 내 보안 및 클라우드의 보안으로 설명합니다.

- 클라우드의 보안 AWS는 AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호합니다. AWS는 또한 안전하게 사용할 수 있는 서비스를 제공합니다. 타사 감사원은 정기적으로 AWS 규제 준수 프로그램의 일환으로 보안 효과를 테스트하고 검증합니다. Amazon ElastiCache에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 AWS규정 준수 프로그램 제공 범위 내 서비스를 참조하십시오.
- 클라우드 내 보안 귀하의 책임은 귀하가 사용하는 AWS 서비스에 의해 결정됩니다. 또한 귀하는 데이터의 민감도, 회사 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon ElastiCache 사용 시 책임 분담 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 항목에서는 보안 및 규정 준수 목표를 충족하도록 Amazon ElastiCache를 구성하는 방법을 보여줍니다. 또한 Amazon ElastiCache 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 배우게 됩니다.

주제

- Amazon ElastiCache의 데이터 보호 (p. 298)
- 인터네트워크 트래픽 개인 정보 (p. 315)
- Amazon ElastiCache의 Identity and Access Management(IAM) (p. 341)
- Elasticache에서 로깅 및 모니터링 (p. 359)
- Amazon ElastiCache 규정 준수 확인 (p. 391)
- Amazon ElastiCache의 복원성 (p. 395)
- AWS Elasticache의 인프라 보안 (p. 398)

Amazon ElastiCache의 데이터 보호

Amazon ElastiCache는 AWS 공동 책임 모델을 준수하며, 여기에는 데이터 보호 관련 규정 및 지침이 포함됩니다. AWS는 모든 AWS 서비스를 실행하는 글로벌 인프라를 보호할 책임이 있습니다. AWS는 고객 콘텐츠 및 개인 데이터의 처리를 위한 보안 구성 제어 등 이 인프라에서 호스팅되는 데이터에 대한 제어권을 유지합니다. 데이터 컨트롤러 또는 데이터 처리자의 역할을 담당하는 AWS 고객과 APN 파트너는 AWS Cloud에 올린 모든 개인 데이터에 대해 책임을 집니다.

데이터를 보호하려면 AWS 계정 자격 증명을 보호하고 AWS Identity and Access Management(IAM)을 사용해 개별 사용자 계정을 설정하는 것이 좋습니다. 이러한 접근 방식에서는 각 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정마다 멀티 팩터 인증(MFA)을 사용합니다.
- SSL/TLS를 사용하여 AWS 리소스와 통신합니다.
- AWS CloudTrail로 API 및 사용자 활동 로깅을 설정합니다.
- AWS 암호화 솔루션을 AWS 서비스 내의 모든 기본 보안 컨트롤과 함께 사용합니다.
- Amazon S3에 저장된 개인 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용합니다.

이름 필드와 같은 자유 형식 필드에 고객 계정 번호와 같은 중요 식별 정보를 절대 입력하지 마십시오. 여기에는 Amazon ElastiCache 또는 기타 AWS 서비스에서 콘솔, API, AWS CLI 또는 AWS SDK를 사용하여 작업하는 경우가 포함됩니다. Amazon ElastiCache 또는 기타 서비스에 입력하는 모든 데이터는 진단 로그에 포함하기 위해 선택될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해자격 증명 정보를 URL에 포함시키지 마십시오.

데이터 보호에 관한 자세한 내용은 AWS 보안 블로그에서 AWS 공동 책임 모델 및 GDPR 블로그 게시물을 참조하십시오.

주제

• Amazon ElastiCache의 데이터 보안 (p. 299)

Amazon ElastiCache의 데이터 보안

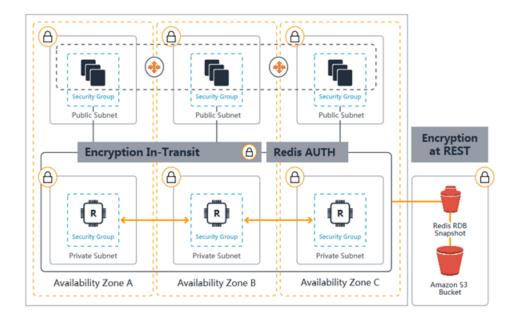
데이터를 안전하게 보관하기 위해 Amazon ElastiCache 및 Amazon EC2에서는 서버의 데이터에 대한 무단액세스를 방지하는 메커니즘을 제공합니다.

Redis용 Amazon ElastiCache는 Redis 버전 3.2.6, 4.0.10 이상을 실행하는 클러스터에서 데이터를 위한 다음 선택적 암호화 기능을 제공합니다.

- 전송 중 데이터 암호화는 클러스터 내 노드 사이 또는 클러스터와 애플리케이션 사이와 같이 한 위치에서 다른 위치로 데이터가 이동 중인 경우 항상 데이터를 암호화합니다.
- 미사용 데이터 암호화는 동기화 및 백업 작업 중 디스크 내 데이터를 암호화합니다.

전송 중 데이터 암호화 또는 미사용 데이터 암호화를 활성화하려면 다음 조건을 충족해야 합니다.

- 클러스터 또는 복제 그룹에서 Redis 3.2.6, 4.0.10 이상 버전을 실행 중이어야 합니다.
- 클러스터 또는 복제 그룹은 Amazon VPC 기반 VPC에 생성되어야 합니다.
- 선택 사항으로 이 클러스터 또는 복제 그룹에서 작업을 수행하는 데 필요한 AUTH 및 AUTH 토큰(암호)을 사용할 수도 있습니다.



Redis용 ElastiCache 보안 다이어그램

주제

- Redis용 ElastiCache 전송 중 데이터 암호화(TLS) (p. 300)
- Redis용 ElastiCache에서 미사용 데이터 암호화 (p. 306)
- Redis AUTH 명령을 사용하여 사용자 인증 (p. 313)

Redis용 ElastiCache 전송 중 데이터 암호화(TLS)

데이터를 안전하게 보관하기 위해 Amazon ElastiCache 및 Amazon EC2에서는 서버의 데이터에 대한 무단액세스를 방지하는 메커니즘을 제공합니다. ElastiCache에서는 전송 중 데이터 암호화 기능을 제공해 한 위치에서 다른 위치로 데이터 이동 시 데이터를 보호할 수 있는 수단을 제공합니다. 예를 들어 복제 그룹 내 기본 노드에서 읽기 전용 복제본 노드로 또는 복제 그룹과 애플리케이션 사이에서 데이터를 이동할 수 있습니다.

전송 중 데이터 암호화는 선택 사항으로, Redis 복제 그룹을 생성할 때에만 해당 복제 그룹에 대해 활성화할수 있습니다. 복제 그룹을 생성할 때 파라미터 TransitEncryptionEnabled를 true(CLI: --transitencryption-enabled)로 설정해 복제 그룹에 대해 전송 중 데이터 암호화를 활성화합니다. AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 복제 그룹을 생성하든 상관 없이 전송 중 데이터 암호화를 활성화할 수 있습니다. 전송 중 데이터 암호화를 활성화하는 경우 CacheSubnetGroup의 값도 제공해야 합니다.

Important

파라미터 TransitEncryptionEnabled(CLI: --transit-encryption-enabled)는 CreateReplicationGroup(CLI: create-replication-group) 작업을 사용하는 경우에만 사용할 수 있습니다.

주제

- 전송 중 데이터 암호화 개요 (p. 300)
- 전송 중 데이터 암호화 조건 (p. 300)
- 전송 중 데이터 암호화 활성화 (p. 301)
- redis-cli를 사용하여 전송 중 데이터 암호화가 활성화된 Redis용 Amazon ElastiCache 노드에 연결 (p. 304)
- 참고 항목 (p. 306)

전송 중 데이터 암호화 개요

Amazon ElastiCache의 전송 중 데이터 암호화는 선택적 기능으로, 가장 취약한 지점— 즉, 한 위치에서 다른 위치로 데이터를 전송할 때 데이터의 보안을 강화합니다. 엔드포인트에서 데이터를 암호화 및 해독하기 위해 몇 가지 처리가 필요하기 때문에 전송 중 데이터 암호화를 활성화하면 성능에 어느 정도 영향이 있을 수 있습니다. 사용 사례에 대한 성능 영향을 파악하기 위해서는 전송 중 데이터 암호화를 사용한 상태와 사용하지 않은 상태에서 데이터를 벤치마크해야 합니다.

ElastiCache의 전송 중 데이터 암호화는 다음 기능을 구현합니다.

- 암호화된 연결—서버와 클라이언트 연결이 둘 다 암호화된 SSL(Secure Socket Layer)입니다.
- 암호화된 복제—기본 노드와 복제본 노드 간에 이동하는 데이터가 암호화됩니다.
- 서버 인증—클라이언트가 자신이 올바른 서버에 연결 중임을 인증할 수 있습니다.
- 클라이언트 인증—Redis AUTH 기능을 사용하여 서버가 클라이언트를 인증할 수 있습니다.

전송 중 데이터 암호화 조건

Amazon ElastiCache의 전송 중 데이터 암호화를 구현하기 위해 계획하는 경우에는 다음 제약을 염두에 둬야합니다.

- 전송 중 데이터 암호화는 Redis 3.2.6, 4.0.10 및 이후 버전을 실행 중인 복제 그룹에서만 지원됩니다.
- 전송 중 데이터 암호화는 Amazon VPC에서 실행 중인 복제 그룹에 대해서만 지원됩니다.
- 전송 중 데이터 암호화는 다음 노드 유형을 실행하는 복제 그룹에 대해서만 지원됩니다.
 - R5, R4, R3
 - M5, M4, M3
 - T2

자세한 내용은 지원되는 노드 유형 (p. 54) 단원을 참조하십시오.

- 전송 중 데이터 암호화는 TransitEncryptionEnabled 파라미터를 명시적으로 true로 설정해 활성화합니다.
- 복제 그룹을 생성하는 경우에만 복제 그룹에 대해 전송 중 데이터 암호화를 활성화할 수 있습니다. 복제 그룹을 수정하여 전송 중 데이터 암호화 켜기 및 끄기를 전환할 수 없습니다. 기존 복제 그룹에 대해 전송 중 데이터 암호화를 구현하는 자세한 내용은 전송 중 데이터 암호화 활성화 (p. 301) 단원을 참조하십시오.
- 전송 중 데이터 암호화가 활성화된 복제 그룹에 연결하려면 데이터베이스에서 TLS(전송 계층 보안)가 활성화되어 있어야 합니다. 전송 중 데이터 암호화가 활성화되지 않은 복제 그룹에 연결하려면 데이터베이스에서 TLS를 활성화할 필요가 없습니다.

엔드포인트에서 데이터를 암호화 및 해독하기 위해 처리가 필요하기 때문에 전송 중 데이터 암호화를 구현하면 성능이 저하될 수 있습니다. 이러한 암호화가 구현 성능에 미치는 영향을 확인하려면 전송 중 데이터 암호화와 데이터를 암호화하지 않은 경우를 비교해 벤치마크하십시오.

Tip

새로운 연결을 생성하면 비용이 높아질 수 있기 때문에 SSL 연결을 지속해 전송 중 암호화가 성능에 미치는 영향을 줄일 수 있습니다.

전송 중 데이터 암호화 활성화

AWS Management 콘솔, AWS CLI 또는ElastiCache API를 사용하여 Redis용 ElastiCache 복제 그룹을 생성하는 경우 전송 중 데이터 암호화를 활성화할 수 있습니다.

기존 클러스터에 대해 전송 중 데이터 암호화 활성화

Redis 복제 그룹을 생성할 때에만 전송 중 데이터 암호화를 활성화할 수 있습니다. 전송 중 데이터 암호화를 활성화하려는 기존 복제 그룹이 있으면 다음 작업을 수행하십시오.

기존 Redis 복제 그룹에 대해 전송 중 데이터 암호화를 활성화하려면

- 1. 복제 그룹의 백업을 수동으로 생성합니다. 자세한 내용은 수동 백업 만들기 (p. 190) 단원을 참조하십시오.
- 2. 백업 설정에서 엔진 버전을 3.2.6, 4.0.10 및 이후 버전으로, 파라미터 TransitEncryptionEnabled를 true(CLI: --transit-encryption-enabled)로 복원해 새 복제 그룹을 생성합니다. 자세한 내용은 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.
- 3. 애플리케이션의 엔드포인트를 새 복제 그룹의 엔드포인트로 업데이트합니다. 자세한 내용은 연결 엔드 포인트 찾기 (p. 180) 단원을 참조하십시오.
- 4. 이전 복제 그룹을 삭제합니다. 자세한 내용은 다음을 참조하십시오.
 - 클러스터 삭제하기 (p. 102)
 - 복제 그룹 삭제 (p. 166)

AWS Management 콘솔를 사용하여 전송 중 데이터 암호화 활성화

AWS Management 콘솔을 사용하여 복제 그룹을 생성하는 경우 전송 중 데이터 암호화를 활성화하려면 다음과 같이 선택합니다.

- 엔진으로 Redis를 선택합니다.
- 다음 엔진 버전을 선택합니다. 3.2.6, 4.0.10 또는 이후 버전.
- [Encryption in-transit] 목록에서 [Yes]를 선택합니다.

단계별 프로세스의 경우 다음을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69)
- Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 72)

AWS CLI를 사용하여 전송 중 데이터 암호화 활성화

AWS CLI를 사용하여 Redis 복제 그룹을 생성할 때 전송 중 데이터 암호화를 활성화하려면 파라미터 transit-encryption-enabled를 설정합니다.

Redis(클러스터 모드 비활성화됨) 클러스터에 대해 전송 중 데이터 암호화 활성화(CLI)

AWS CLI 작업 create-replication-group 및 다음 파라미터를 사용하여 전송 중 데이터 암호화가 활성화된 복제본이 있는 Redis 복제 그룹을 생성합니다.

키 파라미터:

- --engine—반드시 redis여야 합니다.
- --engine-version—반드시 3.2.6, 4.0.10 또는 이상이어야 합니다.
- --transit-encryption-enabled—필수. 전송 중 데이터 암호화를 활성화하면 --cache-subnetgroup 파라미터의 값도 제공해야 합니다.
- --num-cache-clusters— 1 이상이어야 함. 이 파라미터의 최대값은 six입니다.

자세한 내용은 다음을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 149)
- create-replication-group

Redis(클러스터 모드 활성화됨)에 대한 클러스터에 대해 전송 중 데이터 암호화 활성화(CLI)

AWS CLI 작업 create-replication-group 및 다음 파라미터를 사용하여 전송 중 데이터 암호화가 활성화된 Redis(클러스터 모드 활성화됨) 복제 그룹을 생성합니다.

키 파라미터:

- --engine—반드시 redis여야 합니다.
- --engine-version—반드시 3.2.6, 4.0.10 또는 이상이어야 합니다.
- --transit-encryption-enabled—필수. 전송 중 데이터 암호화를 활성화하면 --cache-subnet-group 파라미터의 값도 제공해야 합니다.
- 다음 파라미터 세트 중 하나를 사용해 복제 그룹의 노드 그룹 구성을 지정합니다.

- --num-node-groups—복제 그룹의 샤드(노드 그룹) 수를 지정합니다. 이 파라미터의 최대값은 90입니다.
 - --replicas-per-node-group—각 노드 그룹(샤드) 내 복제 노드 수입니다. 여기서 지정된 값은 복제 그룹의 모든 샤드에 적용됩니다. 이 파라미터의 최대값은 5입니다.
- --node-group-configuration—각 샤드의 구성을 독립적으로 지정합니다.

자세한 내용은 다음 자료를 참조하십시오.

- Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154)
- · create-replication-group

AWS API를 사용하여 전송 중 데이터 암호화 활성화

ElastiCache API를 사용하여 Redis 복제 그룹을 생성할 때 전송 중 데이터 암호화를 활성화하려면 TransitEncryptionEnabled 파라미터를 true로 설정하고 단일 노드 Redis 복제 그룹의 경우에는 CreateCacheCluster를, 읽기 전용 복제본이 있는 복제 그룹에 대해서는 CreateReplicationGroup을 사용합니다.

Redis(클러스터 모드 비활성화됨)에 대한 클러스터에 대해 전송 중 데이터 암호화 활성화(API)

ElastiCache API 작업 CreateReplicationGroup 및 다음 파라미터를 사용하여 전송 중 데이터 암호화가 활성화된 Redis(클러스터 모드 비활성화됨) 복제 그룹을 생성합니다.

키 파라미터

- Engine—반드시 redis여야 합니다.
- EngineVersion—반드시 3.2.6, 4.0.10 또는 이상이어야 합니다.
- TransitEncryptionEnabled—true으로 설정해야 합니다.

TransitEncryptionEnabled를 true로 설정하면 CacheSubnetGroup의 값도 제공해야 합니다.

• NumCacheClusters— 1 이상이어야 함. 이 파라미터의 최대값은 six입니다.

자세한 내용은 다음을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성 (p. 151)
- CreateReplicationGroup

Redis(클러스터 모드 활성화됨)에 대한 클러스터에 대해 전송 중 데이터 암호화 활성화(API)

ElastiCache API 작업 CreateReplicationGroup 및 다음 파라미터를 사용하여 전송 중 데이터 암호화가 활성화된 Redis(클러스터 모드 활성화됨) 복제 그룹을 생성합니다.

키 파라미터

- Engine—반드시 redis여야 합니다.
- EngineVersion—반드시 3.2.6. 4.0.10 또는 이상이어야 합니다.
- TransitEncryptionEnabled—true으로 설정해야 합니다.

TransitEncryptionEnabled를 true로 설정하면 CacheSubnetGroup의 값도 제공해야 합니다.

- 다음 파라미터 세트 중 하나를 사용해 복제 그룹의 노드 그룹 구성을 지정합니다.
 - NumNodeGroups—복제 그룹의 샤드(노드 그룹) 수를 지정합니다. 이 파라미터의 최대 값은 90이지만 서비스 제한 증가 요청을 통해 최대 250까지 증가할 수 있습니다. 자세한 내용은 AWS 서비스 제한을 참 조하십시오.

ReplicasPerNodeGroup—각 노드 그룹(샤드) 내 복제 노드 수입니다. 여기서 지정된 값은 복제 그룹의 모든 샤드에 적용됩니다. 이 파라미터의 최대값은 5입니다.

• NodeGroupConfiguration—각 샤드의 구성을 독립적으로 지정합니다.

자세한 내용은 다음 자료를 참조하십시오.

- Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158)
- CreateReplicationGroup

redis-cli를 사용하여 전송 중 데이터 암호화가 활성화된 Redis용 Amazon ElastiCache 노드에 연결

전송 중 데이터 암호화가 활성화된 Redis 노드용 ElastiCache에서 데이터에 액세스하려면 Secure Sockets Layer(SSL) 작업을 하는 클라이언트를 사용해야 합니다. 그러나 redis-cli는 SSL이나 TLS(전송 계층 보안)를 지원하지 않습니다.

이를 해결하기 위해 stunne1 명령을 사용해 Redis 노드에 SSL 터널을 생성할 수 있습니다. 그런 다음 암호 화된 Redis 노드에서 데이터에 액세스하기 위해 redis-cli를 사용하여 터널에 연결합니다.

redis-cli를 사용하여 전송 중 데이터 암호화가 활성화된 Redis 클러스터에 연결하려면

1. SSH 클라이언트에서 stunnel을 설치합니다.

```
sudo yum install stunnel
```

2. 파일의 생성 및 편집을 위한 다음 코드 '/etc/stunnel/redis-cli.conf'를 동시에 실행하여 Redis용 ElastiCache 클러스터 엔드포인트를 아래 템플릿과 같이 제공된 출력으로 한 개 이상의 연결 매개 변수에 추가하십시오.

```
cat /etc/stunnel/redis-cli.conf
fips = no
setuid = root
setgid = root
pid = /var/run/stunnel.pid
debug = 7
delay = yes
options = NO SSLv2
options = NO_SSLv3
[redis-cli]
  client = ves
   accept = 127.0.0.1:6379
   connect = master.ssltest.wif01h.use1.cache.amazonaws.com:6379
[redis-cli-replica]
  client = yes
   accept = 127.0.0.1:6380
   connect = ssltest-02.ssltest.wif01h.use1.cache.amazonaws.com:6379
```

이 예에서는 구성 파일에 두 가지 연결, 즉 redis-cli와 redis-cli-replica가 있습니다. 파라미터 는 다음과 같이 설정되어 있습니다.

• client는 이 stunnel 인스턴스를 클라이언트로 지정하기 위해 yes로 설정되어 있습니다.

- accept는 클라이언트 IP로 설정되어 있습니다. 이 예에서는 master가 Redis 기본인 포트 6379의 127.0.0.1로 설정되어 있습니다. 복제본은 다른 포트를 호출해야 하며 6380으로 설정되어야 합니다. 휘발성 포트 1024–65535를 사용할 수 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 휘발성 포트를 참조하십시오.
- connect는 Redis 서버 엔드포인트로 설정되어 있습니다. 자세한 내용은 연결 엔드포인트 찾기 (p. 180) 단원을 참조하십시오.
- 3. Start stunnel.

```
sudo stunnel /etc/stunnel/redis-cli.conf
```

netstat 명령을 사용하여 터널이 시작되었는지 확인합니다.

```
      sudo netstat -tulnp | grep -i stunnel

      tcp 0 0 127.0.0.1:6379 0.0.0.0:*
      LISTEN

      3189/stunnel
      0 0 127.0.0.1:6380 0.0.0.0:*
      LISTEN

      3189/stunnel
      0 0 127.0.0.1:6380 0.0.0.0:*
      LISTEN
```

- 4. 터널의 로컬 엔드포인트를 사용하여 암호화된 Redis 노드에 연결합니다.
 - Redis용 ElastiCache 클러스터 생성 중 AUTH 암호를 사용하지 않은 경우, 이 예에서는 Amazon Linux 에서 redis-cli에 대한 완전한 경로로 Redis용 ElastiCache 서버에 연결하기 위해 redis-cli를 사용합니다.

```
/home/ec2-user/redis-stable/src/redis-cli -h localhost -p 6379
```

Redis 클러스터 생성 중 AUTH 암호를 사용한 경우, 이 예에서는 Amazon Linux에서 redis-cli에 대한 완전한 경로로 Redis 서버에 연결하기 위해 redis-cli를 사용합니다.

```
/home/ec2-user/redis-stable/src/redis-cli -h localhost -p 6379 -a my-secret-password
```

또는

• redis-stable로 디렉토리를 변경하고 다음을 수행합니다.

Redis용 ElastiCache 클러스터 생성 중 AUTH 암호를 사용하지 않은 경우, 이 예에서는 Amazon Linux 에서 redis-cli에 대한 완전한 경로로 Redis용 ElastiCache 서버에 연결하기 위해 redis-cli를 사용합니다.

```
src/redis-cli -h localhost -p 6379
```

Redis 클러스터 생성 중 AUTH 암호를 사용한 경우, 이 예에서는 Amazon Linux에서 redis-cli에 대한 완전한 경로로 Redis 서버에 연결하기 위해 redis-cli를 사용합니다.

```
src/redis-cli -h localhost -p 6379 -a my-secret-password
```

이 예에서는 Telnet을 사용하여 Redis 서버에 연결합니다.

```
telnet localhost 6379

Trying 127.0.0.1...
Connected to localhost.
```

Escape character is '^]'.
auth MySecretPassword
+OK
get foo
\$3
bar

5. SSL 터널을 중단하고 닫으려면 stunnel 프로세스를 pkill합니다.

sudo pkill stunnel

참고 항목

- Redis용 ElastiCache에서 미사용 데이터 암호화 (p. 306)
- Redis AUTH 명령을 사용하여 사용자 인증 (p. 313)
- Amazon VPC 및 ElastiCache 보안 (p. 316)
- Amazon ElastiCache의 Identity and Access Management(IAM) (p. 341)

Redis용 ElastiCache에서 미사용 데이터 암호화

데이터를 안전하게 보관하기 위해 Amazon ElastiCache 및 Amazon S3에서는 캐시의 데이터에 대한 액 세스를 제한하는 다른 방식을 제공합니다. 자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 및 Amazon ElastiCache의 Identity and Access Management(IAM) (p. 341) 단원을 참조하십시오.

Redis용 ElastiCache의 미사용 데이터 암호화는 선택적 기능으로, 디스크의 데이터를 암호화해 데이터 보안을 강화합니다. 복제 그룹에서 사용하도록 활성화하면 다음 측면이 암호화됩니다.

- 동기화, 백업 및 스왑 작업 중 디스크
- Amazon S3에 저장된 백업

Redis용 ElastiCache는 기본(서비스 관리형) 유휴 데이터 암호화와 더불어 AWS Key Management Service(KMS)에서 자체 대칭 고객 관리형 고객 마스터 키를 사용할 수 있는 기능을 제공합니다.

미사용 데이터 암호화는 복제 그룹을 생성할 때에만 복제 그룹에 대해 활성화할 수 있습니다. 데이터를 암호화 및 해독하기 위해 몇 가지 처리가 필요하기 때문에 미사용 데이터 암호화를 활성화하면 이러한 작업 중 성능에 영향이 있을 수 있습니다. 사용 사례에 대한 성능 영향을 파악하기 위해서는 미사용 데이터 암호화를 사용한 상태와 사용하지 않은 상태에서 데이터를 벤치마크해야 합니다.

전송 중 데이터 암호화와 관련된 자세한 내용은 Redis용 ElastiCache 전송 중 데이터 암호화 (TLS) (p. 300)을 참조하십시오.

주제

- 미사용 데이터 암호화 조건 (p. 306)
- AWS KMS에서 고객 관리형 CMK 사용 (p. 307)
- 미사용 데이터 암호화 활성화 (p. 308)
- 참고 항목 (p. 312)

미사용 데이터 암호화 조건

ElastiCache의 미사용 데이터 암호화를 구현하기 위해 계획하는 경우에는 ElastiCache의 미사용 데이터 암 호화에 대한 다음 제약을 염두에 둬야 합니다.

- 미사용 데이터 암호화는 Redis 버전 3.2.6, 4.0.10 또는 이후 버전을 실행 중인 복제 그룹에서만 지원됩니다.
- 유휴 데이터 암호화는 Amazon VPC에서 실행 중인 복제 그룹에 대해서만 지원됩니다.
- 유휴 데이터 암호화는 다음 노드 유형을 실행하는 복제 그룹에 대해서만 지원됩니다.
 - R5, R4, R3
 - M5. M4. M3
 - T2

자세한 내용은 지원되는 노드 유형 (p. 54) 단원을 참조하십시오.

- 미사용 데이터 암호화는 AtRestEncryptionEnabled 파라미터를 명시적으로 true로 설정해 활성화합니다
- 복제 그룹을 생성하는 경우에만 복제 그룹에 대해 미사용 데이터 암호화를 활성화할 수 있습니다. 복제 그룹을 수정하는 방법으로는 미사용 데이터 암호화 켜기 또는 끄기로 전환할 수 없습니다. 기존 복제 그룹에 대해 미사용 데이터 암호화를 구현하는 방법에 대한 자세한 내용은 미사용 데이터 암호화 활성화 (p. 308) 단원을 참조하십시오.
- 미사용 데이터 암호화는 cn-north-1(베이징) 및 cn-northwest-1(닝샤). ap-northeast-3 (아시아 태평양(오사 카-로컬) 리전에서 사용할 수 없습니다. 또한 AWS GovCloud(us-gov-east-1 및 us-gov-west-1) 리전에서는 미사용 데이터 암호화를 위한 고객 관리형 CMK를 사용할 수 없습니다.

미사용 데이터 암호화를 구현하면 백업 및 노드 동기화 작업 중 성능이 저하될 수 있습니다. 이러한 암호화가 구현 성능에 미치는 영향을 확인하려면 미사용 데이터 암호화와 데이터를 암호화하지 않은 경우를 비교해 벤치마크하십시오.

AWS KMS에서 고객 관리형 CMK 사용

Redis용 ElastiCache에서는 유휴 데이터 암호화에 대한 대칭 고객 관리형 고객 마스터 키(CMK)를 지원합니다. 고객 관리형 CMK는 사용자가 생성, 소유 및 관리하는 AWS 계정의 암호화 키입니다. 자세한 내용은 AWS Key Management Service 개발자 안내서의 고객 마스터 키를 참조하십시오. Elasticache와 함께 사용하기 전에 AWS KMS에서 키를 생성해야 합니다.

AWS KMS 마스터 키 생성 방법에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 키 생성을 참조하십시오.

Redis용 ElastiCache를 사용하면 AWS KMS와 통합할 수 있습니다. 자세한 내용은 AWS Key Management Service 개발자 안내서의 그랜트 사용을 참조하십시오. Amazon ElastiCache와 AWS KMS의 통합을 활성화하기 위해 고객이 별도로 취해야 할 조치는 없습니다.

Amazon ElastiCache에서는 현재 kms:ViaService를 지원하지 않습니다. ViaService를 사용하여 Amazon ElastiCache에 대한 액세스를 제공/거부해도 키 권한에는 영향을 미치지 않습니다.

AWS CloudTrail을 사용하여 Amazon ElastiCache가 사용자 대신 AWS Key Management Service에 전송하는 요청을 추적할 수 있습니다. 고객 관리형 CMK와 관련된 AWS Key Management Service에 대한 모든 API 호출에는 해당 CloudTrail 로그가 있습니다. ListGlants KMS API 호출을 통해 ElastiCache가 생성하는 그랜트를 볼 수도 있습니다.

고객 관리형 CMK를 사용하여 복제 그룹을 암호화하면 복제 그룹에 대한 모든 백업이 다음과 같이 암호화됩니다.

- 자동 일일 백업은 클러스터와 연결된 고객 관리형 CMK를 사용하여 암호화됩니다.
- 복제 그룹을 삭제할 때 생성된 최종 백업은 복제 그룹에 연결된 고객 관리형 CMK를 사용하여 암호화됩니다.
- 수동으로 생성한 백업은 복제 그룹에 연결된 CMK를 사용하기 위해 기본적으로 암호화됩니다. 다른 고객 관리형 CMK를 선택하여 이를 재정의할 수 있습니다.
- 백업 복사는 소스 백업과 연결된 고객 관리형 CMK를 사용하는 것을 디폴트로 합니다. 다른 고객 관리형 CMK를 선택하여 이를 재정의할 수 있습니다.

Note

- 선택한 Amazon S3 버킷으로 백업을 내보낼 때 고객 관리형 CMK를 사용할 수 없습니다. 그러나 Amazon S3으로 내보낸 모든 백업은 서버 측 암호화를 사용하여 암호화됩니다. 백업 파일을 새 S3 개체로 복사하고 고객 관리형 CMK를 사용하여 암호화하거나, CMK를 사용하여 기본 암호화로 설정된 다른 S3 버킷에 파일을 복사하거나, 파일 자체에서 암호화 옵션을 변경할 수 있습니다.
- 또한 고객 관리형 CMK를 사용하여 암호화에 고객 관리형 CMK를 사용하지 않는 복제 그룹에 대해 수동으로 생성한 백업을 암호화할 수도 있습니다. 이 옵션을 사용하면 원래 복제 그룹에서 데이터가 암호화되지 않더라도 CMK를 사용하여 Amazon S3에 저장된 백업 파일이 암호화됩니다.

백업에서 복원하면 새 복제 그룹을 생성할 때 사용할 수 있는 암호화 옵션과 유사한 암호화 옵션을 선택할 수 있습니다. 고려 사항:

- 복제 그룹을 암호화하는 데 사용한 키에 대해 키를 삭제하거나 키를 비활성화하고 그랜트를 취소하면 복제 그룹을 복구할 수 없게 됩니다. 즉, 하드웨어 장애 후 이를 수정하거나 복구할 수 없습니다. AWS KMS에서는 최소 7일 이상의 대기 기간이 지난 후에만 마스터 키를 삭제합니다. 키를 삭제한 후 다른 고객 관리형 CMK를 사용하여 보관용 백업을 생성할 수 있습니다.
- 자동 키 교체 기능은 AWS KMS 마스터 키의 속성을 그대로 보존하기 때문에 키가 교체되더라도 ElastiCache 데이터에 대한 액세스 권한에는 아무런 영향도 끼치지 않습니다. 암호화된 Amazon ElastiCache 복제 그룹은 새로운 마스터 키를 생성하거나 이전 키에 대한 모든 참조를 업데이트하는 수동 키 교체를 지원하지 않습니다. 자세한 내용은 AWS Key Management Service 개발자 안내서의 고객 마스터 키 교체입니다.
- CMK를 사용하여 ElastiCache 복제 그룹을 암호화하려면 복제 그룹당 1개의 그랜트가 필요합니다. 이 그랜트는 복제 그룹의 수명 동안 사용됩니다. 또한 백업 생성 중에는 백업당 하나의 그랜트가 사용됩니다. 이 그랜트는 백업이 생성되면 폐기됩니다.
- AWS KMS 그랜트와 제한에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 제한을 참조하십시오.

미사용 데이터 암호화 활성화

파라미터 AtrestEncryptionEnabled를 true로 설정하여 Redis 복제 그룹을 생성할 때 ElastiCache의 미사용 데이터 암호화를 활성화할 수 있습니다. 기존 복제 그룹에 대해서는 미사용 데이터 암호화를 설정할수 없습니다.

Redis용 ElastiCache 복제 그룹을 생성할 때 미사용 데이터 암호화를 활성화할 수 있습니다. AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용해 늘릴 수 있습니다.

복제 그룹을 생성할 때. 다음 옵션 중 하나를 선택할 수 있습니다.

- 기본값 이 옵션은 미사용 서비스 관리 암호화를 사용합니다.
- 고객 관리형 CMK 이 옵션을 통해 미사용 데이터 암호화에 대한 AWS KMS의 키 ID/ARN을 제공합니다.

AWS KMS 마스터 키를 생성 방법에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 키 생성을 참조하십시오.

목차

- 기존 Redis 클러스터에 대해 미사용 데이터 암호화 활성화 (p. 309)
- AWS Management 콘솔를 사용하여 미사용 데이터 암호화 활성화 (p. 309)
- AWS CLI를 사용하여 미사용 데이터 암호화 활성화 (p. 309)
 - Redis(클러스터 모드 비활성화됨) 클러스터에 대해 미사용 데이터 암호화 활성화 (CLI) (p. 309)
 - Redis(클러스터 모드 활성화됨)에 대한 클러스터에서 미사용 데이터 암호화 활성화 (CLI) (p. 310)

- ElastiCache API를 사용하여 미사용 데이터 암호화 활성화 (p. 311)
 - Redis(클러스터 모드 비활성화됨) 클러스터에서 미사용 데이터 암호화 활성화(API) (p. 311)
 - Redis(클러스터 모드 활성화됨)에 대한 클러스터에서 미사용 데이터 암호화 활성화 (API) (p. 312)

기존 Redis 클러스터에 대해 미사용 데이터 암호화 활성화

Redis 복제 그룹을 생성할 때에만 미사용 데이터 암호화를 활성화할 수 있습니다. 미사용 데이터 암호화를 활성화하려는 기존 복제 그룹이 있으면 다음 작업을 수행하십시오.

기존 복제 그룹에 대해 미사용 데이터 암호화를 활성화하려면

- 1. 기존 복제 그룹의 백업을 수동으로 생성합니다. 자세한 내용은 수동 백업 만들기 (p. 190) 단원을 참조하십시오.
- 백업에서 복원하여 새 복제 그룹을 생성합니다. 새 복제 그룹에 대해 미사용 데이터 암호화를 활성화합니다. 자세한 내용은 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.
- 3. 새 복제 그룹을 가리키도록 애플리케이션에서 엔드포인트를 업데이트합니다.
- 4. 이전 복제 그룹을 삭제합니다. 자세한 내용은 클러스터 삭제하기 (p. 102) 또는 복제 그룹 삭제 (p. 166) 단원을 참조하십시오.

AWS Management 콘솔를 사용하여 미사용 데이터 암호화 활성화

AWS Management 콘솔을 사용하여 복제 그룹을 생성하는 경우 미사용 데이터 암호화를 활성화하려면 다음과 같이 선택합니다.

- 엔진으로 redis를 선택합니다.
- 엔진 버전으로 버전 3.2.6, 4.0.10 또는 이후 버전을 선택합니다.
- [Encryption at-rest] 목록에서 [Yes]를 선택합니다.

단계별 절차의 경우 다음을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69)
- Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 72)

AWS CLI를 사용하여 미사용 데이터 암호화 활성화

AWS CLI를 사용하여 Redis 클러스터를 생성하는 경우 미사용 데이터 암호화를 활성화하려면 복제 그룹을 생성할 때 --at-rest-encryption-enabled 파라미터를 사용합니다.

Redis(클러스터 모드 비활성화됨) 클러스터에 대해 미사용 데이터 암호화 활성화(CLI)

다음 작업은 세 개의 노드(--num-cache-clusters) 즉, 기본 복제본 한 개와 읽기 전용 복제본 두 개가 있는 Redis(클러스터 모드 비활성화됨) 복제 그룹 my-classic-rg을 생성합니다. 미사용 데이터 암호화가 이 복제 그룹에 대해 활성화되어 있습니다(--at-rest-encryption-enabled).

다음 파라미터와 해당 값은 복제 그룹에 대한 암호화를 활성화하는 데 필요합니다.

키 파라미터

- --engine—반드시 redis여야 합니다.
- --engine-version—반드시 3.2.6, 4.0.10 또는 이상이어야 합니다.
- --at-rest-encryption-enabled—유휴 시 암호화를 활성화하기 위해 필요합니다.

Example 1: 복제본이 있는 Redis(클러스터 모드 비활성화됨) 클러스터

Linux, macOS, or Unix의 경우:

```
aws elasticache create-replication-group \
--replication-group-id my-classic-rg \
--replication-group-description "3 node replication group" \
--cache-node-type cache.m4.large \
--engine redis \
--engine-version 4.0.10 \
--at-rest-encryption-enabled \
--num-cache-clusters 3 \
--cache-parameter-group default.redis4.0
```

Windows의 경우:

추가 정보는 다음을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 149)
- · create-replication-group

Redis(클러스터 모드 활성화됨)에 대한 클러스터에서 미사용 데이터 암호화 활성화(CLI)

다음 작업은 3개의 노드 그룹 또는 샤드(--num-node-groups)가 있는 Redis(클러스터 모드 활성화됨) 복제 그룹 my-clustered-rg을 생성합니다. 각 복제 그룹에는 기본 복제본 한 개와 읽기 전용 복제본 두 개, 이렇게 세 개의 노드가 있습니다(--replicas-per-node-group). 미사용 데이터 암호화가 이 복제 그룹에 대해 활성화되어 있습니다(--at-rest-encryption-enabled).

다음 파라미터와 해당 값은 복제 그룹에 대한 암호화를 활성화하는 데 필요합니다.

키 파라미터

- --engine—반드시 redis여야 합니다.
- --engine-version—반드시 3.2.6, 4.0.10 또는 이상이어야 합니다.
- --at-rest-encryption-enabled—유휴 시 암호화를 활성화하기 위해 필요합니다.
- --cache-parameter-group—이 클러스터 모드가 활성화된 복제 그룹을 만들려면 default-redis4.0.cluster.on 또는 이 클러스터에서 파생된 클러스터여야 합니다.

Example 2: Redis(클러스터 모드 활성화됨) 클러스터

Linux, macOS, or Unix의 경우:

```
aws elasticache create-replication-group \
    --replication-group-id my-clustered-rg \
    --replication-group-description "redis clustered cluster" \
    --cache-node-type cache.m3.large \
```

```
--num-node-groups 3 \
--replicas-per-node-group 2 \
--engine redis \
--engine-version 4.0.10 \
--at-rest-encryption-enabled \
--cache-parameter-group default.redis4.0.cluster.on
```

Windows의 경우:

```
aws elasticache create-replication-group ^
--replication-group-id my-clustered-rg ^
--replication-group-description "redis clustered cluster" ^
--cache-node-type cache.m3.large ^
--num-node-groups 3 ^
--replicas-per-node-group 2 ^
--engine redis ^
--engine-version 4.0.10 ^
--at-rest-encryption-enabled ^
--cache-parameter-group default.redis4.0.cluster.on
```

추가 정보는 다음을 참조하십시오.

- Redis(클러스터 모드 활성화됨) 복제 그룹을 처음부터 새로 생성 (AWS CLI) (p. 154)
- · create-replication-group

ElastiCache API를 사용하여 미사용 데이터 암호화 활성화

ElastiCache API를 사용하여 Redis 복제 그룹을 생성할 때 미사용 데이터 암호화를 활성화하려면 CreateReplicationGroup을 사용하여 파라미터 AtRestEncryptionEnabled를 true로 설정합니다.

Redis(클러스터 모드 비활성화됨) 클러스터에서 미사용 데이터 암호화 활성화(API)

다음 작업은 세 개의 노드(NumCacheClusters) 즉, 기본 복제본 한 개와 읽기 전용 복제본 두 개가 있는 Redis(클러스터 모드 비활성화됨) 복제 그룹 my-classic-rg을 생성합니다. 미사용 데이터 암호화가 이 복 제 그룹에 대해 활성화되어 있습니다(AtRestEncryptionEnabled=true).

다음 파라미터와 해당 값은 복제 그룹에 대한 암호화를 활성화하는 데 필요합니다.

- Engine—반드시 redis여야 합니다.
- EngineVersion—반드시 3.2.6, 4.0.10 또는 이상이어야 합니다.
- AtRestEncryptionEnabled—미사용 데이터 암호화를 활성화려면 true로 설정해야 합니다.

Example 3: Redis(클러스터 모드 비활성화됨) 복제본이 있는 클러스터

읽기 쉽도록 줄바꿈이 추가됩니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=CreateReplicationGroup
    &AtRestEncryptionEnabled=true
    &CacheNodeType=cache.m3.large
    &CacheParameterGroup=default.redis4.0
    &Engine=redis
    &EngineVersion=4.0.10
    &NumCacheClusters=3
    &ReplicationGroupDescription=test%20group
    &ReplicationGroupId=my-classic-rg
    &Version=2015-02-02
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
```

&Timestamp=20150202T192317Z &X-Amz-Credential=<credential>

추가 정보는 다음을 참조하십시오.

- Redis(클러스터 모드 비활성화됨) 복제 그룹을 (ElastiCache API)의 처음부터 새로 생성 (p. 151)
- CreateReplicationGroup

Redis(클러스터 모드 활성화됨)에 대한 클러스터에서 미사용 데이터 암호화 활성화(API)

다음 작업은 세 개의 노드 그룹/샤드(NumNodeGroups)가 있는 Redis(클러스터 모드 활성화됨) 복제 그룹 my-clustered-rg를 생성합니다. 이 복제 그룹은 각각 세 개의 노드인 기본 복제본 한 개와 읽기 전용 복제본 두 개(ReplicasPerNodeGroup)로 구성됩니다. 미사용 데이터 암호화가 이 복제 그룹에 대해 활성화되어 있습니다(AtRestEncryptionEnabled=true).

다음 파라미터와 해당 값은 복제 그룹에 대한 암호화를 활성화하는 데 필요합니다.

- Engine—반드시 redis여야 합니다.
- AtRestEncryptionEnabled—미사용 데이터 암호화를 활성화려면 true로 설정해야 합니다.
- EngineVersion—반드시 3.2.6, 4.0.10 또는 이상이어야 합니다.
- CacheParameterGroup—default-redis4.0.cluster.on 또는 이 클러스터에서 파생된 클러스터 가 Redis(클러스터 모드 활성화됨) 클러스터가 되어야 합니다.

Example 4: Redis(클러스터 모드 활성화됨) 클러스터

읽기 쉽도록 줄바꿈이 추가됩니다.

```
https://elasticache.us-west-2.amazonaws.com/
   ?Action=CreateReplicationGroup
  &AtRestEncryptionEnabled=true
  &CacheNodeType=cache.m3.large
  &CacheParemeterGroup=default.redis4.0.cluster.on
  &Engine=redis
  &EngineVersion=4.0.10
  &NumNodeGroups=3
  &ReplicasPerNodeGroup=2
  &ReplicationGroupDescription=test%20group
  &ReplicationGroupId=my-clustered-rg
  &Version=2015-02-02
  &SignatureVersion=4
  &SignatureMethod=HmacSHA256
  &Timestamp=20150202T192317Z
   &X-Amz-Credential=<credential>
```

추가 정보는 다음을 참조하십시오.

- Redis(클러스터 모드 활성화됨)에서 복제 그룹을 처음부터 새로 생성 (ElastiCache API) (p. 158)
- CreateReplicationGroup

참고 항목

- Amazon VPC 및 ElastiCache 보안 (p. 316)
- Amazon ElastiCache의 Identity and Access Management(IAM) (p. 341)

Redis AUTH 명령을 사용하여 사용자 인증

Redis 인증 토큰은 Redis가 클라이언트의 명령 실행을 허용하기 전에 토큰(암호)을 요구할 수 있게 함으로써 데이터 보안을 개선합니다.

주제

- Redis용 ElastiCache에서의 AUTH 개요 (p. 313)
- Redis용 ElastiCache 클러스터에 인증 적용 (p. 313)
- 기존 Redis용 ElastiCache 클러스터에서 AUTH 토큰 수정 (p. 314)
- 관련 주제 (p. 315)

Redis용 ElastiCache에서의 AUTH 개요

Redis용 ElastiCache에서 Redis AUTH를 사용할 때 몇 가지 제약 사항이 있습니다.

특히 Redis용 ElastiCache에서 AUTH를 사용할 때는 이러한 AUTH 토큰 제약 조건에 유의해야 합니다.

- 토큰은 16-128 길이의 인쇄 가능한 문자여야 합니다.
- 영숫자 외의 특수 문자는 (!, &, #, \$, ^, <, >, -)로 제한됩니다.
- AUTH는 전송 중 데이터 암호화가 활성화된 Redis용 ElastiCache 클러스터에서만 사용할 수 있습니다.

강력한 토큰을 설정하려면 다음 사항을 요구하는 등 엄격한 토큰 정책을 따르는 것이 좋습니다.

- 토큰에는 다음 문자 유형 중 3개 이상이 포함되어야 합니다.
 - 대문자
 - 소문자
 - 숫자
 - 영숫자 이외의 문자(!, &, #, \$, ^, <, >, -)
- 토큰에는 사전에 나오는 단어나 약간 수정한 형태의 사전 단어가 포함되어서는 안 됩니다.
- 최근에 사용한 토큰과 같거나 비슷한 토큰도 안 됩니다.

Redis용 ElastiCache 클러스터에 인증 적용

토큰 보호 Redis 서버에서 토큰을 입력하도록 사용자에게 요구할 수 있습니다. 이렇게 하려면 복제 그룹 또는 클러스터를 생성할 때 올바른 토큰을 가진 파라미터 --auth-token(API: AuthToken)을 포함시킵니다. 또한 복제 그룹 또는 클러스터에 대한 모든 후속 명령에 이를 포함시킵니다.

다음 AWS CLI 작업은 전송 중 데이터 암호화(TLS)가 활성화되고 AUTH 토큰 This-is-α-sample-token이 포함된 복제 그룹을 생성합니다. 서브넷 그룹 sng-test는 존재하는 서브넷 그룹으로 대체합니다.

키 파라미터

- --engine 반드시 redis여야 합니다.
- --engine-version- 반드시 3.2.6, 4.0.10 또는 그 이상이어야 합니다.
- --transit-encryption-enabled- 인증 및 HIPAA 자격 획득에 필요합니다.
- --auth-token HIPAA 자격 획득에 필요합니다. 이 값은 이러한 토큰 보호 Redis 서버를 위한 올바른 토큰이어야 합니다.
- --cache-subnet-group HIPAA 자격 획득에 필요합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-replication-group \
--replication-group-id authtestgroup \
--replication-group-description authtest \
--engine redis \
--engine-version 4.0.10 \
--cache-node-type cache.m4.large \
--num-node-groups 1 \
--replicas-per-node-group 2 \
--cache-parameter-group default.redis3.2.cluster.on \
--transit-encryption-enabled \
--auth-token This-is-a-sample-token \
--cache-subnet-group sng-test
```

Windows의 경우:

기존 Redis용 ElastiCache 클러스터에서 AUTH 토큰 수정

보다 손쉬운 인증 업데이트를 위해 Redis용 ElastiCache 클러스터에서 사용되는 AUTH 토큰을 수정할 수 있습니다. 엔진 버전이 5.0.5 이상이고 Redis용 ElastiCache에서 전송 중 데이터 암호화가 활성화되어 있는 경우에는 이러한 수정이 가능합니다.

auth 토큰 수정은 ROTATE와 SET 등 두 가지 전략을 지원합니다. ROTATE 전략은 이전 토큰을 유지하면서 서버에 새 AUTH 토큰을 추가합니다. SET 전략은 단일 AUTH 토큰만 지원하도록 서버를 업데이트합니다. --apply-immediately 파라미터를 통해 이러한 수정 호출을 수행하면 변경 사항이 즉시 적용됩니다.

AUTH 토큰 교체

새 AUTH 토큰을 갖도록 Redis 서버를 업데이트하려면 새 AUTH 토큰 역할을 하는 --auth-token 파라미터와 값이 ROTATE인 --auth-token-update-strategy로 ModifyReplicationGroup API를 호출합니다. 일단 수정이 완료되면 클러스터는 auth-token 파라미터에 지정된 것 외에 이전의 AUTH 토큰을 지원하게 됩니다.

두 가지 AUTH 토큰을 이미 지원하는 서버에서 이러한 수정을 수행하면 수정 작업이 수행되는 동안 가장 오래된 AUTH 토큰이 제거되기 때문에 서버는 해당 시점에서 가장 최근의 AUTH 토큰을 최대 두 개까지 지원할수 있습니다.

이 시점에서 가장 최근의 AUTH 토큰을 사용하도록 클라이언트를 업데이트하면 작업을 계속 진행할 수 있습니다. 클라이언트가 업데이트되면 AUTH 토큰 교체를 위한 SET 전략(다음 단원에 설명)을 이용해 새 토큰의사용을 단독으로 시작할 수 있습니다.

다음 AWS CLI 작업은 AUTH 토큰을 This-is-the-rotated-token으로 교체하도록 복제 그룹을 수정합니다.

Linux. macOS 또는 Unix의 경우는 다음과 같습니다.

```
aws elasticache modify-replication-group \
--replication-group-id authtestgroup \
--auth-token This-is-the-rotated-token \
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 인터네트워크 트래픽 개인 정보

```
--auth-token-update-strategy ROTATE \
--apply-immediately n
```

Windows의 경우:

```
aws elasticache modify-replication-group ^
--replication-group-id authtestgroup ^
--auth-token This-is-the-rotated-token ^
--auth-token-update-strategy ROTATE ^
--apply-immediately
```

AUTH 토큰 설정

단일 AUTH 토큰을 지원하기 위해 두 개의 AUTH 토큰을 갖도록 Redis 서버를 업데이트하려면 ModifyReplicationGroup API 작업을 호출합니다. 새 AUTH 토큰 역할을 하는 --auth-token 파라미터와 값이 SET인 --auth-token-update-strategy 파라미터로 ModifyReplicationGroup를 호출합니다. auth-token 파라미터는 교체된 마지막 AUTH 토큰과 값이 같아야 합니다. 수정이 완료되고 나면 Redis 서버는 auth-token 파라미터에 지정된 AUTH 토큰만 지원합니다.

다음 AWS CLI 작업은 AUTH 토큰을 This-is-the-set-token으로 설정하도록 복제 그룹을 수정합니다.

Linux, macOS 또는 Unix의 경우는 다음과 같습니다.

```
aws elasticache modify-replication-group \
--replication-group-id authtestgroup \
--auth-token This-is-the-set-token\
--auth-token-update-strategy SET \
--apply-immediately
```

Windows의 경우:

```
aws elasticache modify-replication-group ^
--replication-group-id authtestgroup ^
--auth-token This-is-the-set-token ^
--auth-token-update-strategy SET ^
--apply-immediately
```

기존 Redis용 ElastiCache 클러스터에서 인증 활성화

기존 Redis 서버에서 인증을 활성화하려면 ModifyReplicationGroup API 작업을 호출합니다. 새 토큰 역할을 하는 --auth-token 파라미터와 값이 ROTATE인 --auth-token-update-strategy 파라미터 로 ModifyReplicationGroup을 호출합니다.

수정이 완료되고 나면 클러스터는 인증 없는 연결을 지원하는 것 외에도 auth-token 파라미터에 지정된 AUTH 토큰을 지원합니다. 인증 활성화는 전송 중 데이터 암호화(TLS)가 활성화된 Redis 서버에서만 지원됩니다.

관련 주제

• redis.io 웹 사이트 상의 AUTH 토큰.

인터네트워크 트래픽 개인 정보

Amazon ElastiCache는 다음 기술을 사용하여 캐시 데이터 보안을 유지하고 무단 액세스로부터 보호합니다.

- Amazon VPC 및 ElastiCache 보안 (p. 316)은 설치에 필요한 보안 그룹 유형을 설명합니다.
- 사용자, 그룹 및 역할의 작업을 부여하고 제한하기 위한 Amazon ElastiCache의 Identity and Access Management(IAM) (p. 341)

Amazon VPC 및 ElastiCache 보안

데이터 보안이 중요하기 때문에 ElastiCache는 데이터를 액세스할 수 있는 대상을 제어하는 수단을 제공합니다. 데이터에 대한 액세스를 제어하는 방법은 클러스터를 Amazon Virtual Private Cloud(Amazon VPC)에서 시작했는지 또는 Amazon EC2-Classic에서 시작했는지에 따라 다릅니다.

Important

ElastiCache 클러스터 시작을 위해 Amazon EC2-Classic을 사용하던 방식은 사용이 중지되었습니다. 모든 현재 세대 노드는 Amazon Virtual Private Cloud에서만 시작됩니다.

주제

- ElastiCache 및 Amazon VPC 이해 (p. 316)
- Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴 (p. 320)
- Virtual Private Cloud(VPC) 생성 (p. 326)
- 캐시 서브넷 그룹 생성 (p. 328)
- Amazon VPC에서 캐시 클러스터 생성 (p. 328)
- Amazon VPC에서 복제 그룹 생성 (p. 328)
- Amazon VPC에서 실행되는 클러스터 또는 복제 그룹 에 연결 (p. 328)

Amazon Virtual Private Cloud(Amazon VPC) 서비스는 전통적인 데이터 센터와 매우 유사한 가상 네트워크를 정의합니다. Amazon VPC를 구성할 때 그 IP 주소 범위를 선택하고 서브넷을 생성하고 라우팅 테이블, 네트워크 게이트웨이 및 보안 설정을 구성할 수 있습니다. 또한 Amazon VPC 보안 그룹을 사용하여 가상 네트워크에 캐시 클러스터를 추가하고 캐시 클러스터에 대한 액세스 권한을 제어할 수 있습니다.

이 단원에서는 Amazon VPC의 ElastiCache 클러스터를 수동으로 구성하는 방법을 설명합니다. 이 정보는 ElastiCache 및 Amazon VPC가 연동되는 방식을 더 깊이 이해하고자 하는 사용자를 대상으로 합니다.

주제

- ElastiCache 및 Amazon VPC 이해 (p. 316)
- Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴 (p. 320)
- Virtual Private Cloud(VPC) 생성 (p. 326)
- 캐시 서브넷 그룹 생성 (p. 328)
- Amazon VPC에서 캐시 클러스터 생성 (p. 328)
- Amazon VPC에서 복제 그룹 생성 (p. 328)
- Amazon VPC에서 실행되는 클러스터 또는 복제 그룹 에 연결 (p. 328)

ElastiCache 및 Amazon VPC 이해

ElastiCache은 Amazon Virtual Private Cloud (Amazon VPC)과 완벽하게 통합됩니다. ElastiCache 사용자의 경우 이는 다음을 의미합니다.

- AWS 계정에서 EC2-VPC 플랫폼만 지원하는 경우 ElastiCache는 항상 Amazon VPC에서 클러스터를 시작합니다.
- AWS를 처음 이용하는 경우 클러스터가 Amazon VPC에 배포됩니다. 기본 VPC가 자동으로 생성됩니다.
- 기본 VPC가 있고 클러스터를 시작할 때 서브넷을 지정하지 않으면 클러스터가 기본 Amazon VPC에서 시작합니다.

자세한 정보는 Detecting Your Supported Platforms and Whether You Have a Default VPC를 참조하십시오.

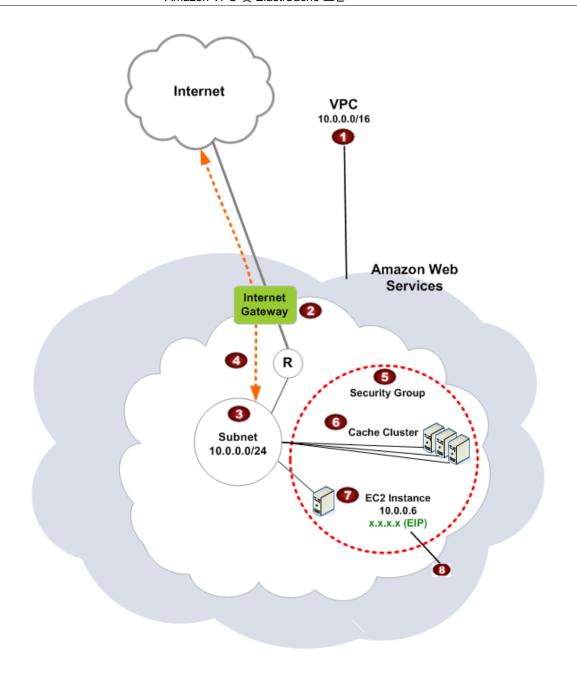
Amazon Virtual Private Cloud를 사용하면 기존의 데이터 센터와 매우 유사한 AWS 클라우드에 가상 네트워크를 생성할 수 있습니다. IP 주소 범위 선택, 서브넷 생성 및 라우팅 테이블, 네트워크 게이트웨이, 보안 설정구성을 포함하여 Amazon VPC를 구성할 수 있습니다.

ElastiCache의 기본 기능은 가상 프라이빗 클라우드에서 동일합니다. ElastiCache는 클러스터가 Amazon VPC의 내부 또는 외부에 배포되는 여부와 관계없이 소프트웨어 업그레이드, 패치 적용, 장애 탐지 및 복구를 관리합니다.

Amazon VPC 외부에 배포된 ElastiCache 캐시 노드에는 엔드포인트/DNS 이름이 확인되는 IP 주소가 할당됩니다. 이는 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스에서 연결을 제공합니다. ElastiCache 클러스터를 Amazon VPC 프라이빗 서브넷으로 실행할 때 모든 캐시 노드는 그 서브넷 내의 프라이빗 IP 주소에 할당됩니다.

Amazon VPC에서의 ElastiCache 개요

다음 다이어그램과 표에는 Amazon VPC에서 시작되는 ElastiCache 클러스터 및 Amazon EC2 인스턴스와 함께 Amazon VPC 환경에 대한 설명이 나와 있습니다.



- Amazon VPC는 자체 IP 주소 블록이 할당된 AWS 클라우드의 격리된 부분입니다.
- 인터넷 게이트웨이는 Amazon VPC를 인터넷에 직접 연결하고 Amazon VPC 외부에서 실행되는 Amazon Simple Storage Service(Amazon S3) 와 같은 다른 AWS 리소스에 대한 액세스를 제공합니다.
- Amazon VPC 서브넷은 보안 및 운영상의 필요에 따라 AWS 리소스를 격리할 수 있는 Amazon VPC의 IP 주소 범위 세그먼트입니다.
- ▲ Amazon VPC의 라우팅 테이블은 서브넷과 인터넷 간 네트워크 트래픽을 지시합니다. Amazon VPC에는 이 다이어그램에서 원 안에 R로 표시된 라우터가 내재되어 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Amazon VPC 및 ElastiCache 보안

- **6** Amazon VPC 보안 그룹은 ElastiCache 클러스터와 Amazon EC2 인스턴스의 인바운드 및 아웃바운드 트래픽을 제어합니다.
- 6 서브넷에서 ElastiCache 클러스터를 실행할 수 있습니다. 캐시 노드는 서브넷 주소 범위의 프라이 빗 IP 주소를 가집니다.
- ▼한 서브넷에서 Amazon EC2 인스턴스를 시작할 수 있습니다. 각각의 Amazon EC2 인스턴스는 서브넷 주소 범위의 프라이빗 IP 주소를 가집니다. Amazon EC2 인스턴스를 동일한 서브넷의 모든 캐시 노드에 연결할 수 있습니다.
- 8 Amazon VPC의 Amazon EC2 인스턴스를 인터넷에 연결하려면 인스턴스에 탄력적 IP 주소라는 고 정 퍼블릭 주소를 할당해야 합니다.

ElastiCache 배포에서 EC2 Classic 대신 Amazon VPC를 사용해야 하는 이유

인스턴스를 Amazon VPC로 시작하면 다음을 수행할 수 있습니다.

- 인스턴스의 시작/중지에 상관 없이 유지되는 고정 IP 주소 할당.
- 인스턴스에 여러 개의 IP 주소 할당이 가능합니다.
- 네트워크 인터페이스를 정의하고, 하나 혹은 그 이상의 네트워크 인터페이스를 귀하의 인스턴스에 설치 가능합니다.
- 인스턴스를 실행하는 동안 인스턴스의 보안 그룹 멤버십을 변경할 수 있습니다.
- 인스턴스의 인바운드 트래픽 제어(인그레스 필터링)뿐만 아니라 아웃바운드 트래픽도 제어(인그레스 필터링)할 수 있습니다.
- 네트워크 액세스 제어 리스트(ACL)를 통해, 인스턴스에 대해 액세스 제어가 한 단계 더 강화되었습니다.
- 단일 테넌트 하드웨어에서 인스턴스 실행.

Amazon EC2 Classic, 기본 VPC 및 기본 외 VPC를 비교하려면 VPC-Classic과 EC2-VPC의 차이점을 참조하십시오.

Amazon VPC는 비전용 Amazon EC2 인스턴스를 허용해야 합니다. 전용 인스턴스 테넌시로 구성된 Amazon VPC에서는 ElastiCache를 사용할 수 없습니다.

사전 조건

Amazon VPC에 ElastiCache 클러스터를 생성하려면 Amazon VPC가 다음 요구 사항을 충족해야 합니다.

- Amazon VPC는 비전용 Amazon EC2 인스턴스를 허용해야 합니다. 전용 인스턴스 테넌시로 구성된 Amazon VPC에서는 ElastiCache를 사용할 수 없습니다.
- Amazon VPC에 대해 캐시 서브넷 그룹을 정의해야 합니다. ElastiCache는 캐시 서브넷 그룹을 사용하여 캐시 노드에 연결된 서브넷 내의 서브넷 및 IP 주소를 선택할 수 있습니다.
- Amazon VPC에 대해 캐시 보안 그룹을 정의하거나 제공된 기본값을 사용할 수 있습니다.
- 각 서브넷의 CIDR 블록은 유지 관리 작업에서 ElastiCache에 사용할 여분의 IP 주소를 제공할 수 있을 만큼 충분히 커야 합니다.

라우팅 및 보안

Amazon VPC에서 라우팅을 구성하여 트래픽 흐름(예: 인터넷 게이트웨이, 가상 프라이빗 게이트웨이)을 제어할 수 있습니다. 인터넷 게이트웨이를 통해 Amazon VPC를 Amazon VPC에서 실행되지 않는 다른 AWS 리소스에 직접 액세스할 수 있습니다. 조직의 로컬 네트워크에 연결된 가상 사설 게이트웨이만을 사용하도록 선택한 경우, VPN을 통해 인터넷 바운드 트래픽을 라우팅하고 출구를 제어하기 위한 로컬 보안 정책 및 방화

벽을 사용할 수 있습니다. 이 경우 인터넷을 통해 AWS 리소스에 액세스할 때 대역폭 요금이 추가로 부과됩니다.

Amazon VPC 보안 그룹을 사용하여 Amazon VPC에서 ElastiCache 클러스터 및 Amazon EC2 인스턴스를 보호할 수 있습니다. 보안 그룹은 서브넷 레벨이 아닌 인스턴스 레벨에서 방화벽처럼 작동합니다.

Note

캐시 노드를 재부팅하는 경우 기본 IP 주소가 변경될 수 있으므로 캐시 노드에 연결할 때 DNS 이름을 사용하는 것이 좋습니다.

Amazon VPC 문서

Amazon VPC에는 Amazon VPC를 생성하고 사용하는 방법을 설명하는 자체 문서 세트가 있습니다. 다음 테이블은 Amazon VPC 지침의 링크를 제공합니다.

설명	설명서
Amazon VPC 사용을 시작하는 방법	Amazon VPC 시작 안내서
AWS Management 콘솔를 통해 Amazon VPC를 사용하는 방법	Amazon VPC 사용 설명서
모든 Amazon VPC 명령의 전체 설명	Amazon EC2 명령줄 레퍼런스 (Amazon VPC 명령 은 Amazon EC2 참조의 일부임)
Amazon VPC API 작업, 데이터 형식 및 오류의 전체 설명	Amazon EC2 API Reference (Amazon VPC API 작 업은 Amazon EC2 참조의 일부임)
선택적인 IPsec VPN 연결 사용자 측의 게이트웨이 를 구성하는 데 필요한 네트워크 관리자를 위한 정 보	AWS Site-to-Site VPN 네트워크 관리자 안내서

Amazon Virtual Private Cloud에 대한 자세한 내용은 Amazon Virtual Private Cloud 단원을 참조하십시오.

Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴

Amazon ElastiCache에서는 Amazon VPC에 있는 클러스터에 액세스할 수 있도록 다음 시나리오를 지원합니다.

목차

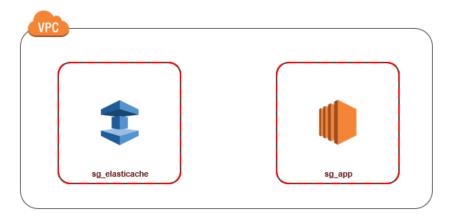
- ElastiCache 클러스터와 Amazon EC2 인스턴스가 같은 Amazon VPC에 있는 경우 클러스터 액세 스 (p. 321)
- ElastiCache 클러스터와 Amazon EC2 인스턴스가 다른 Amazon VPC에 있는 경우 클러스터 액세 스 (p. 322)
 - ElastiCache 클러스터와 Amazon EC2 인스턴스가 같은 리전의 다른 Amazon VPC에 있는 경우 클러스터 액세스 (p. 322)
 - ElastiCache 클러스터와 Amazon EC2 인스턴스가 다른 리전의 다른 Amazon VPC에 있는 경우 클러스터 액세스 (p. 323)
- 고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러스터 액세스 (p. 324)
 - VPN 연결을 사용하여 고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러스터 액세스 (p. 324)

• Direct Connect를 사용하여 고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러스터 액세스 (p. 325)

ElastiCache 클러스터와 Amazon EC2 인스턴스가 같은 Amazon VPC에 있는 경우 클러스터 액세스

가장 일반적인 사용 사례는 EC2 인스턴스에 배포된 애플리케이션이 같은 VPC에 있는 클러스터에 연결해야하는 경우입니다.

다음은 이 시나리오를 설명한 다이어그램입니다



동일한 VPC에서 EC2 인스턴스와 DB 인스턴스 간 액세스를 관리하는 가장 간단한 방법은 다음과 같습니다.

1. 클러스터의 VPC 보안 그룹을 만듭니다. 이 보안 그룹을 사용해 클러스터 인스턴스에 대한 액세스를 제한할 수 있습니다. 예를 들어, 클러스터를 만들 때 할당한 포트와 클러스터에 액세스할 때 이용할 IP 주소를 사용해 TCP 액세스를 허용하는 이 보안 그룹의 사용자 지정 규칙을 만들 수 있습니다.

Redis 클러스터 및 복제 그룹의 기본 포트는 6379입니다.

- 2. EC2 인스턴스(웹 및 애플리케이션 서버)의 VPC 보안 그룹을 만듭니다. 이 보안 그룹은 필요할 경우 VPC의 라우팅 테이블을 통한 EC2 인스턴스 액세스를 허용할 수 있습니다. 예를 들어, 이 보안 그룹에서 TCP가 포트 22를 통해 EC2 인스턴스에 액세스하도록 허용하는 규칙을 설정할 수 있습니다.
- 3. 클러스터에 대한 보안 그룹에서 EC2 인스턴스에 대해 생성한 보안 그룹으로부터의 연결을 허용하는 사용자 지정 규칙을 만듭니다. 그러면 보안 그룹의 모든 구성원이 DB 인스턴스에 액세스하도록 허용됩니다.

VPC 보안 그룹에서 다른 보안 그룹으로부터의 연결을 허용하는 규칙을 만들려면

- 1. AWS Management Console에 로그인하고 https://console.aws.amazon.com/vpc에서 Amazon VPC 콘 솔을 엽니다.
- 2. 탐색 창에서 [Security Groups]를 선택합니다.
- 3. 클러스터 인스턴스에 사용할 보안 그룹을 선택하거나 만듭니다. 규칙 추가를 선택합니다. 이 보안 그룹 은 다른 보안 그룹 멤버에 대한 액세스를 허용합니다.
- 4. [Type]에서 [Custom TCP Rule]을 선택합니다.
 - a. [Port Range]에 대해 클러스터를 만들 때 사용한 포트를 지정합니다.
 - Redis 클러스터 및 복제 그룹의 기본 포트는 6379입니다.
 - b. [Source] 상자에 보안 그룹 ID를 입력합니다. 목록에서 Amazon EC2 인스턴스에 사용할 보안 그룹을 선택합니다.
- 5. 완료되면 [Save]를 선택합니다.



ElastiCache 클러스터와 Amazon EC2 인스턴스가 다른 Amazon VPC에 있는 경우 클러스터 액세스

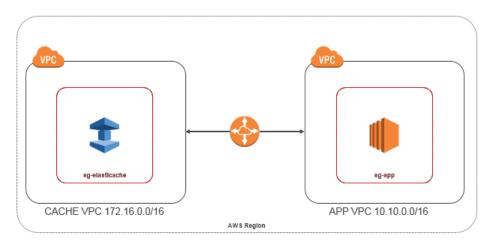
클러스터가 액세스에 사용할 EC2 인스턴스와 다른 VPC에 있는 경우 여러 가지 방법으로 DB 인스턴스에 액세스할 수 있습니다. 클러스터와 EC2 인스턴스가 서로 다른 VPC에 있지만 리전은 동일한 경우 VPC 피어링을 사용할 수 있습니다. 클러스터와 EC2 인스턴스가 서로 다른 리전에 있으면 리전 간에 VPN 연결을 만들수 있습니다.

주제

- ElastiCache 클러스터와 Amazon EC2 인스턴스가 같은 리전의 다른 Amazon VPC에 있는 경우 클러스터 액세스 (p. 322)
- ElastiCache 클러스터와 Amazon EC2 인스턴스가 다른 리전의 다른 Amazon VPC에 있는 경우 클러스터 액세스 (p. 323)

ElastiCache 클러스터와 Amazon EC2 인스턴스가 같은 리전의 다른 Amazon VPC에 있는 경우 클러스터 액세스

다음 다이어그램에서는 Amazon VPC 피어링 연결을 사용하여 같은 리전의 서로 다른 Amazon VPC에 있는 Amazon EC2 인스턴스에서 클러스터에 액세스하는 과정을 보여줍니다.



같은 리전의 다른 Amazon VPC에 있는 Amazon EC2 인스턴스가 클러스터 액세스 - VPC 피어링 연결

VPC 피어링 연결은 프라이빗 IP 주소를 사용하여 두 VPC 간에 트래픽을 라우팅할 수 있도록 하기 위한 두 VPC 사이의 네트워킹 연결입니다. 동일한 네트워크에 속하는 경우와 같이 VPC의 인스턴스가 서로 통신

할 수 있습니다. 자체 Amazon VPC간의 VPC 피어링 연결, 또는 단일 리전 내에 있는 다른 AWS 계정에서 Amazon VPC와의 VPC 피어링 연결을 만들 수 있습니다. Amazon VPC 피어링에 대한 자세한 내용은 VPC 설명서를 참조하십시오.

피어링으로 다른 Amazon VPC에 있는 클러스터에 액세스하려면

- 1. 두 VPC에 겹치는 IP 범위가 없거나 이 VPC를 피어링할 수 없어야 합니다.
- 2. 두 VPC를 피어링합니다. 자세한 내용은 Amazon VPC 피어링 연결 생성 및 수락을 참조하십시오.
- 3. 라우팅 테이블을 업데이트합니다. 자세한 내용은 VPC 피어링 연결을 위한 라우팅 테이블 업데이트를 참조하십시오.

앞에 나온 다이어그램의 예제에 대한 라우팅 테이블은 다음과 같습니다. pcx-a894f1c1이 피어링 연결입니다.

Destination	Target	Destination	Target
172.16.0.0/16	local	10.10.0.0/16	local
10.10.0.0/16	pcx-a894f1c1	0.0.0.0/0	igw-bfdcccd8
		172.16.0.0/16	pcx-a894f1c1

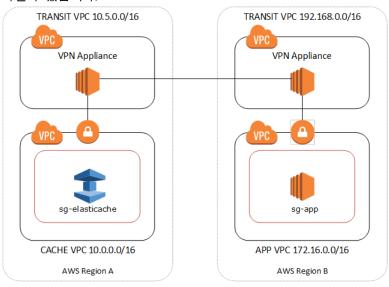
VPC 라우팅 테이블

4. ElastiCache 클러스터의 보안 그룹을 수정하여 피어링된 VPC의 애플리케이션 보안 그룹에서 들어오는 인바운드 연결을 허용합니다. 자세한 내용은 피어 VPC 보안 그룹 참조를 참조하십시오.

피어링 연결을 통해 클러스터에 액세스하면 데이터 전송 비용이 추가로 발생합니다.

ElastiCache 클러스터와 Amazon EC2 인스턴스가 다른 리전의 다른 Amazon VPC에 있는 경우 클러스터 액세스

지리적으로 떨어져 있는 여러 VPC와 원격 네트워크를 연결하는 일반적인 전략 하나는 글로벌 네트워크 전송 센터 역할을 하는 전송 VPC를 만드는 것입니다. 전송 VPC는 네트워크 관리를 간소화하고 여러 VPC와 원격 네트워크를 연결하는 데 필요한 연결 수를 최소화합니다. 이 디자인은 기존의 방식대로 공동 배치 전송 허브에 실제 존재를 만들거나 물리적 네트워크 장비를 배포하지 않고 가상으로 구현되므로 시간, 노력, 비용을 아낄 수 있습니다.



다른 리전의 다른 VPC를 지나는 연결

전송 Amazon VPC가 설정되면 한 리전의 "스포크" VPC에 배포된 애플리케이션이 다른 리전의 "스포크" VPC에 있는 ElastiCache 클러스터에 연결할 수 있습니다.

다른 리전의 다른 VPC에 있는 클러스터에 액세스하려면

- 1. 전송 VPC 솔루션을 배포합니다. 자세한 내용은 How do I build a global transit network on AWS? 단원을 참조하십시오.
- 2. 앱 및 캐시 VPC에서 VPC 라우팅 테이블을 업데이트하여 VGW(가상 프라이빗 게이트웨이) 및 VPN 어 플라이언스를 통해 트래픽을 라우팅합니다. BGP(Border Gateway Protocol)를 사용하는 동적 라우팅의 경우 라우팅이 자동으로 전파됩니다.
- 3. ElastiCache 클러스터의 보안 그룹을 수정하여 애플리케이션 인스턴스 IP 범위에서 들어오는 인바운드 연결을 허용합니다. 이 시나리오에는 애플리케이션 서버 보안 그룹을 참조할 수 없습니다.

여러 리전의 클러스터에 액세스하면 네트워크 지연 시간이 생기고 교차 리전 데이터 전송 비용이 추가로 발생합니다.

고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러스터 액세스

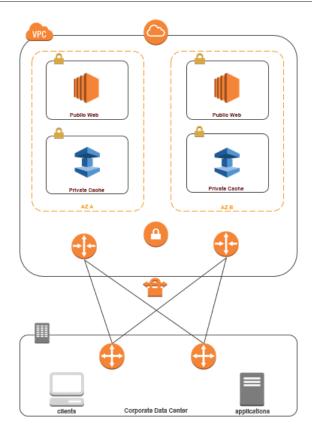
고객 데이터 센터의 클라이언트나 애플리케이션이 VPC의 ElastiCache 클러스터에 액세스해야 하는 하이브 리드 아키텍처도 가능한 시나리오입니다. VPN이나 Direct Connect를 통해 고객의 VPC와 데이터 센터 간에 연결을 제공하여 이 시나리오도 지원됩니다.

주제

- VPN 연결을 사용하여 고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러스터 액세 스 (p. 324)
- Direct Connect를 사용하여 고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러스터 액세스 (p. 325)

VPN 연결을 사용하여 고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러스터 액세스

다음 다이어그램에서는 VPN 연결을 사용하여 회사 네트워크에서 실행되는 애플리케이션에서 ElastiCache 클러스터에 액세스하는 과정을 보여줍니다.



VPN를 통해 데이터 센터에서 ElastiCache에 연결

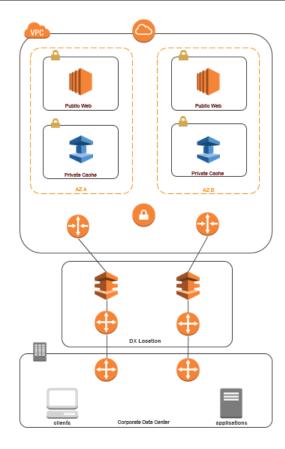
VPN 연결을 통해 온-프레미스 애플리케이션에서 VPC의 클러스터에 액세스하려면

- 1. 하드웨어 가상 프라이빗 게이트웨이를 VPC에 추가하여 VPN 연결을 설정합니다. 자세한 내용은 VPC에 하드웨어 가상 프라이빗 게이트웨이 추가를 참조하십시오.
- 2. ElastiCache 클러스터가 배포되는 서브넷의 VPC 라우팅 테이블을 업데이트하여 온-프레미스 애플리케이션 서버에서 들어오는 트래픽을 허용합니다. BGP를 사용하는 동적 라우팅의 경우 라우팅이 자동으로 전파될 수 있습니다.
- 3. ElastiCache 클러스터의 보안 그룹을 수정하여 온-프레미스 애플리케이션 서버에서 들어오는 인바운드 연결을 허용합니다.

VPN 연결을 통해 클러스터에 액세스하면 네트워크 지연 시간이 생기고 데이터 전송 비용이 추가로 발생합니다.

Direct Connect를 사용하여 고객의 데이터 센터에서 실행되는 애플리케이션에서 ElastiCache 클러 스터 액세스

다음 다이어그램에서는 Direct Connect를 사용하여 회사 네트워크에서 실행되는 애플리케이션에서 ElastiCache 클러스터에 액세스하는 과정을 보여줍니다.



Direct Connect를 통해 데이터 센터에서 ElastiCache에 연결

Direct Connect를 사용하여 네트워크에서 실행되는 애플리케이션에서 ElastiCache 클러스터에 액 세스하려면

- 1. Direct Connect 연결을 설정합니다. 자세한 내용은 AWS Direct Connect 시작하기를 참조하십시오.
- 2. ElastiCache 클러스터의 보안 그룹을 수정하여 온-프레미스 애플리케이션 서버에서 들어오는 인바운드 연결을 허용합니다.

DX 연결을 통해 클러스터에 액세스하면 네트워크 지연 시간이 생기고 데이터 전송 요금이 추가로 발생할 수 있습니다.

Virtual Private Cloud(VPC) 생성

이 예제에서는 각 가용 영역에 대해 프라이빗 서브넷이 있는 Amazon VPC를 생성합니다.

Amazon VPC 만들기 (콘솔)

Amazon Virtual Private Cloud내에서 ElastiCache 클러스터를 생성하려면

- 1. AWS Management Console에 로그인한 후 https://console.aws.amazon.com/vpc/에서 Amazon VPC 콘 솔을 엽니다.
- 2. Amazon Virtual Private Cloud 마법사를 사용하여 새 Amazon VPC를 생성합니다.
 - a. 탐색 목록에서 [VPC Dashboard]를 선택합니다.
 - b. [Start VPC Wizard]를 선택합니다.

- c. Amazon VPC 마법사에서 퍼블릭 및 프라이빗 서브넷이 있는 VPC를 선택한 다음 다음을 선택합니다.
- d. [VPC with Public and Private Subnets] 페이지에서 기본값을 유지하고 [Create VPC]를 선택합니다.
- e. 나타나는 확인 메시지에서 [Close]를 선택합니다.
- 3. Amazon VPC에 두 개의 서브넷(퍼블릭 서브넷, 프라이빗 서브넷)이 있는지 확인합니다. 이러한 서브넷은 자동으로 생성됩니다.
 - a. 탐색 목록에서 [Subnets]를 선택합니다.
 - b. 서브넷 목록에서 Amazon VPC에 두 개의 서브넷이 있는지 찾아봅니다.

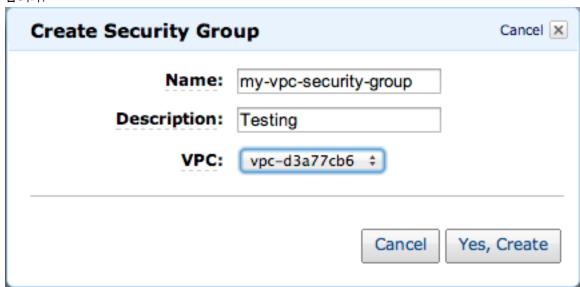


마법사는 프라이빗 서브넷에서 인터넷으로의 아웃바운드 통신을 위해 Amazon EC2 NAT 인스턴스 및 탄력적 IP 주소(Amazon EC2 요금 적용)를 생성하기 때문에 퍼블릭 서브넷은 사용 가능한 IP 주소 수가 하나 더 적습니다.

Tip

서브넷 식별자 2개(퍼블릭 서브넷, 프라이빗 서브넷)를 기록해 둡니다. 추후 캐시 클러스터를 시작하고 Amazon VPC에 Amazon EC2 인스턴스를 추가할 때 이 정보가 필요합니다.

- 4. Amazon VPC 보안 그룹을 생성합니다. 캐시 클러스터 및 Amazon EC2 인스턴스에 이 그룹을 사용합니다.
 - a. Amazon VPC 관리 콘솔의 탐색 창에서 보안 그룹을 선택합니다.
 - b. [Create Security Group]을 선택합니다.
 - c. 보안 그룹의 이름과 설명을 해당 상자에 입력합니다. VPC 상자에서 Amazon VPC의 식별자를 선택합니다.



d. 원하는 대로 설정되었으면 [Yes, Create]를 선택합니다.

- 5. 보안 그룹의 네트워크 수신 규칙을 정의합니다. 이 규칙을 사용하면 SSH(Secure Shell)를 사용하여 Amazon EC2 인스턴스에 연결할 수 있습니다.
 - a. 탐색 목록에서 [Security Groups]를 선택합니다.
 - b. 목록에서 보안 그룹을 찾아 선택합니다.
 - c. [Security Group] 아래에서 [Inbound] 탭을 선택합니다. [Create a new rule] 상자에서 [SSH]를 선택한 다음 [Add Rule]을 선택합니다.
 - d. [Apply Rule Changes]를 선택합니다.

이제 캐시 서브넷 그룹을 생성하고 Amazon VPC에서 캐시 클러스터를 시작할 수 있습니다.

캐시 서브넷 그룹 생성

캐시 서브넷 그룹은 Amazon VPC에서 캐시 클러스터에 대해 지정할 수 있는 서브넷 모음입니다. Amazon VPC에서 캐시 클러스터를 시작할 때 캐시 서브넷 그룹을 선택해야 합니다. 그러면 ElastiCache가 해당 캐시 서브넷 그룹을 사용하여 해당 서브넷의 IP 주소를 클러스터의 각 캐시 노드에 할당합니다.

ElastiCache 관리 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 서브넷 그룹을 생성하는 방법에 관한 지침은 서브넷 그룹 생성 (p. 329) 단원을 참조하십시오.

캐시 서브넷 그룹을 생성하면 Amazon VPC에서 실행할 캐시 클러스터를 시작할 수 있습니다. 다음 항목인 Amazon VPC에서 캐시 클러스터 생성 (p. 328) 단원으로 이동합니다.

Amazon VPC에서 캐시 클러스터 생성

이 예제에서는 Amazon VPC에서 캐시 클러스터를 생성합니다.

Amazon VPC (콘솔)에서 캐시 클러스터 생성

Redis 캐시 클러스터를 시작하려면 Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69)을 참조하십시오. 6.d단계에서 VPC 서브넷 그룹을 선택합니다.

Amazon VPC 내에서 캐시 클러스터를 시작했습니다. Amazon VPC에서 실행 중인 새 캐시 클러스터에 연결 하는 방법의 예는 Amazon VPC에서 실행되는 클러스터 또는 복제 그룹 에 연결 (p. 328) 단원을 참조하십시오.

Amazon VPC에서 복제 그룹 생성

이 예제에서는 Amazon VPC에서 Redis 복제 그룹을 생성합니다.

Amazon VPC에서 복제 그룹 생성 (콘솔)

VPC에서 Redis(클러스터 모드 비활성화됨) 복제 그룹을 시작하려면 Redis(클러스터 모드 비활성화됨) Redis 복제 그룹을 처음부터 새로 생성 (p. 148) 단원을 참조하여 5.b단계의 VPC 서브넷 그룹을 선택합니다.

VPC에서 Redis(클러스터 모드 활성화됨) 복제 그룹을 시작하려면 Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 154) 단원을 참조하여 6.d 단계의 VPC 서브넷 그룹을 선택합니다.

Amazon VPC 내에서 Redis 복제 그룹을 시작했습니다. Amazon VPC에서 실행 중인 새 복제 그룹에 연결하는 방법의 예는 Amazon VPC에서 실행되는 클러스터 또는 복제 그룹 에 연결 (p. 328) 단원을 참조하십시오.

Amazon VPC에서 실행되는 클러스터 또는 복제 그룹 에 연결

이 예제에서는 Amazon VPC에서 Amazon EC2 인스턴스를 시작하는 방법을 소개합니다. 이 인스턴스에 로그인하고 Amazon VPC에서 실행 중인 ElastiCache 클러스터를 액세스할 수 있습니다.

Note

Amazon EC2 사용에 대한 자세한 내용은 Amazon EC2 설명서의 Amazon EC2 Getting Started Guide를 참조하십시오.

Important

AWS 계정에 추가 요금이 발생하는 것을 방지하기 위해 이 예제를 실행한 후에 더 이상 필요하지 않은 AWS 리소스를 삭제하십시오.

클러스터 연결에 대한 자세한 내용은 ElastiCache 사용 설명서의 3단계: 클러스터 노드에 연결 (p. 23) 단원을 참조하십시오.

서브넷 및 서브넷 그룹

서브넷 그룹은 Amazon Virtual Private Cloud (VPC) 환경에서 실행 중인 클러스터에 대해 지정할 수 있는 서 브넷(일반적으로 프라이빗 서브넷) 모음입니다.

Amazon VPC에서 클러스터를 생성할 경우 캐시 서브넷 그룹을 지정해야 합니다. ElastiCache는 해당 캐시 서브넷 그룹을 사용하여 캐시 노드에 연결된 서브넷 내의 서브넷 및 IP 주소를 선택합니다.

이 섹션에서는 서브넷과 서브넷 그룹을 생성 및 활용하여 ElastiCache 리소스에 대한 액세스를 관리하는 방법을 알아봅니다.

Amazon VPC 환경에서 서브넷 그룹 사용에 대한 자세한 내용은 2단계: 액세스 자격 부여 (p. 23) 단원을 참조 하십시오.

주제

- 서브넷 그룹 생성 (p. 329)
- 클러스터 또는 복제 그룹에 서브넷 그룹 지정 (p. 331)
- 서브넷 그룹 수정 (p. 332)
- 서브넷 그룹 삭제 (p. 333)

서브넷 그룹 생성

새 서브넷 그룹을 생성할 때 사용 가능한 IP 주소의 수를 기록하십시오. 서브넷에 무료 IP 주소가 거의 없는 경우 클러스터에 추가할 수 있는 노드의 수가 제약될 수 있습니다. 이 문제를 해결하기 위해 클러스터의 가용 영역에 충분한 수의 IP 주소가 있도록 서브넷 그룹에 하나 이상의 서브넷을 지정할 수 있습니다. 그 이후 클러스터에 더 많은 노드를 추가할 수 있습니다.

다음 절차는 mysubnetgroup (console), AWS CLI 및 ElastiCache API라는 서브넷 그룹을 생성하는 방법을 보여줍니다.

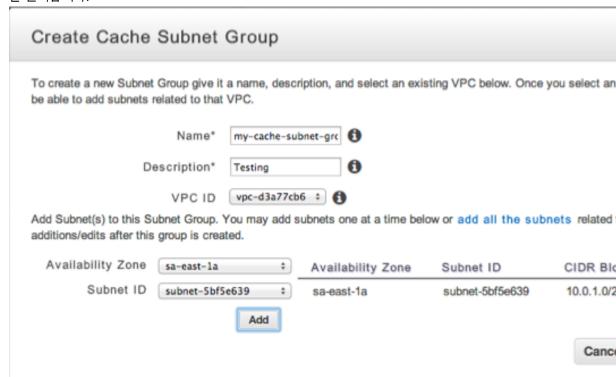
서브넷 그룹 생성 (콘솔)

다음 절차는 서브넷 그룹 (console)을 생성하는 방법을 보여줍니다.

서브넷 그룹을 생성하려면 (콘솔)

- 1. AWS Management Console에 로그인한 후 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 목록에서 [Subnet Groups]를 선택합니다.
- 3. [Create Subnet Group]을 선택합니다.
- 4. [Create Subnet Group] 마법사에서 다음을 수행합니다. 모든 설정이 원하는 대로 설정되었으면 [Yes, Create]를 선택합니다.

- a. [Name] 상자에 서브넷 그룹의 이름을 입력합니다.
- b. [Description] 상자에 서브넷 그룹에 대한 설명을 입력합니다.
- c. VPC ID 상자에서 생성한 Amazon VPC를 선택합니다.
- d. [Availability Zone] 및 [Subnet ID] 목록에서 프라이빗 서브넷의 ID와 가용 영역을 선택한 다음 [Add]를 선택합니다.



5. 나타나는 확인 메시지에서 [Close]를 선택합니다.

새 서브넷 그룹이 ElastiCache 콘솔의 서브넷 그룹 목록에 나타납니다. 창 하단에서 서브넷 그룹을 선택하여 이 그룹과 연결된 모든 서브넷 등의 상세 내용을 확인할 수 있습니다.

서브넷 그룹 생성 (AWS CLI)

명령 프롬프트에서 create-cache-subnet-group 명령을 사용하여 서브넷 그룹을 생성합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache create-cache-subnet-group \
    --cache-subnet-group-name mysubnetgroup \
    --cache-subnet-group-description "Testing" \
    --subnet-ids subnet-53df9c3a
```

Windows의 경우:

```
aws elasticache create-cache-subnet-group ^
--cache-subnet-group-name mysubnetgroup ^
--cache-subnet-group-description "Testing" ^
--subnet-ids subnet-53df9c3a
```

이 명령은 다음과 유사한 출력을 생성합니다.

자세한 내용은 AWS CLI 항목 create-cache-subnet-group를 참조하십시오.

서브넷 그룹 생성 (ElastiCache API)

ElastiCache API를 사용하여 다음 파라미터와 함께 CreateCacheSubnetGroup을 호출합니다.

- CacheSubnetGroupName=mysubnetgroup
- CacheSubnetGroupDescription==Testing
- SubnetIds.member.1=subnet-53df9c3a

Example

```
https://elasticache.us-west-2.amazonaws.com/
?Action=CreateCacheSubnetGroup
&CacheSubnetGroupDescription=Testing
&CacheSubnetGroupName=mysubnetgroup
&SignatureMethod=HmacSHA256
&SignatureVersion=4
&SubnetIds.member.1=subnet-53df9c3a
&Timestamp=20141201T220302Z
&Version=2014-12-01
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=<credential>
&X-Amz-Date=20141201T220302Z
&X-Amz-Date=20141201T220302Z
&X-Amz-Expires=20141201T220302Z
&X-Amz-Signature=<signature>
&X-Amz-Signature=<signature>
&X-Amz-SignedHeaders=Host
```

클러스터 또는 복제 그룹에 서브넷 그룹 지정

서브넷 그룹을 생성한 후 Amazon VPC에서 클러스터 또는 복제 그룹을 시작할 수 있습니다. 자세한 내용은 다음을 참조하십시오.

- 독립 실행형 Redis 클러스터 단일 노드 Redis 클러스터를 시작하려면 Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69) 단원을 참조하십시오. 5.a 단계(고급 Redis 설정)에서 VPC 서브넷 그룹을 선 택합니다.
- Redis(클러스터 모드 비활성화됨) 복제 그룹 VPC에서 Redis(클러스터 모드 비활성화됨) 복제 그룹을 시작하려면 Redis(클러스터 모드 비활성화됨) Redis 복제 그룹을 처음부터 새로 생성 (p. 148) 단원을 참조하십시오. 5.b 단계(고급 Redis 설정)에서 VPC 서브넷 그룹을 선택합니다.
- Redis(클러스터 모드 활성화됨) 복제 그룹 Redis(클러스터 모드 활성화됨) 클러스터 생성 (콘솔) (p. 154). 5.a 단계(고급 Redis 설정)에서 VPC 서브넷 그룹을 선택합니다.

서브넷 그룹 수정

서브넷 그룹의 설명을 수정하거나 서브넷 그룹과 연관된 서브넷 ID의 목록을 수정할 수 있습니다. 클러스터 가 현재 해당 서브넷을 사용하고 있는 경우 서브넷 그룹에서 서브넷 ID를 삭제할 수 없습니다.

다음 절차는 서브넷 그룹을 수정하는 방법을 보여줍니다.

서브넷 그룹 수정 (콘솔)

서브넷 그룹을 수정하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Subnet Groups]를 선택합니다.
- 3. 서브넷 그룹 목록에서 수정하려는 서브넷 그룹을 선택합니다.
- 4. ElastiCache 콘솔의 하단부에서 설명 또는 서브넷 그룹에 대한 서브넷 ID의 목록을 변경합니다. 변경 사항을 저장하려면 [Save]를 선택합니다.

서브넷 그룹 수정 (AWS CLI)

명령 프롬프트에서 modify-cache-subnet-group 명령을 사용하여 서브넷 그룹을 수정합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-subnet-group \
    --cache-subnet-group-name mysubnetgroup \
    --cache-subnet-group-description "New description" \
    --subnet-ids "subnet-42df9c3a" "subnet-48fc21a9"
```

Windows의 경우:

```
aws elasticache modify-cache-subnet-group ^
--cache-subnet-group-name mysubnetgroup ^
--cache-subnet-group-description "New description" ^
--subnet-ids "subnet-42df9c3a" "subnet-48fc21a9"
```

이 명령은 다음과 유사한 출력을 생성합니다.

}

자세한 내용은 AWS CLI 항목 modify-cache-subnet-group를 참조하십시오.

서브넷 그룹 수정 (ElastiCache API)

ElastiCache API를 사용하여 다음 파라미터와 함께 ModifyCacheSubnetGroup을 호출합니다.

- CacheSubnetGroupName=mysubnetgroup
- 변경할 값이 있는 다른 파라미터. 이 예제는 CacheSubnetGroupDescription=New %20description을 사용하여 서브넷 그룹의 설명을 변경합니다.

Example

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=ModifyCacheSubnetGroup
    &CacheSubnetGroupDescription=New%20description
    &CacheSubnetGroupName=mysubnetgroup
    &SubnetIds.member.1=subnet-42df9c3a
    &SubnetIds.member.2=subnet-48fc21a9
    &SignatureMethod=HmacSHA256
    &SignatureVersion=4
    &Timestamp=20141201T220302Z
    &Version=2014-12-01
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Credential = < credential >
    &X-Amz-Date=20141201T220302Z
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Signature=<signature>
    &X-Amz-SignedHeaders=Host
```

Note

새 서브넷 그룹을 생성할 때 사용 가능한 IP 주소의 수를 기록하십시오. 서브넷에 무료 IP 주소가 거의 없는 경우 클러스터에 추가할 수 있는 노드의 수가 제약될 수 있습니다. 이 문제를 해결하기 위해 클러스터의 가용 영역에 충분한 수의 IP 주소가 있도록 서브넷 그룹에 하나 이상의 서브넷을 지정할수 있습니다. 그 이후 클러스터에 더 많은 노드를 추가할 수 있습니다.

서브넷 그룹 삭제

서브넷 그룹이 더 이상 필요하지 않다고 판단되면 삭제할 수 있습니다. 현재 클러스터에서 사용 중인 경우 서 브넷 그룹을 삭제할 수 없습니다.

다음 절차는 서브넷 그룹을 삭제하는 방법을 보여줍니다.

서브넷 그룹 삭제 (콘솔)

서브넷 그룹을 삭제하는 방법

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Subnet Groups]를 선택합니다.
- 3. 서브넷 그룹 목록에서 삭제할 서브넷 그룹을 선택한 다음 [Delete]를 선택합니다.
- 4. 이 작업을 확인하라는 메시지가 표시되면 [Yes, Delete]를 선택합니다.

서브넷 그룹 삭제 (AWS CLI)

AWS CLI를 사용하여 다음 파라미터로 delete-cache-subnet-group 명령을 호출할 수 있습니다.

• --cache-subnet-group-name mysubnetgroup

Linux. macOS. or Unix의 경우:

```
aws elasticache delete-cache-subnet-group \
--cache-subnet-group-name mysubnetgroup
```

Windows의 경우:

```
aws elasticache delete-cache-subnet-group ^
--cache-subnet-group-name mysubnetgroup
```

이 명령은 출력을 생성하지 않습니다.

자세한 내용은 AWS CLI 항목 delete-cache-subnet-group를 참조하십시오.

서브넷 그룹 삭제 (ElastiCache API)

ElastiCache API를 사용하여 다음 파라미터와 함께 DeleteCacheSubnetGroup을 호출합니다.

• CacheSubnetGroupName=mysubnetgroup

Example

읽기 쉽도록 줄바꿈이 추가됩니다.

이 명령은 출력을 생성하지 않습니다.

자세한 내용은 ElastiCache API 항목 DeleteCacheSubnetGroup을 참조하십시오.

보안 그룹: EC2-Classic

Important

Amazon ElastiCache 보안 그룹은 Amazon Virtual Private Cloud 환경(VPC)에서 실행되지 않는 클러스터에만 적용됩니다. Amazon Virtual Private Cloud를 실행하는 경우 콘솔 탐색 창에서 보안 그룹을 사용할 수 없습니다.

Amazon VPC에서 ElastiCache 노드를 실행하는 경우, ElastiCache 보안 그룹과 다른 Amazon VPC 보안 그룹으로 클러스터에 대한 액세스를 제어합니다. Amazon VPC에서 ElastiCache를 사용하는 방법에 대한 자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서

보안 그룹: EC2-Classic

Amazon ElastiCache에서는 ElastiCache 보안 그룹을 통해 클러스터에 대한 액세스를 제어할 수 있습니다. ElastiCache 보안 그룹은 클러스터에 대한 네트워크 액세스를 제어하는 방화벽 역할을 합니다. 기본적으로 네트워크 액세스는 클러스터에서 해제됩니다. 애플리케이션이 클러스터에 액세스하도록 하려면 특정 Amazon EC2 보안 그룹의 호스트에서 명시적으로 액세스를 활성화해야 합니다. 수신 규칙이 구성되면 동일한 규칙이 해당 보안 그룹과 관련된 모든 클러스터에 적용됩니다.

클러스터에 대한 네트워크 액세스를 허용하려면, 보안 그룹을 생성하고

AuthorizeCacheSecurityGroupIngress API 작업(CLI: authorize-cache-security-group-ingress)을 사용하여 원하는 Amazon EC2 보안 그룹에 권한을 부여합니다(그런 다음 허용된 Amazon EC2 인스턴스를 지정). 보안 그룹은 생성 시 또는 ModifyCacheCluster API 작업(CLI: modify-cache-cluster)을 사용하여 클러스터에 연결할 수 있습니다.

Important

IP 범위 기반 액세스 제어는 현재 개별 클러스터 수준에서 비활성화되어 있습니다. 클러스터에 액세 스하는 모든 클라이언트는 EC2 네트워크 내에 있어야 하며, 이전에 설명한 대로 보안 그룹을 통해 권한을 부여받아야 합니다.

Amazon VPC에서 ElastiCache를 사용하는 방법에 대한 자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

Amazon VPC에서 실행되는 Amazon EC2 인스턴스는 EC2-Classic의 ElastiCache 클러스터에 연결할 수 없습니다.

주제

- 보안 그룹 생성 (p. 335)
- 사용 가능한 보안 그룹 나열 (p. 336)
- 보안 그룹 보기 (p. 338)
- Amazon EC2 보안 그룹에 대한 네트워크 액세스 권한 부여 (p. 339)

보안 그룹 생성

이 주제는 Amazon VPC에서 실행하지 않는 경우에만 관련이 있습니다. Amazon VPC에서 실행하는 경우 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

보안 그룹을 생성하려면 이름과 설명을 제공해야 합니다.

다음 절차는 새 보안 그룹을 생성하는 방법을 보여줍니다.

보안 그룹 생성 (콘솔)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Security Groups]를 선택합니다.
- 3. [Create Security Group]을 선택합니다.
- 4. [Create Security Group]의 [Security Group]에 새 보안 그룹의 이름을 입력합니다.
- 5. [Description]에 새 보안 그룹에 대한 설명을 입력합니다.
- 6. Create를 선택합니다.

보안 그룹 생성 (AWS CLI)

명령 프롬프트에서 다음 파라미터와 함께 create-cache-security-group 명령을 사용합니다.

• --cache-security-group-name – 생성 중인 보안 그룹의 이름입니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 보안 그룹: EC2-Classic

예: mysecuritygroup

• --description – 이 보안 그룹에 대한 설명입니다.

예: "나의 새 보안 그룹"

Linux, macOS, or Unix의 경우:

```
aws elasticache create-cache-security-group \
    --cache-security-group-name mysecuritygroup \
    --description "My new security group"
```

Windows의 경우:

```
aws elasticache create-cache-security-group ^
    --cache-security-group-name mysecuritygroup ^
    --description "My new security group"
```

자세한 내용은 create-cache-security-group 단원을 참조하십시오.

보안 그룹 생성 (ElastiCache API)

ElastiCache API 작업 CreateCacheSecurityGroup을 다음 파라미터와 함께 사용:

• CacheSecurityGroupName – 생성 중인 보안 그룹의 이름입니다.

예: mysecuritygroup

• Description - 이 보안 그룹에 대한 URL로 인코딩된 설명입니다.

예: My%20security%20group

Example

읽기 쉽도록 줄바꿈이 추가됩니다.

```
https://elasticache.us-west-2.amazonaws.com /
    ?Action=CreateCacheSecurityGroup
    &CacheSecurityGroupName=mysecuritygroup
    &Description=My%20security%20group
    &Version=2015-02-02
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20150202T220302Z
    &X-Amz-Date=20150202T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20150202T220302Z
    &X-Amz-Credential=<credential>
    &X-Amz-Signature=<signature>
```

사용 가능한 보안 그룹 나열

이 주제는 Amazon VPC에서 실행하지 않는 경우에만 관련이 있습니다. Amazon VPC에서 실행하는 경우 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

AWS 계정에 대해 생성된 보안 그룹을 나열할 수 있습니다.

다음 절차는 AWS 계정에 대해 사용할 수 있는 보안 그룹을 나열하는 방법을 보여줍니다.

사용 가능한 보안 그룹 나열 (콘솔)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Security Groups]를 선택합니다.

사용 가능한 보안 그룹이 [Security Groups] 목록에 나타납니다.

사용 가능한 보안 그룹 나열 (AWS CLI)

명령 프롬프트에서 describe-cache-security-groups 명령을 사용하여 AWS 계정에 사용할 수 있는 모든 보안 그룹을 나열합니다.

```
aws elasticache describe-cache-security-groups
```

이 명령의 JSON 출력은 다음과 같습니다.

자세한 내용은 describe-cache-security-groups을 참조하십시오.

사용 가능한 보안 그룹 나열 (ElastiCache API)

ElastiCache API를 사용하여 DescribeCacheSecurityGroups를 호출합니다.

Example

읽기 쉽도록 줄바꿈이 추가됩니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=DescribeCacheSecurityGroups
    &MaxRecords=100
    &Version=2015-02-02
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &XY-Amz-Date=20150202T220302Z
    &XY-Amz-Date=20150202T220302Z
    &XY-Amz-SignedHeaders=Host
    &XY-Amz-Expires=20150202T220302Z
    &XY-Amz-Credential=<credential>
```

&X-Amz-Signature=<signature>

보안 그룹 보기

이 주제는 Amazon VPC에서 실행하지 않는 경우에만 관련이 있습니다. Amazon VPC에서 실행하는 경우 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

보안 그룹에 대한 세부 정보를 볼 수 있습니다.

다음 절차는 ElastiCache 콘솔, AWS CLI 및 ElastiCache API를 사용하여 보안 그룹의 속성을 보는 방법을 보여줍니다.

보안 그룹 보기 (콘솔)

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Security Groups]를 선택합니다.

사용 가능한 캐시 보안 그룹이 [Security Groups] 목록에 나타납니다.

3. [Security Groups] 목록에서 캐시 보안 그룹을 선택합니다.

보안 그룹에 대해 정의된 권한 부여의 목록이 창 하단의 세부 정보 단원에 표시됩니다.

보안 그룹 보기 (AWS CLI)

명령 프롬프트에서 보려는 보안 그룹의 이름과 함께 AWS CLI describe-cache-security-groups 명령을 사용합니다.

• --cache-security-group-name – 세부 정보가 반환되는 보안 그룹의 이름입니다.

aws elasticache describe-cache-security-groups --cache-security-group-name mysecuritygroup

이 명령의 JSON 출력은 다음과 같습니다.

자세한 내용은 describe-cache-security-groups 단원을 참조하십시오.

보안 그룹 보기 (ElastiCache API)

ElastiCache API를 사용하여 보려는 보안 그룹의 이름으로 DescribeCacheSecurityGroups를 호출합니다.

• CacheSecurityGroupName – 세부 정보가 반환되는 캐시 보안 그룹의 이름입니다.

Example

읽기 쉽도록 줄바꿈이 추가됩니다.

Amazon EC2 보안 그룹에 대한 네트워크 액세스 권한 부여

이 주제는 Amazon VPC에서 실행하지 않는 경우에만 관련이 있습니다. Amazon VPC에서 실행하는 경우 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.

Amazon EC2 인스턴스에서 클러스터에 액세스하려면 EC2 인스턴스가 속한 Amazon EC2 보안 그룹에 대한 액세스 권한을 부여해야 합니다. 다음 절차는 Amazon EC2 보안 그룹에 대한 액세스 권한을 부여하는 방법을 보여줍니다.

Important

- Amazon EC2 보안 그룹에 대한 권한 부여는 Amazon EC2 보안 그룹에 속한 모든 EC2 인스턴스의 클러스터에 대한 액세스 권한만 부여합니다.
- 액세스 권한 변경이 적용되는 데는 약 1분이 걸립니다.

Amazon EC2 보안 그룹에 대한 네트워크 액세스 권한 부여(콘솔)

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 [Security Groups]를 선택합니다.
- 3. [Security Groups] 목록에서 액세스 권한을 부여할 보안 그룹의 왼쪽에 있는 상자를 선택합니다.
- 4. 창 하단에 있는 [EC2 Security Group Name] 목록에서 Amazon EC2 보안 그룹을 선택합니다.
- 5. [추가]를 선택합니다.

Amazon EC2 보안 그룹에 대한 네트워크 액세스 권한 부여(AWS CLI)

명령 프롬프트에서 다음 파라미터와 함께 authorize-cache-security-group-ingress 명령을 사용 하여 Amazon EC2 보안 그룹에 대한 액세스 권한을 부여합니다.

- --cache-security-group-name Amazon EC2 액세스 권한을 부여할 보안 그룹의 이름입니다.
- --ec2-security-group-name Amazon EC2 인스턴스가 속한 Amazon EC2 보안 그룹의 이름입니다.
- --ec2-security-group-owner-id-Amazon EC2 보안 그룹의 소유자 ID입니다.

Example

Linux, macOS, or Unix의 경우:

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 보안 그룹: EC2-Classic

```
aws elasticache authorize-cache-security-group-ingress \
--cache-security-group-name default \
--ec2-security-group-name myec2group \
--ec2-security-group-owner-id 987654321021
```

Windows의 경우:

```
aws elasticache authorize-cache-security-group-ingress ^
--cache-security-group-name default ^
--ec2-security-group-name myec2group ^
--ec2-security-group-owner-id 987654321021
```

다음과 비슷한 출력이 생성되어야 합니다.

자세한 내용은 authorize-cache-security-group-ingress 단원을 참조하십시오.

Amazon EC2 보안 그룹에 대한 네트워크 액세스 권한 부여(ElastiCache API)

ElastiCache API를 사용하여 다음 파라미터와 함께 AuthorizeCacheSecurityGroupIngress을 호출합니다.

- CacheSecurityGroupName Amazon EC2 액세스 권한을 부여할 보안 그룹의 이름입니다.
- EC2SecurityGroupName Amazon EC2 인스턴스가 속한 Amazon EC2 보안 그룹의 이름입니다.
- EC2SecurityGroupOwnerId Amazon EC2 보안 그룹의 소유자 ID입니다.

Example

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=AuthorizeCacheSecurityGroupIngress
    &EC2SecurityGroupOwnerId=987654321021
    &EC2SecurityGroupName=myec2group
    &Version=2015-02-02
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20150202T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20150202T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20150202T220302Z
    &X-Amz-Expires=20150202T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Signature
```

자세한 내용은 AuthorizeCacheSecurityGroupIngress 단원을 참조하십시오.

Amazon ElastiCache의 Identity and Access Management(IAM)

Amazon ElastiCache에 액세스하려면 AWS가 요청을 인증하는 데 사용할 수 있는 자격 증명이 필요합니다. 이러한 자격 증명에는 ElastiCache 캐시 클러스터 또는 Amazon Elastic Compute Cloud (Amazon EC2) 인스턴스 같은 AWS 리소스에 액세스할 수 있는 권한이 있어야 합니다. 다음 단원에서는 리소스에 액세스할 수 있는지 대상을 제어하여 리소스를 보호할 수 있도록 AWS Identity and Access Management(IAM) 및 ElastiCache를 사용하는 방법에 대한 세부 정보를 제공합니다.

- 인증 (p. 341)
- 액세스 제어 (p. 342)

인증

다음과 같은 ID 유형으로 AWS에 액세스할 수 있습니다.

- AWS 계정 루트 사용자 AWS 계정을 처음 생성할 때는 해당 계정의 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 SSO(Single Sign-In) ID로 시작합니다. 이 자격 증명은 AWS 계정 루트 사용 자라고 하며, 계정을 생성할 때 사용한 이메일 주소와 암호로 로그인하여 액세스합니다. 일상적인 작업은 물론 관리 작업에도 루트 사용자를 사용하지 않는 것이 좋습니다. 대신 IAM 사용자를 처음 생성할 때만 루트 사용자를 사용하는 모범 사례를 준수하십시오. 그런 다음 루트 사용자 자격 증명을 안전하게 보관해 두고 몇 가지 계정 및 서비스 관리 작업을 수행할 때만 해당 자격 증명을 사용합니다.
- IAM 사용자 IAM 사용자는 특정 사용자 지정 권한(예: ElastiCache에서 a cluster을 생성할 권한)이 있는 AWS 계정 내 자격 증명입니다. IAM 사용자 이름과 암호를 사용하여 AWS Management 콘솔, AWS 토론 포럼 또는 AWS Support Center 같은 보안 AWS 웹 페이지에 로그인할 수 있습니다.

사용자 이름과 암호 외에도 각 사용자에 대해 액세스 키를 생성할 수 있습니다. 여러 SDK 중 하나를 통해 또는 AWS Command Line Interface(CLI)를 사용하여 AWS 제품에 프로그래밍 방식으로 액세스할 때 이러한 키를 사용할 수 있습니다. SDK 및 CLI 도구는 액세스 키를 사용하여 암호화 방식으로 요청에 서명합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. ElastiCache supports는 인바운드 API 요청을 인증하기 위한 프로토콜인 서명 버전 4를 지원합니다. 요청 인증에 대한 자세한 내용은 AWS General Reference의 서명 버전 4 서명 프로세스 단원을 참조하십시오.

- IAM 역할 IAM 역할은 특정 권한을 가진 계정에서 생성할 수 있는 IAM 자격 증명입니다. IAM 역할은 AWS에서 자격 증명으로 할 수 있는 것과 할 수 없는 것을 결정하는 권한 정책을 포함하는 AWS 자격 증명이라는 점에서 IAM 사용자와 유사합니다. 그러나 역할은 한 사람과만 연관되지 않고 해당 역할이 필요한 사람이라면 누구든지 말을 수 있어야 합니다. 또한 역할에는 그와 연관된 암호 또는 액세스 키와 같은 표준 장기 자격 증명이 없습니다. 대신에 역할을 수임한 사람에게는 해당 역할 세션을 위한 임시 보안 자격 증명이 제공됩니다. 임시 자격 증명이 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.
 - 연합된 사용자 액세스 IAM 사용자를 생성하는 대신 AWS Directory Service의 기존 ID, 엔터프라이 즈 사용자 디렉터리 또는 웹 ID 공급자를 사용할 수 있습니다. 이 사용자를 연합된 사용자라고 합니다. AWS에서는 ID 공급자를 통해 액세스가 요청되면 연합된 사용자에게 역할을 할당합니다. 연합된 사용자에 대한 자세한 내용은 IAM 사용 설명서의 연합된 사용자 및 역할을 참조하십시오.
 - AWS 서비스 액세스 서비스 역할은 서비스가 사용자를 대신하여 사용자 계정에서 작업을 수행하기 위해 수임하는 IAM 역할입니다. 일부 AWS 서비스 환경을 설정할 때 서비스에서 맡을 역할을 정의해야 합니다. 이 서비스 역할에는 서비스가 AWS 리소스에 액세스하는 데 필요한 모든 권한이 포함되어야 합니

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 액세스 제어

다. 서비스 역할은 서비스마다 다르지만 해당 서비스에 대한 문서화된 요구 사항을 충족하는 한 대부분의 경우 권한을 선택할 수 있습니다. 서비스 역할은 해당 계정 내 액세스 권한만 제공하며 다른 계정의서비스에 대한 액세스 권한을 부여하는 데 사용될 수 없습니다. IAM 내에서 서비스 역할을 생성, 수정 및삭제할 수 있습니다. 예를 들어 Amazon Redshift에서 사용자 대신 Amazon S3 버킷에 액세스하도록 허용하는 역할을 생성한 후 해당 버킷에 있는 데이터를 Amazon Redshift 클러스터로 로드할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성을 참조하십시오.

• Amazon EC2에서 실행 중인 애플리케이션 – IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 AWS CLI 또는 AWS API 요청을 수행하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. 이는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 해당 역할을 모든 애플리케이션에서 사용할 수 있도록 하려면 인스턴스에 연결된 인스턴스 프로파일을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 자격 증명을 얻을 수 있습니다. 자세한 내용은 IAM 사용 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하십시오.

액세스 제어

요청을 인증할 수 있는 유효한 자격 증명이 있더라도 권한이 없다면 Amazon ElastiCache 리소스를 생성하거나 액세스할 수 없습니다. 예를 들어. ElastiCache 캐시 클러스터를 생성할 권한이 있어야 합니다.

다음 단원에서는 Amazon ElastiCache에 대한 권한을 관리하는 방법을 설명합니다. 먼저 개요를 읽어 보면 도움이 됩니다.

- ElastiCache 리소스에 대한 액세스 권한 관리 개요 (p. 342)
- Amazon ElastiCache에 대한 자격 증명 기반 정책(IAM 정책) 사용 (p. 345)

ElastiCache 리소스에 대한 액세스 권한 관리 개요

모든 AWS 리소스는 AWS 계정의 소유이고, 리소스 생성 또는 리소스 액세스 권한은 권한 정책에 따라 결정됩니다. 계정 관리자는 IAM 자격 증명(사용자, 그룹 및 역할)에 권한 정책을 연결할 수 있습니다. 또한, AWS Lambda 같은 일부 서비스에서도 권한 정책을 리소스에 연결할 수 있습니다.

Note

계정 관리자 또는 관리자 사용자는 관리자 권한이 있는 사용자입니다. 자세한 정보는 IAM 사용 설명 서에서 IAM 모범 사례를 참조하십시오.

사용 권한을 부여할 때 누구에게 권한을 부여할지 결정합니다. 또한 권한 대상이 되는 리소스와 해당 리소스에 허용되는 특정 작업을 결정합니다.

주제

- Amazon ElastiCache 리소스 및 작업 (p. 342)
- 리소스 소유권 이해 (p. 343)
- 리소스 액세스 관리 (p. 343)
- 정책 요소 지정: 작업, 효과, 리소스, 보안 주체 (p. 344)
- 정책에서 조건 지정 (p. 345)

Amazon ElastiCache 리소스 및 작업

Amazon ElastiCache에서는 캐시 클러스터가 기본 리소스입니다.

다음 표에 나와 있는 것처럼 이러한 리소스에는 고유한 Amazon 리소스 이름(ARN)이 연결됩니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 액세스 관리 개요

리소스 유형	ARN 형식
캐시 클러스터	arn:aws:elasticache:region:account-id:cluster:resource-name

ElastiCache의 경우 ElastiCache 리소스를 처리하기 위한 작업을 제공합니다. 사용 가능한 작업 목록은 Amazon ElastiCache 작업을 참조하십시오.

리소스 소유권 이해

리소스 소유자는 해당 리소스를 만든 AWS 계정입니다. 즉, 리소스 소유자는 리소스를 생성하는 요청을 인증하는 보안 주체 엔터티의 AWS 계정입니다. 보안주체 엔터티는 루트 계정, IAM 사용자 또는 IAM 역할일 수있습니다. 다음 예에서는 이 계정의 작동 방식을 설명합니다.

- AWS 계정의 루트 계정 자격 증명을 사용하여 캐시 클러스터를 생성한다고 가정합니다. 이 경우, AWS 계정이 리소스의 소유자입니다. ElastiCache에서 리소스는 캐시 클러스터입니다.
- AWS 계정에서 IAM 사용자를 생성하고 해당 사용자에게 캐시 클러스터 생성 권한을 부여한다고 가정합니다. 이러한 경우, 이 사용자가 캐시 클러스터를 생성할 수 있습니다. 하지만 캐시 클러스터 리소스는 사용자가 속한 AWS 계정의 소유입니다.
- AWS 계정에서 IAM 역할을 생성하고 해당 역할에게 캐시 클러스터 생성 권한을 부여한다고 가정합니다. 이러한 경우, 이 역할을 수임할 사람은 캐시 클러스터를 생성할 수 있습니다. 캐시 클러스터 리소스는 역할 이 속한 AWS 계정의 소유입니다.

리소스 액세스 관리

권한 정책은 누가 무엇에 액세스 할 수 있는지를 나타냅니다. 다음 단원에서는 권한 정책을 만드는 데 사용 가능한 옵션에 대해 설명합니다.

Note

이 단원에서는 Amazon ElastiCache의 맥락에서 IAM을 사용하는 방법에 대해 설명하며, IAM 서비스에 대한 자세한 내용은 다루지 않습니다. 전체 IAM 설명서는 IAM 사용 설명서의 IAM이란 무엇인가?를 참조하십시오. IAM 정책 구문과 설명에 대한 자세한 정보는 IAM 사용 설명서의 AWS IAM 정책 참조를 참조하십시오.

IAM 자격 증명에 연결된 정책은 자격 증명 기반 정책(IAM 정책)이라고 합니다. 리소스에 연결된 정책은 리소스 기반 정책이라고 합니다. Amazon ElastiCache는 자격 증명 기반 정책(IAM 정책)만 지원합니다.

주제

- 자격 증명 기반 정책(IAM 정책) (p. 343)
- 리소스 기반 정책 (p. 344)

자격 증명 기반 정책(IAM 정책)

정책을 IAM 자격 증명에 연결할 수 있습니다. 예를 들면,

- 계정 내 사용자 또는 그룹에 권한 정책 연결 계정 관리자는 권한을 부여하기 위해 특정 사용자에 연결된 권한 정책을 사용할 수 있습니다. 이 경우 해당 사용자는 캐시 클러스터, 파라미터 그룹 또는 보안 그룹과 같은 ElastiCache 리소스를 생성할 수 있습니다.
- 역할에 권한 정책 연결(교차 계정 권한 부여) 자격 증명 기반 권한 정책을 IAM 역할에 연결하여 교차 계정 권한을 부여할 수 있습니다. 예를 들어, 계정 A의 관리자는 다음과 같이 다른 AWS 계정(예: 계정 B) 또는 AWS 서비스에 교차 계정 권한을 부여할 역할을 생성할 수 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 액세스 관리 개요

- 1. 계정 A 관리자는 IAM 역할을 생성하고 계정 A의 리소스에 대한 권한을 부여하는 역할에 권한 정책을 연결합니다.
- 2. 계정 A 관리자는 계정 B를 역할을 수임할 보안 주체로 식별하는 역할에 신뢰 정책을 연결합니다.
- 3. 계정 B 관리자는 계정 B의 사용자에게 역할을 수임할 권한을 위임할 수 있습니다. 그러면 계정 B의 사용자가 계정 A에서 리소스를 생성하거나 액세스할 수 있습니다. 일부 경우에는, 역할에 할당할 AWS 서비스 권한을 부여하려고 할 수 있습니다. 이러한 접근 방식을 지원하려면, 신뢰 정책의 보안 주체는 AWS 서비스 보안 주체일 수도 있습니다.

IAM을 사용하여 권한을 위임하는 방법에 대한 자세한 정보는 IAM 사용 설명서의 액세스 관리를 참조하십시오.

다음은 사용자가 ASW 계정에 대해 DescribeCacheClusters 작업을 수행할 수 있도록 허용하는 정책 예제입니다. 현재 구현에서 ElastiCache에서는 API 작업에 대한 리소스 ARN을 사용하여 특정 리소스를 식별할 수 있도록 지원하지 않습니다. (이러한 접근 방식을 리소스 수준 권한이라고도 합니다.) 따라서 와일드카드 문자(*)를 지정해야 합니다.

```
"Version": "2012-10-17",
"Statement": [{
    "Sid": "DescribeCacheClusters",
    "Effect": "Allow",
    "Action": [
         "elasticache:DescribeCacheClusters"],
    "Resource": "*"
    }
]
```

ElastiCache에서 자격 증명 기반 정책을 사용하는 방법에 대한 자세한 내용은 Amazon ElastiCache에 대한 자격 증명 기반 정책(IAM 정책) 사용 (p. 345) 단원을 참조하십시오. 사용자, 그룹, 역할 및 권한에 대한 자세한 내용은 IAM 사용 설명서의 자격 증명(사용자, 그룹 및 역할)을 참조하십시오.

리소스 기반 정책

Amazon S3와 같은 다른 서비스도 리소스 기반 권한 정책을 지원합니다. 예를 들어, 정책을 S3 버킷에 연결하여 해당 버킷에 대한 액세스 권한을 관리할 수 있습니다. Amazon ElastiCache는 리소스 기반 정책을 지원하지 않습니다.

정책 요소 지정: 작업, 효과, 리소스, 보안 주체

각 Amazon ElastiCache 리소스(Amazon ElastiCache 리소스 및 작업 (p. 342) 참조)에 대해 서비스는 API 작업을 정의합니다(작업 참조). 이러한 API 작업에 대한 권한을 부여하기 위해 ElastiCache에서는 정책에서 지정할 수 있는 작업을 정의합니다. 예를 들어 ElastiCache 스냅샷 리소스에 대해 CreateCacheCluster, DeleteCacheCluster 및 DescribeCacheCluster 작업을 정의합니다. API 작업을 실시하려면 둘 이상의 작업에 대한 권한이 필요할 수 있습니다.

다음은 가장 기본적인 정책 요소입니다.

- 리소스 정책에서 Amazon 리소스 이름(ARN)을 사용하여 정책을 적용할 리소스를 식별합니다. ElastiCache 리소스의 경우에는 IAM 정책에 항상 와일드카드 문자(*)를 사용합니다. 자세한 내용은 Amazon ElastiCache 리소스 및 작업 (p. 342) 단원을 참조하십시오.
- 작업 작업 키워드를 사용하여 허용 또는 거부할 리소스 작업을 식별합니다. 예를 들어 지정한 Effect에 따라 elasticache:CreateCacheCluster 권한은 Amazon ElastiCache CreateCacheCluster 작업을 수행할 수 있는 사용자 권한을 허용하거나 거부합니다.
- 결과 사용자가 특정 작업을 요청하는 경우의 결과를 지정합니다. 이는 허용 또는 거부 중에 하나가 될 수 있습니다. 명시적으로 리소스에 대한 액세스 권한을 부여(허용)하지 않는 경우, 액세스는 묵시적으로 거부

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 자격 증명 기반 정책(IAM 정책) 사용

됩니다. 리소스에 대한 액세스를 명시적으로 거부할 수도 있습니다. 예를 들어, 다른 정책에서 액세스 권한을 부여하더라도 사용자가 해당 리소스에 액세스할 수 없도록 하려고 할 때 이러한 작업을 수행할 수 있습니다.

• 보안 주체 – 자격 증명 기반 정책(IAM 정책)에서 정책이 연결되는 사용자는 암시적인 보안 주체입니다. 리소스 기반 정책의 경우 사용자, 계정, 서비스 또는 권한의 수신자인 기타 개체를 지정합니다(리소스 기반 정책에만 해당). ElastiCache의 경우 리소스 기반 정책을 지원하지 않습니다.

IAM 정책 구문과 설명에 대한 자세한 내용은 IAM 사용 설명서의 AWS IAM 정책 참조를 참조하십시오.

모든 Amazon ElastiCache API 작업을 보여 주는 표는 ElastiCache API 권한: 작업, 리소스 및 조건 참조 (p. 355) 단원을 참조하십시오.

정책에서 조건 지정

권한을 부여할 때 IAM 정책 언어를 사용하여 정책이 적용되는 조건을 지정할 수 있습니다. 예를 들어, 특정 날짜 이후에만 정책을 적용할 수 있습니다. 정책 언어에서의 조건 지정에 관한 자세한 내용은 IAM 사용 설명 서의 조건 단원을 참조하십시오.

조건을 표시하려면 미리 정의된 조건 키를 사용합니다. Amazon ElastiCache에만 해당되는 특정한 조건 키는 없습니다. 하지만 적절하게 사용할 수 있는 AWS 차원의 조건 키가 있습니다. AWS 차원 키의 전체 목록은 IAM 사용 설명서의 사용 가능한 조건 키 단원을 참조하십시오.

Amazon ElastiCache에 대한 자격 증명 기반 정책(IAM 정책) 사용

이 단원에서는 계정 관리자가 IAM 자격 증명(즉 사용자, 그룹, 역할)에 권한 정책을 연결할 수 있는 자격 증명 기반 정책의 예를 제시합니다.

Important

Amazon ElastiCache 리소스 액세스를 관리하기 위한 기본 개념과 옵션을 설명하는 주제를 먼저 읽어 보는 것이 좋습니다. 자세한 내용은 ElastiCache 리소스에 대한 액세스 권한 관리 개요 (p. 342) 단원을 참조하십시오.

이 주제의 섹션에서는 다음 내용을 학습합니다.

- Amazon ElastiCache 콘솔 사용에 필요한 권한 (p. 346)
- Amazon ElastiCache에 대한 AWS 관리형(미리 정의된) 정책 (p. 347)
- 고객 관리형 정책 예 (p. 347)

다음은 권한 정책의 예입니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 자격 증명 기반 정책(IAM 정책) 사용

```
"Sid": "AllowUserToPassRole",
    "Effect": "Allow",
    "Action": [ "iam:PassRole" ],
    "Resource": "*"
    }
]
```

이 정책에는 두 명령문이 있습니다:

- 첫 번째 명령문은 계정이 소유하는 임의의 캐시 클러스터에서 Amazon ElastiCache 작업 (elasticache:CreateCacheCluster, elasticache:DescribeCacheClusters, elasticache:ModifyCacheCluster 및 elasticache:RebootCacheCluster)에 대한 권한을 부여합니다. 현재 Amazon ElastiCache는 리소스 수준의 작업에 대한 권한을 지원하지 않습니다. 따라서 정책은 와일드카드 문자(*)를 Resource 값으로 지정합니다.
- 두 번째 명령문은 IAM 역할에서 IAM 작업(iam: PassRole)에 대한 권한을 부여합니다. Resource 값의 끝에 있는 와일드카드 문자(*)는 명령문이 모든 IAM 역할에서 iam: PassRole 작업에 대한 권한을 허용한다는 의미입니다. 이러한 권한을 특정 역할로 제한하려면 리소스 ARN에 있는 와일드카드 문자(*)를 특정역할 이름으로 대체합니다.

자격 증명 기반 정책에서는 권한을 가질 보안 주체를 지정하지 않으므로 이 정책은 Principal 요소를 지정하지 않습니다. 정책을 사용자에게 연결할 경우, 사용자는 묵시적인 보안 주체가 됩니다. IAM 역할에 권한 정책을 연결하면 역할의 신뢰 정책에서 식별된 보안 주체가 권한을 얻습니다.

모든 Amazon ElastiCache API 작업과 해당 작업이 적용되는 리소스를 보여주는 표는 ElastiCache API 권한: 작업, 리소스 및 조건 참조 (p. 355) 단원을 참조하십시오.

Amazon ElastiCache 콘솔 사용에 필요한 권한

권한 참조 표에 Amazon ElastiCache API 작업과 각 작업에 필요한 권한이 나열되어 있습니다. ElastiCache API 작업에 대한 자세한 내용은 ElastiCache API 권한: 작업, 리소스 및 조건 참조 (p. 355) 단원을 참조하십시오.

Amazon ElastiCache 콘솔을 사용하려면 먼저 다음 권한 정책과 같이 추가 작업에 대한 권한을 부여해야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "MinPermsForECConsole",
        "Effect": "Allow",
        "Action": [
            "elasticache:Describe*",
            "elasticache:List*",
            "ec2:DescribeAvailabilityZones",
            "ec2:DescribeVpcs",
            "ec2:DescribeAccountAttributes".
            "ec2:DescribeSecurityGroups",
            "cloudwatch:GetMetricStatistics",
            "cloudwatch:DescribeAlarms",
            "s3:ListAllMyBuckets",
            "sns:ListTopics",
            "sns:ListSubscriptions" ],
        "Resource": "*"
        }
    ]
}
```

ElastiCache 콘솔에 이러한 추가 권한이 필요한 이유는 다음과 같습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 자격 증명 기반 정책(IAM 정책) 사용

- ElastiCache 작업 권한이 있으면 콘솔에서 계정의 ElastiCache 리소스를 표시할 수 있습니다.
- 콘솔에 Amazon EC2 쿼리를 위한 ec2 작업 권한이 있어야 합니다. 그래야 가용 영역, VPC, 보안 그룹 및 계정 속성을 표시할 수 있습니다.
- cloudwatch 작업에 대한 권한이 있으면 콘솔에서 Amazon CloudWatch 지표와 경보를 검색하고 콘솔에 표시할 수 있습니다.
- sns 작업에 대한 권한이 있으면 콘솔에서 Amazon Simple Notification Service(Amazon SNS) 주제와 구독을 검색하고 콘솔에 표시할 수 있습니다.

Amazon ElastiCache에 대한 AWS 관리형(미리 정의된) 정책

AWS는 AWS에서 생성하고 관리하는 독립형 IAM 정책을 제공하여 많은 일반 사용 사례를 처리합니다. 관리형 정책은 사용자가 필요한 권한을 조사할 필요가 없도록 일반 사용 사례에 필요한 권한을 부여합니다. 자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책 단원을 참조하십시오.

계정의 사용자에게 연결할 수 있는 다음 AWS 관리형 정책은 ElastiCache에 대해 고유합니다.

- AmazonElastiCacheReadOnlyAccess Amazon ElastiCache 리소스에 대한 읽기 전용 액세스 권한을 부여합니다.
- AmazonElastiCacheFullAccess Amazon ElastiCache 리소스에 대한 모든 액세스 권한을 부여합니다.

Note

IAM 콘솔에 로그인하고 이 콘솔에서 특정 정책을 검색하면 이러한 권한 정책을 검토할 수 있습니다.

Amazon ElastiCache API 작업에 대한 권한을 허용하는 고유의 사용자 지정 IAM 정책을 생성할 수도 있습니다. 해당 권한이 필요한 IAM 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다.

고객 관리형 정책 예

기본 정책을 사용하지 않고 고객 관리형 정책을 사용할 경우 두 가지 중 하나가 가능한지 확인해야 합니다. iam:createServiceLinkedRole을 호출할 수 있는 권한이 있어야 합니다(자세한 내용은 예제 5: 사용자에게 IAM CreateServiceLinkedRole API 호출 허용 (p. 349) 단원을 참조하십시오). 또는 ElastiCache 서비스 연결 역할을 생성해야 합니다.

Amazon ElastiCache 콘솔을 사용하는 데 필요한 최소 권한과 함께 이 단원의 예제 정책을 사용할 경우 추가 권한이 부여됩니다. 예제는 AWS SDK 및 AWS CLI와도 관련이 있습니다. ElastiCache 콘솔을 사용하는 데 필요한 권한에 대한 자세한 내용은 Amazon ElastiCache 콘솔 사용에 필요한 권한 (p. 346) 단원을 참조하십시오.

IAM 사용자 및 그룹 설정 지침은 IAM 사용 설명서의 첫 번째 IAM 사용자 및 관리자 그룹 생성을 참조하십시오.

Important

프로덕션에서 IAM 정책을 사용하기 전에 항상 철저히 테스트하십시오. 나타나는 일부 ElastiCache 작업을 ElastiCache 콘솔을 사용할 때 다른 작업이 해당 작업을 지원해야 할 수도 있습니다. 예를 들 어, elasticache:CreateCacheCluster가 ElastiCache 캐시 클러스터를 생성하기 위한 권한을 부여합니다. 그러나 이 작업을 수행하기 위해 ElastiCache 콘솔에서 여러 Describe 및 List 작업을 사용하여 콘솔 목록을 채웁니다.

여

- 예제 1: 사용자에게 보안 그룹 생성 및 관리 허용 (p. 348)
- 예제 2: 사용자에게 ElastiCache 리소스에 대한 읽기 전용 액세스 허용 (p. 348)
- 예제 3: 사용자가 일반 ElastiCache 시스템 관리자 작업을 수행하도록 허용 (p. 348)

- 예제 4: 사용자가 모든 ElastiCache API 작업에 액세스하도록 허용 (p. 349)
- 예제 5: 사용자에게 IAM CreateServiceLinkedRole API 호출 허용 (p. 349)

예제 1: 사용자에게 보안 그룹 생성 및 관리 허용

다음 정책은 보안 그룹의 특정한 ElastiCache 작업에 대한 권한을 부여합니다. 일반적으로 이 유형의 권한 정책을 시스템 관리자 그룹에 연결합니다.

예제 2: 사용자에게 ElastiCache 리소스에 대한 읽기 전용 액세스 허용

다음 정책은 사용자에게 리소스를 나열하도록 허용하는 ElastiCache 작업에 대한 권한을 부여합니다. 일반적으로 이 유형의 권한 정책을 관리자 그룹에 연결합니다.

예제 3: 사용자가 일반 ElastiCache 시스템 관리자 작업을 수행하도록 허용

일반 시스템 관리자 작업에는 캐시 클러스터, 파라미터 및 파라미터 그룹 수정이 포함됩니다. 시스템 관리자가 ElastiCache 이벤트에 대한 정보를 보고 싶어할 수도 있습니다. 다음 정책은 이러한 일반 시스템 관리자작업을 위한 ElastiCache 작업을 수행할 권한을 사용자에게 부여합니다. 일반적으로 이 유형의 권한 정책을 시스템 관리자 그룹에 연결합니다.

```
"elasticache:ModifyCacheParameterGroup",
    "elasticache:DescribeCacheParameterGroups",
    "elasticache:DescribeCacheParameters",
    "elasticache:ResetCacheParameterGroup",
    "elasticache:DescribeEngineDefaultParameters"],
    "Resource":"*"
    }
]
```

예제 4: 사용자가 모든 ElastiCache API 작업에 액세스하도록 허용

다음 정책은 사용자가 모든 ElastiCache 작업을 호출할 수 있도록 허용합니다. 관리자 사용자에게만 이 유형 의 권한 정책을 부여하는 것이 좋습니다.

예제 5: 사용자에게 IAM CreateServiceLinkedRole API 호출 허용

다음 정책은 사용자가 IAM CreateServiceLinkedRole API를 호출하도록 허용합니다. 변화하기 쉬운 ElastiCache 작업을 호출하는 사용자에게 이 유형의 권한 정책을 부여하는 것이 좋습니다.

Amazon ElastiCache에 서비스 연결 역할 사용

Amazon ElastiCache은 AWS Identity and Access Management(IAM) 서비스 연결 역할을 사용합니다. 서비스 연결 역할은 AWS 서비스(예: Amazon ElastiCache, Amazon ElastiCache)에 직접 연결된 고유한 유형의 IAM 역할입니다. 서비스 연결 연할은 Amazon ElastiCache에서 사전 정의됩니다. 서비스에서 cluster를 대신하여 AWS 서비스를 자동으로 호출하기 위해 필요한 모든 권한을 포함합니다.

필요한 권한을 수동으로 추가할 필요가 없으므로 서비스 연결 역할은 Amazon ElastiCache를 더 쉽게 설정할 수 있습니다. 이 역할은 AWS 계정에 이미 속해 있으나 Amazon ElastiCache 사용 사례에 연결되어 있으

며 사전 정의된 권한을 가지고 있습니다. Amazon ElastiCache만 이 역할을 부여할 수 있고 이러한 역할만 사전 정의된 권한 정책을 사용할 수 있습니다. 먼저 역할의 관련 리소스를 삭제해야만 역할을 삭제할 수 있습니다. 이렇게 하면 Amazon ElastiCache 리소스에 대한 필수 액세스 권한을 부주의로 삭제할 수 없기 때문에 리소스가 보호됩니다.

서비스 연결 역할을 지원하는 기타 서비스에 대한 자세한 내용은 서비스 연결 역할 열에 예 가 있는 IAM으로 작업하는 AWS 서비스를 참조하십시오. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 예 링크를 선택합니다.

목차

- Amazon ElastiCache에 대한 서비스 연결 역할 권한 (p. 350)
- 서비스 연결 역할 만들기(IAM) (p. 351)
 - 서비스 연결 역할 만들기(IAM 콘솔) (p. 351)
 - 서비스 연결 역할 만들기(IAM CLI) (p. 352)
 - 서비스 연결 역할 만들기(IAM API) (p. 352)
- Amazon ElastiCache에 대한 서비스 연결 역할의 설명 편집 (p. 352)
 - 서비스 연결 역할 설명 편집(IAM 콘솔) (p. 352)
 - 서비스 연결 역할 설명 편집(IAM CLI) (p. 352)
 - 서비스 연결 역할 설명 편집(IAM API) (p. 353)
- Amazon ElastiCache에 대한 서비스 연결 역할 삭제 (p. 353)
 - 서비스 연결 역할 정리 (p. 353)
 - 서비스 연결 역할 삭제(IAM 콘솔) (p. 354)
 - 서비스 연결 역할 삭제(IAM CLI) (p. 354)
 - 서비스 연결 역할 삭제(IAM API) (p. 355)

Amazon ElastiCache에 대한 서비스 연결 역할 권한

Amazon ElastiCache에서는 AWSServiceRoleForElastiCache – This policy allows ElastiCache to manage AWS resources on your behalf as necessary for managing your cache. 서비스 연결 역할을 사용합니다.

AWSServiceRoleForElastiCache 서비스 연결 역할 권한 정책은 Amazon ElastiCache가 지정된 리소스에서 다음 작업을 완료하도록 허용합니다.

```
Permission policy:
  "Version": "2012-10-17",
  "Statement": [
      "Effect": "Allow",
      "Action": [
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:CreateNetworkInterface",
        "ec2:CreateSecurityGroup",
        "ec2:DeleteNetworkInterface".
        "ec2:DeleteSecurityGroup",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSubnets".
        "ec2:DescribeVpcs",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:RevokeSecurityGroupIngress",
      "Resource": "*"
    }
  ]
```

}

IAM 개체가 AWSServiceRoleForElastiCache 서비스 연결 역할을 생성하도록 허용하려면

IAM 개체에 대한 권한에 다음 정책 설명을 추가합니다.

IAM 개체가 AWSServiceRoleForElastiCache 서비스 연결 역할을 삭제하도록 허용하려면

IAM 개체에 대한 권한에 다음 정책 설명을 추가합니다.

또는 AWS 관리형 정책을 사용하여 Amazon ElastiCache에 대한 모든 액세스 권한을 제공할 수 있습니다.

서비스 연결 역할 만들기(IAM)

IAM 콘솔 CLI 또는 API를 사용하면 서비스 연결 역할을 만들 수 있습니다.

서비스 연결 역할 만들기(IAM 콘솔)

IAM 콘솔을 사용하여 서비스 연결 역할을 생성할 수 있습니다.

서비스 연결 역할을 만들려면(콘솔 사용)

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/iam/에서 IAM 콘솔을 엽니다.
- 2. IAM 콘솔의 탐색 창에서 [Roles]를 선택합니다. 그런 다음 [Create new role]을 선택합니다.
- AWS 서비스 연결 역할 단원을 확장한 후 새 서비스 연결 역할을 부여하려는 서비스를 선택합니다.
- 4. AWSServiceRoleForElastiCache 서비스 연결 역할 옆에 있는 선택을 선택합니다.
- 5. [Role name]에 서비스 연결 역할의 기본 이름에 추가할 접미사를 입력합니다. 이 접미사는 이 역할의 목적을 파악하는 데 도움이 됩니다. 역할 이름은 AWS 계정 내에서 고유해야 합니다. 대소문자는 구별하지 않습니다. 예를 들어, 이름이 <service-linked-role-name>_SAMPLE과 <service-linked-role-name>_sample, 두 가지로 지정된 역할을 만들 수는 없습니다. 다양한 주체가 역할을 참조할 수있기 때문에 역할이 생성된 후에는 역할 이름을 편집할 수 없습니다.
- 6. (선택 사항) [Role description]에서 새로운 서비스 연결 역할에 대한 설명을 편집합니다.
- 7. 역할을 검토한 다음 [Create role]을 선택합니다.

서비스 연결 역할 만들기(IAM CLI)

AWS Command Line Interface에서 IAM 작업을 사용하여 서비스 연결 역할을 생성할 수 있습니다. 이 역할에는 서비스가 역할을 수임하는 데 필요한 신뢰 정책 및 인라인 정책이 포함될 수 있습니다.

서비스 연결 역할을 만들려면(CLI 사용)

다음 작업을 사용합니다.

* aws iam create-service-linked-role --aws-service-name elasticache.amazonaws.com

서비스 연결 역할 만들기(IAM API)

IAM API를 사용하여 서비스 연결 역할을 생성할 수 있습니다. 이 역할에는 서비스가 역할을 수임하는 데 필요한 신뢰 정책 및 인라인 정책이 포함될 수 있습니다.

서비스 연결 역할을 만들려면(API 사용)

CreateServiceLinkedRole API 호출을 사용합니다. 요청 시 elasticache.amazonaws.com 서비스 이름을 지정합니다.

Amazon ElastiCache에 대한 서비스 연결 역할의 설명 편집

Amazon ElastiCache에서는 AWSServiceRoleForElastiCache 서비스 연결 역할을 편집하도록 허용하지 않습니다. 서비스 연결 역할을 생성한 후에는 다양한 개체가 역할을 참조할 수 있기 때문에 역할 이름을 변경할수 없습니다. 그러나 IAM를 사용하여 역할의 설명을 편집할 수 있습니다.

서비스 연결 역할 설명 편집(IAM 콘솔)

IAM 콘솔을 사용하여 서비스 연결 역할의 설명을 편집할 수 있습니다.

서비스 연결 역할의 설명을 편집하려면(콘솔 사용)

- 1. IAM 콘솔의 탐색 창에서 역할을 선택합니다.
- 2. 변경할 역할 이름을 선택합니다.
- 3. Role description(역할 설명)의 맨 오른쪽에서 편집을 선택합니다.
- 4. 상자에 새 설명을 입력하고 저장을 선택합니다.

서비스 연결 역할 설명 편집(IAM CLI)

AWS Command Line Interface의 IAM 작업을 사용하여 서비스 연결 역할의 설명을 편집할 수 있습니다.

서비스 연결 역할의 설명을 변경하려면(CLI 사용)

1. (선택 사항) 역할에 대한 현재 설명을 보려면 IAM 작업 get-role에 대해 AWS CLI를 사용하십시오.

Example

\$ aws iam get-role --role-name AWSServiceRoleForElastiCache

CLI 작업에서 역할을 참조하려면 ARN이 아니라 역할 이름을 사용해야 합니다. 예를 들어 어떤 역할의 ARN이 arn:aws:iam::123456789012:role/myrole인 경우 참조할 역할은 myrole입니다.

2. 서비스 연결 역할의 설명을 업데이트하려면 IAM 작업 update-role-description에 대해 AWS CLI를 사용하십시오.

Linux, macOS, or Unix의 경우:

```
$ aws iam update-role-description \
    --role-name AWSServiceRoleForElastiCache \
    --description "new description"
```

Windows의 경우:

```
$ aws iam update-role-description ^
    --role-name AWSServiceRoleForElastiCache ^
    --description "new description"
```

서비스 연결 역할 설명 편집(IAM API)

IAM API를 사용하여 서비스 연결 역할의 설명을 편집할 수 있습니다.

서비스 연결 역할의 설명을 변경하려면(API 사용)

1. (선택 사항) 역할의 현재 설명을 보려면 IAM API 작업 GetRole을 사용하십시오.

Example

```
https://iam.amazonaws.com/
?Action=GetRole
&RoleName=AWSServiceRoleForElastiCache
&Version=2010-05-08
&AUTHPARAMS
```

2. 역할 설명을 업데이트하려면 IAM API 작업 UpdateRoleDescription을 사용하십시오.

Example

```
https://iam.amazonaws.com/
?Action=UpdateRoleDescription
&RoleName=AWSServiceRoleForElastiCache
&Version=2010-05-08
&Description="New description"
```

Amazon ElastiCache에 대한 서비스 연결 역할 삭제

서비스 연결 역할이 필요한 기능 또는 서비스가 더 이상 필요 없는 경우에는 해당 역할을 삭제할 것을 권합니다. 따라서 적극적으로 모니터링하거나 유지하지 않는 미사용 개체가 없도록 합니다. 단, 삭제 전에 서비스 연결 역할을 정리해야 합니다.

Amazon ElastiCache에서는 서비스 연결 역할을 자동으로 삭제하지 않습니다.

서비스 연결 역할 정리

IAM을 사용하여 서비스 연결 역할을 삭제하기 전에 먼저 역할에 해당 역할과 연결된 리소스(클러스터 또는 복제 그룹)가 없는지 확인해야 합니다.

IAM 콘솔에서 서비스 연결 역할에 활성 세션이 있는지 확인하려면

1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/iam/에서 IAM 콘솔을 엽니다.

- 2. IAM 콘솔의 탐색 창에서 역할을 선택합니다. 그런 다음 AWSServiceRoleForElastiCache 역할의 이름 (확인란 아님)을 선택합니다.
- 3. 선택한 역할의 [Summary] 페이지에서 [Access Advisor] 탭을 선택합니다.
- 4. [Access Advisor] 탭에서 서비스 연결 역할의 최근 활동을 검토합니다.

AWSServiceRoleForElastiCache가 필요한 Amazon ElastiCache 리소스를 삭제하는 방법(콘솔)

- 클러스터를 삭제하려면 다음을 참조하십시오.
 - AWS Management 콘솔 사용 (p. 102)
 - AWS CLI 사용 (p. 103)
 - ElastiCache API 사용 (p. 103)
- 복제 그룹을 삭제하려면 다음을 참조하십시오.
 - 복제 그룹 (콘솔) 삭제 (p. 167)
 - 복제 그룹 삭제 (AWS CLI) (p. 167)
 - 복제 그룹 삭제 (ElastiCache API) (p. 167)

서비스 연결 역할 삭제(IAM 콘솔)

IAM 콘솔을 사용하여 서비스 연결 역할을 삭제할 수 있습니다.

서비스 연결 역할을 삭제하려면(콘솔)

- 1. AWS Management 콘솔에 로그인한 다음 https://console.aws.amazon.com/iam/에서 IAM 콘솔을 엽니다.
- 2. IAM 콘솔의 탐색 창에서 [Roles]를 선택합니다. 그런 다음 삭제할 역할의 이름이나 행이 아닌 이름 옆에 있는 확인란을 선택합니다.
- 3. 페이지 상단의 [Role actions]에서 [Delete role]을 선택합니다.
- 4. 확인 대화 상자가 나타나면 서비스 마지막 액세스 데이터를 검토합니다. 이 데이터는 선택한 각 역할이 AWS 서비스를 마지막으로 액세스한 일시를 보여 줍니다. 이를 통해 역할이 현재 활동 중인지 여부를 확인할 수 있습니다. 계속 진행하려면 [Yes, Delete]을 선택하여 삭제할 서비스 연결 역할을 제출합니다.
- 5. IAM 콘솔 알림을 보고 서비스 연결 역할 삭제 진행 상황을 모니터링합니다. IAM 서비스 연결 역할 삭제는 비동기이므로 삭제할 역할을 제출한 후에 삭제 작업이 성공하거나 실패할 수 있습니다. 작업에 실패할 경우 알림의 [View details] 또는 [View Resources]를 선택하면 삭제 실패 이유를 확인할 수 있습니다.

서비스 연결 역할 삭제(IAM CLI)

AWS Command Line Interface에서 IAM 작업을 사용하여 서비스 연결 역할을 삭제할 수 있습니다.

서비스 연결 역할(CLI)을 삭제하려면

1. 삭제할 서비스 연결 역할의 이름을 모르는 경우 다음 명령을 입력합니다. 이 명령은 계정의 역할과 해당 Amazon 리소스 이름(ARN)을 나열합니다.

\$ aws iam get-role --role-name role-name

CLI 작업에서 역할을 참조하려면 ARN이 아니라 역할 이름을 사용해야 합니다. 예를 들어 역할의 ARN 이 arn:aws:iam::123456789012:role/myrole인 경우 해당 역할을 **myrole**으로 참조합니다.

2. 서비스 연결 역할이 사용되지 않거나 연결된 리소스가 없는 경우에는 서비스 연결 역할을 삭제할 수 없으므로 삭제 요청을 제출해야 합니다. 이러한 조건이 충족되지 않으면 요청이 거부될 수 있습니다. 삭제 작업 상태를 확인하려면 응답의 deletion-task-id를 캡처해야 합니다. 다음을 입력하여 서비스 연결 역할 삭제 요청을 제출합니다.

\$ aws iam delete-service-linked-role --role-name role-name

3. 다음을 입력하여 삭제 작업의 상태를 확인합니다.

\$ aws iam get-service-linked-role-deletion-status --deletion-task-id deletion-task-id

삭제 작업은 NOT_STARTED, IN_PROGRESS, SUCCEEDED 또는 FAILED 상태일 수 있습니다. 삭제에 실패할 경우 문제를 해결할 수 있도록 실패 이유가 호출에 반환됩니다.

서비스 연결 역할 삭제(IAM API)

IAM API를 사용하여 서비스 연결 역할을 삭제할 수 있습니다.

서비스 연결 역할(API)을 삭제하려면

1. 서비스 연결 역할 삭제 요청을 제출하려면 DeleteServiceLinkedRole을 호출합니다. 요청에 역할 이름을 지정합니다.

서비스 연결 역할이 사용되지 않거나 연결된 리소스가 없는 경우에는 서비스 연결 역할을 삭제할 수 없으므로 삭제 요청을 제출해야 합니다. 이러한 조건이 충족되지 않으면 요청이 거부될 수 있습니다. 삭제작업 상태를 확인하려면 응답의 DeletionTaskId를 캡처해야 합니다.

2. 삭제 상태를 확인하려면 GetServiceLinkedRoleDeletionStatus을 호출합니다. 요청에 DeletionTaskId를 지정합니다.

삭제 작업은 NOT_STARTED, IN_PROGRESS, SUCCEEDED 또는 FAILED 상태일 수 있습니다. 삭제에 실패할 경우 문제를 해결할 수 있도록 실패 이유가 호출에 반환됩니다.

ElastiCache API 권한: 작업, 리소스 및 조건 참조

IAM 자격 증명에 연결할 액세스 제어 (p. 342) 및 쓰기 권한 정책(자격 증명 기반 정책)을 설정할 때 다음 표를 참조로 사용합니다. 표에는 각 Amazon ElastiCache API 작업과 이 작업을 수행할 수 있는 권한을 부여할수 있는 작업이 나와 있습니다. 정책의 Action 필드에서 작업을 지정하고, 정책의 Resource 필드에서 리소스 값으로 와일드카드 문자 (*)를 지정합니다.

ElastiCache 정책에서 AWS 차원 조건 키를 사용하여 조건을 표시할 수 있습니다. AWS 차원 키의 전체 목록은 IAM 사용 설명서의 사용 가능한 조건 키 단원을 참조하십시오.

Note

작업을 지정하려면 elasticache: 접두사 다음에 API 작업 이름을 사용합니다(예: elasticache: DescribeCacheClusters). 모든 ElastiCache 작업에 와일드카드 문자(*)를 리소스로 지정하십시오.

Amazon ElastiCache API 및 작업에 대한 필수 권한

AddTagsToResource

작업: elasticache: AddTagsToResource

리소스: *

AuthorizeCacheSecurityGroupIngress

작업: elasticache: Authorize Cache Security Group Ingress

리소스: *

```
CopySnapshot
   작업: elasticache: CopySnapshot
   리소스: *
CreateCacheCluster
   작업: elasticache: CreateCacheCluster
   s3:GetObject
      Note
      SnapshotArns 파라미터를 사용하는 경우 SnapshotArns 목록의 각 멤버에 s3 ARN이 리소
      스인 자체 s3:GetObject 권한이 필요합니다.
   리소스: *
   arn:aws:s3:::my_bucket/snapshot1.rdb
   여기서 my_bucket/snapshot1은 캐시 클러스터를 생성할 S3 버킷 및 스냅샷입니다.
CreateCacheParameterGroup
   작업: elasticache:CreateCacheParameterGroup
   리소스: *
CreateCacheSecurityGroup
   작업: elasticache:CreateCacheSecurityGroup
   리소스: *
CreateCacheSubnetGroup
   작업: elasticache: CreateCacheSubnetGroup
   리소스: *
CreateReplicationGroup
   작업: elasticache: CreateReplicationGroup
   s3:GetObject
      Note
      SnapshotArns 파라미터를 사용하는 경우 SnapshotArns 목록의 각 멤버에 s3 ARN이 리소
      스인 자체 s3:GetObject 권한이 필요합니다.
   리소스: *
   arn:aws:s3:::my_bucket/snapshot1.rdb
   여기서 my bucket/snapshot1은 캐시 클러스터를 생성할 S3 버킷 및 스냅샷입니다.
CreateSnapshot
   작업: elasticache:CreateSnapshot
   리소스: *
DeleteCacheCluster
   작업: elasticache: DeleteCacheCluster
```

리소스: * DeleteCacheParameterGroup 작업: elasticache:DeleteCacheParameterGroup 리소스: * DeleteCacheSecurityGroup 작업: elasticache: DeleteCacheSecurityGroup 리소스: * DeleteCacheSubnetGroup 작업: elasticache:DeleteCacheSubnetGroup 리소스: * DeleteReplicationGroup 작업: elasticache: DeleteReplicationGroup 리소스: * DeleteSnapshot 작업: elasticache: DeleteSnapshot 리소스: * DescribeCacheClusters 작업: elasticache:DescribeCacheClusters 리소스: * DescribeCacheEngineVersions 작업: elasticache: DescribeCacheEngineVersions 리소스: * DescribeCacheParameterGroups 작업: elasticache:DescribeCacheParameterGroups 리소스: * DescribeCacheParameters 작업: elasticache: DescribeCacheParameters 리소스: * DescribeCacheSecurityGroups 작업: elasticache:DescribeCacheSecurityGroups 리소스: * DescribeCacheSubnetGroups 작업: elasticache:DescribeCacheSubnetGroups 리소스: *

DescribeEngineDefaultParameters 작업: elasticache: DescribeEngineDefaultParameters 리소스: * DescribeEvents 작업: elasticache: DescribeEvents 리소스: * DescribeReplicationGroups 작업: elasticache: DescribeReplicationGroups 리소스: * DescribeReservedCacheNodes 작업: elasticache:DescribeReservedCacheNodes 리소스 * DescribeReservedCacheNodesOfferings 작업: elasticache:DescribeReservedCacheNodesOfferings 리소스: * **DescribeSnapshots** 작업: elasticache: DescribeSnapshots 리소스: * ListTagsForResource 작업: elasticache:ListTagsForResource 리소스: * ModifyCacheCluster 작업: elasticache: ModifyCacheCluster 리소스: * ModifyCacheParameterGroup 작업: elasticache: ModifyCacheParameterGroup 리소스: * ModifyCacheSubnetGroup 작업: elasticache: ModifyCacheSubnetGroup 리소스: * ModifyReplicationGroup 작업: elasticache: ModifyReplicationGroup 리소스: * PurchaseReservedCacheNodesOffering

작업: elasticache: PurchaseReservedCacheNodesOffering

리소스: *

RebootCacheCluster

작업: elasticache: RebootCacheCluster

리소스: *

RemoveTagsFromResource

작업: elasticache: RemoveTagsFromResource

리소스: *

ResetCacheParameterGroup

작업: elasticache: ResetCacheParameterGroup

리소스: *

RevokeCacheSecurityGroupIngress

작업: elasticache:RevokeCacheSecurityGroupIngress

리소스: *

TestFailover

작업: elasticache: TestFailover

리소스: *

Elasticache에서 로깅 및 모니터링

엔터프라이즈 캐싱 솔루션을 관리하려면 클러스터가 어떻게 수행하고 클러스터에서 사용하는 리소스를 알 아야 합니다. 또한 생성되는 이벤트와 배포 비용을 파악하는 것도 중요합니다.

Amazon CloudWatch는 캐시 성능을 모니터링하기 위한 측정치를 제공합니다. 또한, 비용 할당 태그는 비용 을 모니터링하고 관리하는 데 도움이 됩니다.

주제

- CloudWatch 측정치를 통한 모니터링 사용 (p. 359)
- ElastiCache 이벤트 모니터링 (p. 367)
- 비용 할당 태그를 사용한 비용 모니터링 (p. 375)
- 예약 노드를 통해 비용 관리 (p. 382)

CloudWatch 측정치를 통한 모니터링 사용

ElastiCache는 클러스터를 모니터링할 수 있는 측정치를 제공합니다. CloudWatch를 통해 이러한 측정치에 액세스할 수 있습니다. CloudWatch에 대한 자세한 내용은 CloudWatch 설명서를 참조하십시오.

ElastiCache는 호스트 수준 측정치(예: CPU 사용) 및 캐시 엔진 소프트웨어별 측정치(예: 캐시가 얻은 것과 잃은 것) 모두를 제공합니다. 이러한 메트릭은 60초 간격으로 각 캐시 노드에 대해 측정되어 게시됩니다.

Important

특정 키 측정치에 CloudWatch 경보를 설정하면 캐시 클러스터의 성능이 나빠지기 시작하면 알림을 받을 수 있습니다. 자세한 내용은 이 가이드의 어떤 메트릭을 모니터링해야 합니까? (p. 364)을 참조하십시오.

주제

- 호스트 수준 지표 (p. 360)
- Redis 지표 (p. 360)
- 어떤 메트릭을 모니터링해야 합니까? (p. 364)
- 메트릭 통계 및 기간 선택 (p. 365)
- Monitoring CloudWatch Cluster and Node Metrics (p. 365)

호스트 수준 지표

AWS/ElastiCache 네임스페이스에는 다음과 같이 개별 캐시 노드에 대한 호스트 수준 지표가 포함되어 있습니다.

참고 항목

• Redis 지표 (p. 360)

ス 班	설명	단위
CPUUtilization	전체 호스트에 대한 CPU 사용률 비율. Redis는 단일 스레드이므로 vCPU가 4개 이상인 노드에 대해 EngineCPUUtilization 지표를 모니터링하는 것이 좋습니다.	%
FreeableMemory	호스트에서 사용 가능한 메모리의 양. 이 값은 OS가 여유 공간으로 보고한 RAM, 버퍼, 캐시에서 나왔습 니다.	바이트
NetworkBytesIn	호스트가 네트워크에서 읽어온 바이트 수	바이트
NetworkBytesOut	인스턴스가 모든 네트워크 인터페이스에서 보낸 바 이트 수입니다.	바이트
NetworkPacketsIn	인스턴스가 모든 네트워크 인터페이스에서 받은 패 킷 수입니다. 이 지표는 단일 인스턴스에서 수신 트 래픽의 볼륨을 패킷 수 기준으로 식별합니다.	개수
NetworkPacketsOut	인스턴스가 모든 네트워크 인터페이스에서 보낸 패 킷 수입니다. 이 지표는 단일 인스턴스에서 발신 트 래픽의 볼륨을 패킷 수 기준으로 식별합니다.	개수
SwapUsage	호스트에서 사용되는 스왑의 양.	바이트

Redis 지표

AWS/ElastiCache 네임스페이스에는 다음 Redis 지표가 포함되어 있습니다.

ReplicationLag 및 EngineCPUUtilization를 제외하고 이 지표는 Redis info 명령에서 파생됩니다. 각 지표는 캐시 노드 수준에서 계산됩니다.

Redis info 명령에 대한 전체 설명서는 http://redis.io/commands/info를 참조하십시오.

참고 항목

• 호스트 수준 지표 (p. 360)

지표	설명	단위
ActiveDefragHits	활성 조각 모음 프로세스에서 수행된 분당 값 재할당 수입니다. 이 수는 Redis INFO의 active_defrag_hits 통계 자료에서 나왔습니 다.	숫자
BytesUsedForCache	데이터 세트, 버퍼 등을 포함하여 모든 목적을 위해 Redis에서 할당한 전체 바이트 수. 이 수는 Redis INFO의 used_memory 통계 자료에서 나왔습니다.	바이트
CacheHits	주 딕셔너리의 성공한 읽기 전용 키 조회수. 이 수는 Redis INFO의 keyspace_hits 통계 자료에서 나 왔습니다.	개수
CacheMisses	주 딕셔너리의 성공하지 못한 읽기 전용 키 조회 수. 이 수는 Redis INFO의 keyspace_misses에서 나 왔습니다.	개수
CurrConnections	읽기 전용 복제본의 연결을 제외한 클라이언트 연결 횟수 ElastiCache에서는 연결 4회 중 2회를 이용하 여 각 사례의 클러스터를 모니터링합니다. 이 수는 Redis INFO의 connected_clients 통계 자료에 서 나왔습니다.	개수
EngineCPUUtilization	Redis 엔진 스레드의 CPU 사용률을 제공합니다. Redis는 단일 스레드이므로 이 지표를 사용하여 Redis 프로세스 자체의 로드를 분석할 수 있습니다. EngineCPUUtilization 지표는 Redis 프로세스에 대한 보다 정확한 정보를 제공하며, 다른 운영 체제 및 관리 프로세스를 포함하여 전체적인 서버 인스턴스의 CPU 사용률을 표시하는 CPUUtilization 지표와 함께 사용할 수 있습니다. 4vCPU 이상을 포함하는 대규모 노드 유형에는 EngineCPUUtilization 지표를 사용하여 조정임계값을 모니터링하고 설정하십시오. 2vCPU 이하의 작은 노드 유형에는 CPUUtilization 지표를 사용하십시오.	%
Evictions	maxmemory 제한으로 인해 제거된 키 수 이 수는 Redis INFO의 evicted_keys 통계 자료에서 나왔 습니다.	개수
MasterLinkHealthStatus	이 상태에는 0 또는 1의 두 가지 값이 있습니다. 값 0 은 Elasticache 기본 노드의 데이터가 Redis on EC2 와 동기화되지 않았음을 나타냅니다. 값 1은 데이터 가 동기화되었음을 나타냅니다. 마이그레이션을 완 료하려면 CompleteMigration API를 사용합니다.	부울
NewConnections	이 기간에 서버에서 허용된 총 연결 수 이 수는 Redis INFO의 total_connections_received 통계 자료에서 나왔습니다.	개수
Reclaimed	키 만료 이벤트 총 수 이 수는 Redis INFO의 expired_keys 통계 자료에서 나왔습니다.	개수
ReplicationBytes	복제된 구성 노드의 경우 ReplicationBytes가 기본 노드에서 모든 복제본에 전송하는 바이트 수를 보고합니다. 이 지표는 복제 그룹에 대한 쓰	바이트

지표	설명	단위
	기 부하를 나타냅니다. 이 수는 Redis INFO의 master_repl_offset 통계 자료에서 나왔습니 다.	
ReplicationLag	이 지표는 읽기 복제본으로 실행되는 노드에 한해 적용됩니다. 기본 노드에서 변경 내용을 적용할 때 복제본에서 경과된 시간(초)을 나타냅니다.	Redis 엔진 버 전 5.0.6의 경우 밀리초. 지원되 는 다른 모든 엔 진 버전의 경우 에는 초
SaveInProgress	이 이진 지표는 백그라운드 저장(forked 또는 forkless)가 진행 중일 때마다 1을, 그렇지 않으면 0을 반환합니다. 백그라운드 저장 프로세스는 일반적으로 스냅샷 작업과 동기화 작업에 사용됩니다. 하지만 이 두 가지 작업은 성능 저하의 원인이 되기도 합니다. 이때는 SaveInProgress 지표를 사용하면 백그라운드 저장 프로세스로 인한 성능 저하여부를 진단할 수 있습니다. 이 수는 Redis INFO의 rdb_bgsave_in_progress 통계 자료에서 나왔습니다.	개수

EngineCPUUtilization 가용성

다음 표에 지정된 날짜 및 시간 이후에 생성 또는 교체된 리전의 노드에는 EngineCPUUtilization 지표가 포함되어 있습니다.

리전	리전 이름	EngineCPUUtilization 가용성	
us-east-2	미국 동부(오하이오)	2017년 2월 16일	17:21(UTC)
us-east-1	미국 동부(버지니아 북부)	2017년 2월 8일	21:20(UTC)
us-west-1	미국 서부(캘리포니아 북부 지역)	2017년 2월 14일	22:23(UTC)
us-west-2	미국 서부(오레곤)	2017년 2월 20일	19:20(UTC)
ap-northeast-1	아시아 태평양(도쿄)	2017년 2월 14일	19:58(UTC)
ap-northeast-2	아시아 태평양(서울)	모든 노드에서 사용 가능.	
ap-northeast-3	아시아 태평양(오사카-로컬)	모든 노드에서 사용 가능.	
ap-south-1	아시아 태평양(뭄바이)	2017년 2월 7일	02:51(UTC)
ap-southeast-1	아시아 태평양(싱가포르)	2017년 2월 13일	23:40(UTC)
ap-southeast-2	아시아 태평양(시드니)	2017년 2월 14일	03:33(UTC)
ca-central-1	캐나다(중부)	모든 노드에서 사용 가능.	
cn-north-1	중국(베이징)	2017년 2월 16일	22:39(UTC)
cn-northwest-2	중국(닝샤)	모든 노드에서 사용 가능.	
eu-central-1	유럽(프랑크푸르트)	2017년 2월 15일	00:46(UTC)

리전	리전 이름	EngineCPUUtilization 가용성	
eu-west-1	유럽(아일랜드)	2017년 2월 7일	21:30(UTC)
eu-west-2	유럽(런던)	2017년 2월 16일	18:58(UTC)
eu-west-3	EU(파리)	모든 노드에서 사용 가능.	
sa-east-1	남아메리카(상파울루)	2017년 2월 7일 04:35(UT	
us-gov-west-1	AWS GovCloud (US-West)	2017년 2월 16일	20:11(UTC)

다음은 info commandstats에서 파생된 몇 가지 유형의 명령 모음입니다. 사용 가능한 전체 명령의 목록은 redis 명령을 참조하십시오.

ス 班	설명	단위
CurrItems	캐시 항목 수. 이 지표는 전체 키 스페이스의 모든 키를 합산하여 Redis keyspace 통계에서 파생됩니다.	개수
GetTypeCmds	read-only 유형 명령의 총 건수. 이 수는 read-only 모든 유형 명령(get, hget, scard, Irange 등)을 합산하여 commandstats 통계에서 나왔습니다.	개수
HashBasedCmds	해시 기반 명령 총 수. 이 지표는 1개 이상의 해시 (hget, hkeys, hvals, hdel 등)를 기반으로 실행되는 모든 명령을 합산하여 Redis commandstats 통계 에서 파생됩니다.	개수
HyperLogLogBasedCmds	HyperLogLog 기반 명령 총 건수. 모든 pf 유형의 명령(pfadd, pfcount, pfmerge 등)을 합산하여 Redis commandstats 통계에서 나왔습니다.	개수
KeyBasedCmds	키 기반 명령 총 수. 이 수는 여러 데이터 구조(del, expire, rename 등) 상에서 1개 이상의 키에서 작동 하는 모든 명령을 합산하여 Redis commandstats 통계에서 나왔습니다.	개수
ListBasedCmds	목록 기반 명령 총 수. 이 지표는 1개 이상의 목록 (lindex, Irange, Ipush, Itrim 등)을 기반으로 실행되는 모든 명령을 합산하여 Redis commandstats 통계 에서 파생됩니다.	개수
SetBasedCmds	집합 기반 명령 총 수. 이 지표는 1개 이상의 집합 (scard, sdiff, sadd, sunion 등)을 기반으로 실행되 는 모든 명령을 합산하여 Redis commandstats 통 계에서 파생됩니다.	개수
SetTypeCmds	write 유형의 총 명령 건수. 이 수는 데이터(set, hset, sadd, lpop 등)에서 작동하는 모든 mutative 유형의 명령을 합산하여 Redis commandstats 통계에서 나왔습니다.	개수
SortedSetBasedCmds	정렬된 집합 기반 명령 총 수. 이 지표는 1개 이상 의 정렬된 집합(zcount, zrange, zrank, zadd 등)을 기반으로 실행되는 모든 명령을 합산하여 Redis commandstats 통계에서 파생됩니다.	개수

ス 班	설명	단위
StringBasedCmds	문자열 기반 명령 총 수. 이 지표는 1개 이상의 문자 열(strlen, setex, setrange 등)을 기반으로 실행되는 모든 명령을 합산하여 Redis commandstats 통계 에서 파생됩니다.	개수
StreamBasedCmds	총 스트림 기반 명령 수. 이 지표는 1개 이상의 스 트림 데이터 형식(xrange, xlen, xadd, xdel 등)을 기반으로 실행되는 모든 명령을 합산하여 Redis commandstats 통계에서 파생됩니다.	개수

어떤 메트릭을 모니터링해야 합니까?

다음 CloudWatch 측정치는 ElastiCache 성능에 대한 좋은 정보를 제공합니다. 대부분의 경우 이러한 측정치에 대해 성능 문제가 생기기 전에 수정 조치를 취할 수 있도록 CloudWatch 경보를 설정하는 것이 좋습니다.

모니터링할 측정치

- CPUUtilization (p. 364)
- EngineCPUUtilization (p. 364)
- SwapUsage (p. 365)
- Evictions (p. 365)
- CurrConnections (p. 365)

CPUUtilization

이는 백분율(%)로 보고된 호스트 수준 지표입니다. 자세한 내용은 호스트 수준 지표 (p. 360) 단원을 참조하십시오.

2vCPU 이하의 작은 노드 유형에는 CPUUtilization 지표를 사용하여 워크로드를 모니터링하십시오.

일반적으로, 사용 가능한 CPU의 90%로 임계값을 설정하는 것이 좋습니다. Redis는 단일 스레드이기 때문에 실제 임계값은 노드 총 용량의 일부로 계산해야 합니다. 2개의 코어가 있는 노드 유형을 사용하는 경우를 예로 들어보겠습니다. 이 경우 CPUUtilization의 임계값은 90/2 또는 45%입니다. 노드 유형에 있는 코어(vCPU)의 수를 확인하려면 Amazon ElastiCache요금을 참조하십시오.

사용 중인 캐시 노드에 있는 코어 개수에 따라 임계값을 결정해야 합니다. 이 임계값을 초과하고, 주된 워크로드가 읽기 요청에서 비롯되는 경우에는 읽기 전용 복제본을 추가하여 캐시 클러스터를 스케일 아웃합니다. 주된 워크로드가 쓰기 요청에서 비롯되는 경우에는 클러스터 구성에 따라 다음을 권장합니다.

- Redis(클러스터 모드 비활성화됨) 클러스터: 더 큰 캐시 인스턴스 유형을 사용하여 확장합니다.
- Redis(클러스터 모드 활성화됨) 클러스터: 샤드를 추가하여 쓰기 워크로드를 기본 노드 전체에 더 많이 배 포합니다.

Tip

호스트 수준 지표 CPUUtilization을 사용하는 대신, Redis 사용자는 Redis 엔진 코어의 대한 사용률을 보고하는 Redis 지표 EngineCPUUtilization을 사용할 수 있습니다. 노드에서 이 측정치를 사용할 수 있는지 확인하고 자세한 내용을 보려면 Redis 지표를 참조하십시오.

EngineCPUUtilization

4vCPU 이상의 대규모 노드 유형에는 Redis 엔진 코어의 대한 사용률을 보고하는 EngineCPUUtilization 지표를 사용할 수 있습니다. 노드에서 이 측정치를 사용할 수 있는지 확인하고 자세한 내용을 보려면 Redis 지표를 참조하십시오.

SwapUsage

이는 바이트로 보고된 호스트 수준 측정치입니다. 자세한 내용은 호스트 수준 지표 (p. 360) 단원을 참조하십시오.

이 메트릭은 50MB를 초과하지 않아야 합니다. 초과한다면 다음 주제를 참조하십시오.

- 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)
- 예약된 메모리 관리 (p. 405)

Evictions

이것은 캐시 엔진 메트릭입니다. 애플리케이션 요구 사항에 따라 이 메트릭에 대한 경보 임계값을 결정하는 것이 좋습니다.

CurrConnections

이것은 캐시 엔진 메트릭입니다. 애플리케이션 요구 사항에 따라 이 메트릭에 대한 경보 임계값을 결정하는 것이 좋습니다.

CurrConnections 개수가 증가하는 것은 애플리케이션에 문제가 있음을 나타내며 이 문제를 해결하려면 애플리케이션 동작을 확인해야 합니다.

메트릭 통계 및 기간 선택

CloudWatch에서 각 측정치의 통계 및 기간을 선택하도록 허용하며 모든 조합이 유용한 것은 아닙니다. 예를 들어, CPUUtilization에 대한 Average, Minimum 및 Maximum 통계는 유용하지만 Sum 통계는 유용하지 않습 니다.

모든 ElastiCache 샘플은 각 개별 캐시 노드에 대해 60초 동안 게시됩니다. 60초 기간 동안 캐시 노드 메트릭은 단일 샘플만 포함할 수 있습니다.

캐시 노드 메트릭을 검색하는 방법에 대한 자세한 내용은 Monitoring CloudWatch Cluster and Node Metrics (p. 365)를 참조하십시오.

Monitoring CloudWatch Cluster and Node Metrics

ElastiCache와 CloudWatch가 서로 통합되어 있어서 다양한 지표를 수집할 수 있습니다. CloudWatch를 사용 하여 이러한 지표를 모니터링할 수 있습니다.

Note

다음 예제를 실행하려면 CloudWatch 명령줄 도구가 필요합니다. CloudWatch에 대해 자세한 내용을 알아보고 개발자 도구를 다운로드하려면 CloudWatch제품 페이지를 참조하십시오.

다음 절차에서는 CloudWatch를 사용하여 지난 시간 캐시 클러스터의 스토리지 공간 통계를 수집하는 방법을 보여줍니다.

Note

아래 예제에 나온 StartTime 및 EndTime 값은 설명을 돕기 위해 지정되었습니다. 따라서 캐시 노드의 올바른 시작 및 종료 시간 값으로 대체해야 합니다.

ElastiCache 제한에 대한 정보는 ElastiCache의 AWS 서비스 제한을 참조하십시오.

Monitoring CloudWatch Cluster and Node Metrics (콘솔)

캐시 클러스터의 CPU 사용률 통계를 수집하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 지표를 확인할 캐시 노드를 선택합니다.

Note

20개보다 많은 노드를 선택하면 콘솔에 지표가 표시되지 않습니다.

a. AWS Management Console의 [Cache Clusters] 페이지에서 하나 이상의 캐시 클러스터 이름을 클릭합니다.

캐시 클러스터의 세부 정보 페이지가 나타납니다.

- b. 창 맨 위의 [Nodes] 탭을 클릭합니다.
- c. 세부 정보 창의 [Nodes] 탭에서 지표를 확인할 캐시 노드를 선택합니다.

사용 가능한 CloudWatch 지표 목록이 콘솔 창 하단에 나타납니다.

d. [CPU Utilization] 지표를 클릭합니다.

선택한 지표가 표시된 CloudWatch 콘솔이 열립니다. [Statistic] 및 [Period] 드롭다운 목록 상자와 [Time Range] 탭을 사용하여 표시되는 지표를 변경할 수 있습니다.

Monitoring CloudWatch Cluster and Node Metrics CloudWatch CLI 사용

캐시 클러스터의 CPU 사용률 통계를 수집하려면

Linux, macOS, or Unix의 경우:

```
aws cloudwatch get-metric-statistics \
    --namespace AWS/ElastiCache \
    --metric-name CPUUtilization \
    --dimensions="'[{"Name":"CacheClusterId", "Value":"mycachecluster"},
{"Name":"CacheNodeId", "Value":"0002"}]' \
    --statistics=Average \
    --start-time 2018-07-05T00:00:00 \
    --end-time 2018-07-06T00:00:00 \
    --period=3600
```

Windows의 경우:

Monitoring CloudWatch Cluster and Node Metrics CloudWatch API 사용

캐시 클러스터의 CPU 사용률 통계를 수집하려면

- CloudWatch API GetMetricStatistics를 다음 파라미터와 함께 호출합니다. 시작 및 종료 시간은 예제와 같이 표시되며 적절한 시작 및 종료 시간으로 대체해야 합니다.
 - Statistics.member.1=Average

- Namespace=AWS/ElastiCache
- StartTime=2013-07-05T00:00:00
- EndTime=2013-07-06T00:00:00
- Period=60
- MeasureName=CPUUtilization
- Dimensions=CacheClusterId=mycachecluster, CacheNodeId=0002

Example

http://monitoring.amazonaws.com/
 ?Action=GetMetricStatistics
 &SignatureVersion=4
 &Version=2014-12-01
 &StartTime=2018-07-05T00:00:00
 &EndTime=2018-07-06T23:59:00
 &Period=3600
 &Statistics.member.1=Average
 &Dimensions.member.1="CacheClusterId=mycachecluster"
 &Dimensions.member.2="CacheNodeId=0002"
 &Namespace=AWS/ElastiCache
 &MeasureName=CPUUtilization
 &Timestamp=2018-07-07T17%3A48%3A21.746Z
 &AWSAccessKeyId=<AWS Access Key ID>
 &Signature=<Signature>

ElastiCache 이벤트 모니터링

클러스터에서 중요 이벤트가 발생하면 ElastiCache는 특정 Amazon SNS 주제에 알림을 전송합니다. 이러한 예에는 노드 추가 실패, 노드 추가 성공, 보안 그룹 수정 등이 있습니다. 주요 이벤트를 모니터링하면 클러스 터의 현재 상태를 파악할 수 있으며, 이벤트에 따라 교정 작업을 수행할 수도 있습니다.

주제

- ElastiCache Amazon SNS 알림 관리 (p. 367)
- ElastiCache 이벤트 보기 (p. 370)
- 이벤트 알림 및 Amazon SNS (p. 372)

ElastiCache Amazon SNS 알림 관리

Amazon Simple Notification Service(Amazon SNS)를 사용하여 중요한 클러스터 이벤트에 대해 알림을 보내 도록 ElastiCache를 구성할 수 있습니다. 이러한 예에서는 Amazon SNS 항목의 ARN(Amazon 리소스 이름) 으로 클러스터를 구성하여 알림을 받습니다.

Note

이 항목에서는 Amazon SNS에 가입했으며 Amazon SNS 주제를 설정 및 구독했다고 가정합니다. 이렇게 하는 방법에 대한 정보는 Amazon Simple Notification Service 개발자 안내서 단원을 참조하십시오.

Amazon SNS 주제 추가

다음 단원은 AWS 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 Amazon SNS 주제를 추가하는 방법을 보여줍니다.

Amazon SNS 주제 (콘솔) 추가

다음 절차는 클러스터에 대해 Amazon SNS 주제를 추가하는 방법을 보여줍니다. 복제 그룹에 대해 Amazon SNS 주제를 추가하려면 2단계에서 클러스터를 선택하는 대신 복제 그룹을 선택한 다음 동일한 남은 단계를 따르십시오.

Note

이 프로세스는 Amazon SNS 주제를 수정하는 데에도 사용할 수 있습니다.

(콘솔) 클러스터에 대해 Amazon SNS 주제를 추가 또는 수정하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 클러스터에서 Amazon SNS 주제 ARN을 추가 또는 수정할 클러스터를 선택합니다.
- 3. [Modify]를 선택합니다.
- 4. SNS 알림 주제 아래의 클러스터 수정에서 추가할 SNS 주제를 선택하거나 수동 ARN 입력을 선택하고 Amazon SNS 주제의 ARN을 입력합니다.
- 5. [Modify]를 선택합니다.

Amazon SNS 주제 (AWS CLI) 추가

클러스터에 대해 Amazon SNS 주제를 추가 또는 수정하려면 AWS CLI 명령 modify-cache-cluster을 사용합니다.

다음 코드 예제는 Amazon SNS 주제를 my-cluster에 추가합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-cluster \
    --cache-cluster-id my-cluster \
    -notification-topic-arn arn:aws:sns:us-west-2:565419523791:ElastiCacheNotifications
```

Windows의 경우:

```
aws elasticache modify-cache-cluster ^
--cache-cluster-id my-cluster ^
--notification-topic-arn arn:aws:sns:us-west-2:565419523791:ElastiCacheNotifications
```

자세한 내용은 modify-cache-cluster를 참조하십시오.

Amazon SNS 주제 (ElastiCache API) 추가

클러스터에 대해 Amazon SNS 주제를 추가 또는 수정하려면 다음 파라미터와 함께 ModifyCacheCluster 작업을 호출합니다.

- CacheClusterId=my-cluster
- TopicArn=arn%3Aaws%3Asns%3Auswest-2%3A565419523791%3AElastiCacheNotifications

Example

```
https://elasticache.amazon.com/
?Action=ModifyCacheCluster
&ApplyImmediately=false
&CacheClusterId=my-cluster
```

```
&NotificationTopicArn=arn%3Aaws%3Asns%3Aus-
west-2%3A565419523791%3AElastiCacheNotifications
&Version=2014-12-01
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
&X-Amz-Expires=20141201T220302Z
&X-Amz-Signature=<signature>
```

자세한 내용은 ModifyCacheCluster를 참조하십시오.

Amazon SNS 알림 활성화 및 비활성화

클러스터에 대해 알림을 켜거나 끌 수 있습니다. 다음 절차는 Amazon SNS 알림을 비활성화하는 방법을 보여줍니다.

Amazon SNS 알림 (콘솔) 활성화 및 비활성화

AWS Management 콘솔을 사용하여 Amazon SNS 알림을 비활성화하려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. Redis를 실행 중인 클러스터의 목록을 보려면 탐색 창에서 [Redis]를 선택합니다.
- 3. 알림을 수정할 클러스터의 이름 왼쪽에 있는 확인란을 선택합니다.
- 4. [Modify]를 선택합니다.
- 5. [Topic for SNS Notification] 아래의 [Modify Cluster]에서 [Disable Notifications]를 선택합니다.
- 6. [Modify]를 선택합니다.

Amazon SNS 알림 (AWS CLI) 활성화 및 비활성화

Amazon SNS 알림을 비활성화하려면 다음 파라미터와 함께 modify-cache-cluster 명령을 사용합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache modify-cache-cluster \
    --cache-cluster-id my-cluster \
    --notification-topic-status inactive
```

Windows의 경우:

```
aws elasticache modify-cache-cluster ^
--cache-cluster-id my-cluster ^
--notification-topic-status inactive
```

Amazon SNS 알림(ElastiCache API) 활성화 및 비활성화

Amazon SNS 알림을 비활성화하려면 다음 파라미터와 함께 ModifyCacheCluster 작업을 호출합니다.

- CacheClusterId=my-cluster
- NotificationTopicStatus=inactive
- 이 호출은 다음과 비슷한 출력을 반환합니다.

Example

https://elasticache.us-west-2.amazonaws.com/
?Action=ModifyCacheCluster
&ApplyImmediately=false
&CacheClusterId=my-cluster
&NotificationTopicStatus=inactive
&Version=2014-12-01
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&XX-Amz-Expires=20141201T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Credential=<credential>

ElastiCache 이벤트 보기

ElastiCache는 클러스터 인스턴스, 보안 그룹 및 파라미터 그룹과 관련된 이벤트를 기록합니다. 여기에는 이벤트 날짜 및 시간, 이벤트의 원본 이름 및 원본 유형, 이벤트 설명 등의 정보가 포함됩니다. ElastiCache 콘솔, AWS CLI describe-events 명령 또는 ElastiCache API 작업 DescribeEvents을 사용하여 로그에서 이벤트를 쉽게 검색할 수 있습니다.

다음 절차는 지난 24시간(1440분) 동안의 모든 ElastiCache 이벤트를 보는 방법을 보여줍니다.

ElastiCache 이벤트 (콘솔) 보기

다음 절차는 ElastiCache 콘솔을 사용하여 이벤트를 표시합니다.

ElastiCache 콘솔을 사용하여 이벤트를 보려면

- AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 사용 가능한 모든 이벤트의 목록을 보려면 탐색 창에서 이벤트를 선택합니다.

이벤트 화면에서 목록의 각 행은 이벤트 하나를 나타내며 이벤트 소스, 이벤트 유형(cache-cluster, cache-parameter-group, cache-security-group 또는 cache-subnet-group), 이벤트의 GMT 시간 및 이벤트 설명을 표시합니다.

[Filter]를 사용하여 이벤트 목록에서 모든 이벤트를 볼지 특정 유형의 이벤트만 볼지를 지정할 수 있습니다.

ElastiCache 이벤트 (AWS CLI) 보기

ElastiCache를 사용하여 AWS CLI 이벤트의 목록을 생성하려면 describe-events 명령을 사용합니다. 선택적 파라미터를 사용하여 나열된 이벤트의 유형, 나열된 이벤트의 기간, 나열할 이벤트의 최대 수 등을 제어할 수 있습니다.

다음 코드는 최대 40개의 캐시 클러스터 이벤트를 나열합니다.

aws elasticache describe-events --source-type cache-cluster --max-items 40

다음 코드는 지난 24시간(1440분) 동안의 모든 이벤트를 나열합니다.

aws elasticache describe-events --duration 1440

describe-events 명령의 출력은 다음과 같습니다.

사용 가능한 파라미터 및 허용된 파라미터 값과 같은 자세한 내용은 describe-events를 참조하십시오.

ElastiCache 이벤트 (ElastiCache API) 보기

ElastiCache API를 사용하여 ElastiCache 이벤트의 목록을 생성하려면 DescribeEvents 작업을 사용합니다. 선택적 파라미터를 사용하여 나열된 이벤트의 유형, 나열된 이벤트의 기간, 나열할 이벤트의 최대 수 등을 제어할 수 있습니다.

다음 코드는 40개의 최신 cache-cluster 이벤트를 나열합니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeEvents
&MaxRecords=40
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&SourceType=cache-cluster
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

다음 코드는 지난 24시간(1440분) 동안의 cache-cluster 이벤트를 나열합니다.

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=DescribeEvents
    &Duration=1440
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &SourceType=cache-cluster
    &Timestamp=20150202T192317Z
    &Version=2015-02-02
    &X-Amz-Credential=<credential>
```

위 작업을 통해 다음과 비슷한 출력이 생성되어야 합니다.

사용 가능한 파라미터 및 허용된 파라미터 값과 같은 자세한 내용은 DescribeEvents를 참조하십시오.

이벤트 알림 및 Amazon SNS

ElastiCache는 캐시 클러스터에서 중요 이벤트가 발생하는 경우 Amazon Simple Notification Service(SNS)를 사용하여 메시지를 게시할 수 있습니다. 이 기능은 캐시 클러스터의 개별 캐시 노드 Endpoint에 연결된 클라이언트 머신의 서버 목록을 새로 고침하는 데 사용될 수 있습니다.

Note

이용 요금 정보 및 Amazon SNS 설명서 링크 등 Amazon Simple Notification Service(SNS)에 대한 자세한 내용은 Amazon SNS 제품 페이지를 참조하십시오.

알림은 지정된 Amazon SNS 주제에 대해 게시됩니다. 다음은 알림에 대한 요구 사항입니다.

- ElastiCache 알림에 대해 주제 하나만 구성할 수 있습니다.
- Amazon SNS 주제를 소유한 AWS 계정은 알림이 활성화된 캐시 클러스터를 소유하는 계정과 동일해야 합니다.

예제 ElastiCache SNS 알림

다음은 캐시 클러스터 생성 성공에 대한 ElastiCache Amazon SNS 알림을 보여주는 예입니다.

Example

```
{
    "Date": "2015-12-05T01:02:18.336Z",
    "Message": "Cache cluster created",
    "SourceIdentifier": "memcache-ni",
    "SourceType": "cache-cluster"
}
```

ElastiCache 이벤트

다음 ElastiCache이벤트는 Amazon SNS 알림을 트리거합니다.

이벤트 이름	메시지	설명
ElastiCache:AddCacheNodeComp	eteuster-name	캐시 노드가 캐시 클러스터에 추 가되었고 사용할 준비가 되어 있 습니다.
ElastiCache무료 IP 주 소가 부족함으로 인 한 :AddCacheNodeFailed	cluster-name	사용 가능한 IP 주소가 충분하지 않아 캐시 노드를 추가하지 못했 습니다.
ElastiCache:CacheClusterParamet	ecsClsangechame	하나 이상의 캐시 클러스터 매개 변수가 변경되었습니다.

이벤트 이름	메시지	설명
ElastiCache:CacheClusterProvision	ningConpleteame	캐시 클러스터의 프로비저닝이 완 료되어 캐시 클러스터에 있는 캐 시 노드를 사용할 수 있습니다.
ElastiCache호환되지 않 는 네트워크 상태로 인 한 :CacheClusterProvisioningFailed	cluster-name	존재하지 않는 Virtual Private Cloud(VPC)에서 새로운 캐시 클 러스터를 실행하려고 시도했습니 다.
ElastiCache:CacheClusterScalingC	Campleter-name	: 캐시 클러스터의 스케일 업이 성 공적으로 완료되었습니다.
ElastiCache:CacheClusterScalingF	ailēdster-name	캐시 클러스터에 대한 스케일 업 작업이 실패했습니다.
ElastiCache:CacheClusterSecurity	GraupMedified <i>me</i>	다음 이벤트 중 하나가 발생했습 니다.
		 캐시 클러스터를 위한 승인된 캐시 보안 그룹이 수정되었습니다. 하나 이상의 새로운 EC2 보안 그룹이 캐시 클러스터와 연결된 캐시 보안 그룹 중 하나에서 승인되었습니다. 하나 이상의 EC2 보안 그룹이 캐시 클러스터와 연결된 캐시 보안 그룹 중 하나에서 승인이 취소되었습니다.
ElastiCache:CacheNodeReplaceSt	artedster-name	ElastiCache가 캐시 노드를 실행하는 호스트 성능이 저하되었거나 연결되지 않음을 감지하여 캐시노드 교체를 시작했습니다. Note 교체된 캐시 노드의 DNS 항목은 변경되지 않습니다. 대부분의 경우에 이 이벤트가 발생할 때 클라이언트의 서버 목록을 새로 고침하지 않아도 됩니다.하지만 일부 캐시 클라이언트 라이브러리는 ElastiCache가 캐시노드를 교체한 후에도 캐시노드 사용을 중단할 수 있습니다. 이 경우에 애플리케이션은 이 이벤트가 발생할 때 서버 목록을 새로 고침해야 합니다.

이벤트 이름	메시지	설명
ElastiCache:CacheNodeReplaceC	o mplete er-name	ElastiCache가 캐시 노드를 실행 하는 호스트 성능이 저하되었거나 연결되지 않음을 감지하여 캐시 노드 교체를 완료했습니다.
		Note
		교체된 캐시 노드의 DNS 항목은 변경되지 않습니 다.
		대부분의 경우에 이 이벤트가 발생할 때 클라이언트의 서버 목록을 새로 고침하지 않아도 됩니다.하지만 일부 캐시 클라이언트 라이브러리는 ElastiCache가 캐시노드를 교체한 후에도 캐시노드 사용을 중단할 수 있습니다. 이 경우에 애플리케이션은 이 이벤트가발생할 때 서버 목록을 새로 고침해야합니다.
ElastiCache:CacheNodesReboote	dcluster-name	하나 이상의 캐시 노드가 재부팅 되었습니다.
		메시지(Memcached): "Cache node %s shutdown", 두 번 째 메시지: "Cache node %s restarted"
ElastiCache:DeleteCacheClusterC	o mplete er-name	캐시 클러스터 및 연결된 모든 캐 시 노드 삭제를 완료했습니다.
ElastiCache:FailoverComplete(Red 전용)	li <mark>sluster-name</mark>	복제본 노드에 대한 장애 조치가 성공했습니다.
ElastiCache:NodeReplacementCa	n <mark>ælled</mark> ter-name	교체가 예약되어 있는 클러스터의 노드가 더 이상 교체 예약이 되지 않습니다.
ElastiCache:NodeReplacementRe	s <mark>atiedsted</mark> r-name	이전에 교체가 예약되어 있는 클 러스터의 노드가 알림에 설명된 새 기간 동안 교체가 예약됩니다.
		수행할 수 있는 작업에 대한 자세 한 내용은 노드 교체 (p. 60) 단원 을 참조하십시오.
ElastiCache:NodeReplacementSch	n edule cter-name	클러스터의 노드가 알림에 설명된 기간 동안 교체가 예약됩니다.
		수행할 수 있는 작업에 대한 자세 한 내용은 노드 교체 (p. 60) 단원 을 참조하십시오.
ElastiCache:RemoveCacheNodeC	o <mark>mplete</mark> er-name	캐시 노드가 캐시 클러스터에서 제거되었습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 태그를 사용한 비용 모니터링

이벤트 이름	메시지	설명
ElastiCache:SnapshotComplete(Re 전용)	e <mark>di\$uster-name</mark>	캐시 스냅샷이 성공적으로 완료되 었습니다.
ElastiCache:SnapshotFailed(Redis 전용)	cluster-name	캐시 스냅샷이 실패했습니다. 원 인에 대한 자세한 내용은 클러스 터의 캐시 이벤트를 참조하십시 오.
		스냅샷을 설명할 경우 DescribeSnapshots를 참조하 십시오. 상태는 failed입니다.

관련 항목

• ElastiCache 이벤트 보기 (p. 370)

비용 할당 태그를 사용한 비용 모니터링

Amazon ElastiCache에서 리소스에 비용 할당 태그를 추가하면 인보이스 비용을 리소스 태그 값으로 그룹화 하여 비용을 추적할 수 있습니다.

ElastiCache 비용 할당 태그는 사용자가 정의하고 ElastiCache 리소스에 연결하는 키 값 페어입니다. 키와 값은 대/소문자를 구분합니다. 태그 키를 사용하여 범주를 정의할 수 있으며 태그 값은 해당 범주의 항목일 수있습니다. 예를 들어, CostCenter의 태그 키와 10010의 태그 값을 정의하여 리소스가 10010 코스트 센터에 할당됨을 나타냅니다. Environment와 같은 키 및 test 또는 production과 같은 값을 사용하여 태그로 리소스를 테스트나 프로덕션에 사용되는 것으로 지정할 수도 있습니다. 리소스 관련 비용을 보다 쉽게 추적할 수 있도록 하기 위해 일관된 태그 키 세트를 사용하는 것이 좋습니다.

비용 할당 태그를 사용하여 자신만의 비용 구조를 반영하도록 AWS 청구서를 구성합니다. 이렇게 하려면 가입하여 태그 키 값이 포함된 AWS 계정 청구서를 가져옵니다. 그런 다음 같은 태그 키 값을 가진 리소스에 따라 결제 정보를 구성하여 리소스 비용의 합을 볼 수 있습니다. 예를 들어, 특정 애플리케이션 이름으로 여러리소스에 태그를 지정한 다음 결제 정보를 구성하여 여러 서비스에 걸친 해당 애플리케이션의 총 비용을 볼수 있습니다.

또한 태그를 결합하여 보다 세부적인 수준으로 비용을 추적할 수 있습니다. 예를 들어, 리전별 서비스 비용을 추적하려면 Service 및 Region 태그 키를 사용할 수 있습니다. 하나의 리소스에서 ElastiCache 및 ### ###(####) 값이 있을 수 있으며 다른 리소스에서 ElastiCache 및 ##(#####) 값이 있을 수 있습니다. 리전별로 구분된 총 ElastiCache 비용을 볼 수 있습니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 비용 할당 태그 사용을 참조하십시오.

Redis 노드에 ElastiCache 비용 할당 태그를 추가할 수 있습니다. 태그를 추가, 나열, 수정, 복사 또는 제거할 때 이 작업은 지정된 노드에만 적용됩니다.

ElastiCache 비용 할당 태그 특성

• 비용 할당 태그는 CLI 및 API 작업에서 ARN으로 지정된 ElastiCache 리소스에 적용됩니다. 리소스 유형은 "클러스터"입니다.

샘플 ARN: arn:aws:elasticache:<region>:<customer-id>:<resource-type>:<resource-name>

샘플 arn: arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 태그를 사용한 비용 모니터링

- 태그 키는 태그의 필수 이름입니다. 키의 문자열 값은 1~128자(유니코드 문자) 사이가 될 수 있으며 aws:로 시작할 수 없습니다. 문자열에는 유니코드 문자, 숫자, 공백, 밑줄(_), 마침표(.), 콜론(:), 백슬래시(\), 등호(=), 더하기 기호(+), 하이픈(-), at 기호(@) 집합만 포함될 수 있습니다.
- 태그 값은 태그의 선택적 값입니다. 값의 문자열 값은 1~256자(유니코드 문자) 사이가 될 수 있으며 aws:로 시작할 수 없습니다. 문자열에는 유니코드 문자, 숫자, 공백, 밑줄(_), 마침표(.), 콜론(:), 백슬래시(\), 등호(=), 더하기 기호(+), 하이픈(-), at 기호(@) 집합만 포함될 수 있습니다.
- ElastiCache 리소스는 최대 50개의 태그를 보유할 수 있습니다.
- 태그 세트의 값이 고유하지 않습니다. 예를 들어, 두 키 Service와 Application에 ElastiCache 값이 있는 태그 세트가 있을 수 있습니다.

AWS는 태그에 의미론적 의미를 적용하지 않습니다. 태그는 엄격히 문자열로 해석됩니다. AWS에서는 ElastiCache 리소스에 어떠한 태그도 자동으로 설정하지 않습니다.

ElastiCache 관리 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 ElastiCache 리소스의 태그를 추가, 나열, 수정 또는 제거할 수 있습니다.

주제

- ElastiCache 콘솔을 사용한 태그 관리 (p. 376)
- AWS CLI를 사용하여 비용 할당 태그 관리 (p. 378)
- ElastiCache API를 사용하여 비용 할당 태그 관리 (p. 381)

ElastiCache 콘솔을 사용한 태그 관리

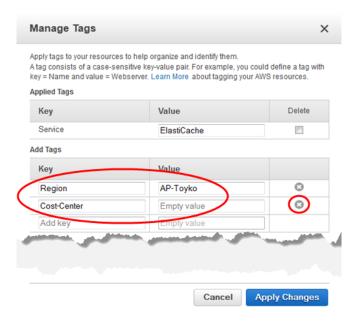
Amazon ElastiCache 콘솔을 사용하여 비용 할당 태그를 추가, 수정 또는 제거할 수 있습니다.

다음 절차에서는 ElastiCache 관리 콘솔을 사용하여 하나 이상의 비용 할당 태그를 보거나 추가, 수정 또는 삭제하는 과정을 안내합니다.

Redis 클러스터에는 샤드가 없거나 한 개 또는 여러 개 있을 수 있습니다. 태그는 전체 클러스터가 아닌 Redis 노드에 추가되므로 AWS Management 콘솔 사용 시 Redis 클러스터에서 태그를 관리하는 절차는 세 가지 구성 각각에 대해 약간 다릅니다. 다음 두 절차 중 한 가지를 수행하여 Redis 클러스터에서 태그를 관리 합니다.

AWS Management 콘솔을 사용하여 샤드가 없는 Redis 클러스터에서 태그 관리

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. [Redis]를 선택합니다.
- 3. 태그를 추가할 샤드 없는 클러스터의 이름을 선택합니다.
- 4. 클러스터의 노드 이름 왼쪽에 있는 확인란을 선택합니다.
- 5. [Actions] 목록에서 [Manage Tags]를 선택한 다음 대화 상자를 사용하여 태그를 관리합니다.



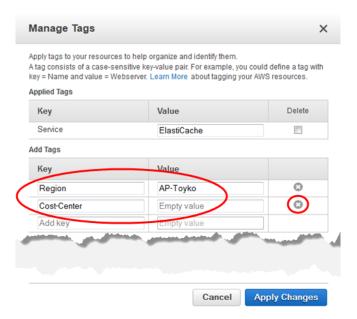
6. 추가, 수정 또는 제거하려는 각 태그에 대해 다음을 수행합니다.

태그를 추가, 수정 또는 제거하려면

- 태그를 추가하려면 [Key] 열의 [Add key]가 표시된 상자에 키 이름을 입력하고 키 이름 오른쪽에 있는 상자에 선택적 값을 입력합니다.
- 태그를 수정하려면 [Value] 열에 새 값을 입력하거나 태그의 기존 값을 제거합니다.
- 태그를 제거하려면 태그 오른쪽에 있는 [X]를 선택합니다.
- 7. 완료되면 [Apply Changes]를 선택합니다.

AWS Management 콘솔을 사용하여 샤드가 한 개 이상인 Redis 클러스터에서 태그 관리

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. [Redis]를 선택합니다.
- 3. 태그를 추가할 샤드 있는 클러스터의 이름을 선택합니다.
- 4. 이 클러스터에 샤드가 하나만 있는 경우 6단계로 건너뜁니다. 이 클러스터에 샤드가 여러 개 있는 경우 5 단계부터 계속합니다.
- 5. 샤드 목록에서 태그를 추가할 노드가 있는 샤드의 이름을 선택합니다.
- 6. 태그를 추가할 노드의 이름 왼쪽에 있는 확인란을 선택합니다.
- 7. [Actions] 목록에서 [Manage Tags]를 선택한 다음 대화 상자를 사용하여 태그를 관리합니다.



8. 추가, 수정 또는 제거하려는 각 태그에 대해 다음을 수행합니다.

태그를 추가, 수정 또는 제거하려면

- 태그를 추가하려면 [Key] 열의 [Add key]가 표시된 상자에 키 이름을 입력하고 키 이름 오른쪽에 있는 상자에 선택적 값을 입력합니다.
- 태그를 수정하려면 [Value] 열에 새 값을 입력하거나 태그의 기존 값을 제거합니다.
- 태그를 제거하려면 태그 오른쪽에 있는 [X]를 선택합니다.
- 9. 완료되면 [Apply Changes]를 선택합니다.

AWS CLI를 사용하여 비용 할당 태그 관리

AWS CLI를 사용하여 비용 할당 태그를 추가, 수정 또는 제거할 수 있습니다.

샘플 arn: arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster

비용 할당 태그는 Redis용 ElastiCache 노드에 적용됩니다. 태그를 지정할 노드는 Amazon 리소스 이름 (ARN)을 사용해 지정합니다.

샘플 arn: arn: aws:elasticache:us-west-2:1234567890:cluster:my-cluster

주제

- AWS CLI를 사용하여 태그 나열 (p. 378)
- AWS CLI를 사용하여 태그 추가 (p. 379)
- AWS CLI를 사용하여 태그 수정 (p. 380)
- AWS CLI를 사용하여 태그 제거 (p. 380)

AWS CLI를 사용하여 태그 나열

list-tags-for-resource 작업을 사용하여 기존 ElastiCache 리소스에서 태그를 나열하는 데 AWS CLI를 사용할 수 있습니다.

다음 코드에서는 AWS CLI를 사용하여 us-west-2 리전에 있는 my-cluster 클러스터의 Redis 노드 my-cluster-001에 태그를 나열합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache list-tags-for-resource \
    --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001
```

Windows의 경우:

```
aws elasticache list-tags-for-resource ^
--resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001
```

이 작업의 출력은 다음 리소스의 모든 태그 목록과 유사합니다.

리소스에 태그가 없으면 출력은 빈 TagList가 됩니다.

```
{
    "TagList": []
}
```

자세한 내용은 ElastiCache에 대한 AWS CLI list-tags-for-resource를 참조하십시오.

AWS CLI를 사용하여 태그 추가

add-tags-to-resource CLI 작업을 사용하여 기존 ElastiCache 리소스에 태그를 추가하는 데 AWS CLI를 사용할 수 있습니다. 리소스에 태그 키가 없으면 키와 값이 리소스에 추가됩니다. 리소스에 이미 키가 있는 경우해당 키와 연결된 값이 새 값으로 업데이트됩니다.

다음 코드에서는 AWS CLI를 사용하여 us-west-2 리전의 my-cluster 클러스터에 있는 my-cluster-001노드 에 각각 elasticache 및 us-west-2 값을 갖는 Service 및 Region 키를 추가합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache add-tags-to-resource \
   --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 \
   --tags Key=Service, Value=elasticache \
   Key=Region, Value=us-west-2
```

Windows의 경우:

```
aws elasticache add-tags-to-resource ^
    --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 ^
    --tags Key=Service, Value=elasticache ^
    Key=Region, Value=us-west-2
```

이 작업의 출력은 다음 작업 후 리소스의 모든 태그 목록과 유사합니다.

자세한 내용은 ElastiCache에 대한 AWS CLI add-tags-to-resource를 참조하십시오.

AWS CLI을 통해 create-cache-cluster 작업을 사용하여 새 클러스터를 생성할 때 클러스터에 태그를 추가할 수도 있습니다. ElastiCache 관리 콘솔을 사용하는 경우 클러스터를 생성할 때 태그를 추가할 수 없습니다. 클러스터가 생성된 후에는 콘솔을 사용하여 클러스터에 태그를 추가할 수 있습니다.

AWS CLI를 사용하여 태그 수정

AWS CLI를 사용하여 Redis용 ElastiCache 클러스터의 노드에서 태그를 수정할 수 있습니다.

태그를 수정하려면

- add-tags-to-resource를 사용하여 새 태그 및 값을 추가하거나 기존 태그에 연결된 값을 변경합니다.
- remove-tags-from-resource를 사용하여 리소스에서 지정된 태그를 제거합니다.

두 작업 중 하나의 출력은 지정된 클러스터의 태그와 이 태그의 값이 나열된 목록입니다.

AWS CLI를 사용하여 태그 제거

remove-tags-from-resource 작업을 사용하여 Redis용 ElastiCache 클러스터의 기존 노드에서 태그를 제거하는 데 AWS CLI를 사용할 수 있습니다.

다음 코드에서는 AWS CLI를 사용하여 us-west-2 리전의 my-cluster 클러스터 에 있는 my-cluster-001노드 에서 Service 및 Region 키를 가진 태그를 제거합니다.

Linux, macOS, or Unix의 경우:

```
aws elasticache remove-tags-from-resource \
--resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 \
--tag-keys PM Service
```

Windows의 경우:

```
aws elasticache remove-tags-from-resource ^
--resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 ^
--tag-keys PM Service
```

이 작업의 출력은 다음 작업 후 리소스의 모든 태그 목록과 유사합니다.

```
{
    "TagList": []
```

}

자세한 내용은 ElastiCache에 대한 AWS CLI remove-tags-from-resource를 참조하십시오.

ElastiCache API를 사용하여 비용 할당 태그 관리

ElastiCache API를 사용하여 비용 할당 태그를 추가, 수정 또는 제거할 수 있습니다.

비용 할당 태그는 Memcached용 ElastiCache 클러스터에 적용됩니다. 태그를 지정할 클러스터는 Amazon 리소스 이름(ARN)을 사용해 지정합니다.

샘플 arn: arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster

주제

- ElastiCache API를 사용하여 태그 나열 (p. 381)
- ElastiCache API를 사용하여 태그 추가 (p. 381)
- ElastiCache API를 사용하여 태그 수정 (p. 382)
- ElastiCache API를 사용하여 태그 제거 (p. 382)

ElastiCache API를 사용하여 태그 나열

ElastiCache API를 통해 ListTagsForResource 작업을 사용하여 기존 리소스에서 태그를 나열할 수 있습니다.

다음 코드는 ElastiCache API를 사용하여 us-west-2 리전에 있는 리소스 my-cluster-001의 태그를 나열합니다.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=ListTagsForResource
&ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Version=2015-02-02
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

ElastiCache API를 사용하여 태그 추가

AddTagsToResource 작업을 사용하여 기존 ElastiCache 클러스터에 태그를 추가하는 데 ElastiCache API를 사용할 수 있습니다. 리소스에 태그 키가 없으면 키와 값이 리소스에 추가됩니다. 리소스에 이미 키가 있는 경우 해당 키와 연결된 값이 새 값으로 업데이트됩니다.

다음 코드는 ElastiCache API 사용하여 us-west-2 리전의 리소스 my-cluster-001에 각각 elasticache 및 us-west-2 값을 갖는 Service 및 Region 키를 추가합니다.

&X-Amz-Credential=<credential>

자세한 내용은 Amazon ElastiCache API Reference의 AddTagsToResource를 참조하십시오.

ElastiCache API를 사용하여 태그 수정

ElastiCache API를 사용하여 ElastiCache 클러스터의 태그를 수정할 수 있습니다.

태그 값을 수정하려면

- AddTagsToResource 작업을 사용하여 새 태그와 값을 추가하거나 기존 태그의 값을 변경합니다.
- RemoveTagsFromResource를 사용하여 리소스에서 태그를 제거합니다.

두 작업 중 하나의 출력은 지정된 리소스의 태그 목록과 값입니다.

RemoveTagsFromResource를 사용하여 리소스에서 태그를 제거합니다.

ElastiCache API를 사용하여 태그 제거

RemoveTagsFromResource 작업을 사용하여 기존 Redis용 ElastiCache 노드에서 태그를 제거하는 데 ElastiCache API를 사용할 수 있습니다.

다음 코드에서는 ElastiCache API를 사용하여 us-west-2 리전의 my-cluster 클러스터 에 있는 my-cluster-001 노드 에서 Service 및 Region 키를 가진 태그를 제거합니다.

https://elasticache.us-west-2.amazonaws.com/
 ?Action=RemoveTagsFromResource
 &ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001
 &SignatureVersion=4
 &SignatureMethod=HmacSHA256
 &TagKeys.member.1=Service
 &TagKeys.member.2=Region
 &Version=2015-02-02
 &Timestamp=20150202T192317Z
 &X-Amz-Credential=<credential>

예약 노드를 통해 비용 관리

하나 이상의 노드를 예약하여 비용을 줄일 수 있습니다. 노드 유형과 예약 기간—(1년 또는 3년)에 따라 예약 노드에 선납금이 부과됩니다. 선결제 요금 외에도 시간당 사용 요금이 부과됩니다. 이 요금은 온디맨드 노드 에서 발생하는 시간당 사용 요금보다 훨씬 낮습니다.

예약된 노드가 사용 사례에 대해 비용이 절감되는지 확인하려면 먼저 필요한 노드 수와 노드 크기를 결정합니다. 그런 다음 노드의 사용량을 예측하고 온디맨드 노드와 예약된 노드의 총 비용을 비교합니다. 클러스터에서 예약 노드와 온디맨드 노드를 함께 사용할 수 있습니다. 요금 정보는 Amazon ElastiCache 요금을 참조하십시오.

AWS Management 콘솔, AWS CLI 또는 ElastiCache API를 사용하여 사용 가능한 예약 노드 제품을 나열하고 구입할 수 있습니다.

예약 노드에 대한 자세한 내용은 Amazon ElastiCache 예약 캐시 노드를 참조하십시오.

주제

- 표준 예약 노드 캐시 유형 (p. 383)
- 이전 예약 노드 캐시 제공 (p. 383)

- 예약 노드 상품에 대한 정보 가져오기 (p. 384)
- 예약 노드 구입 (p. 386)
- 예약 노드에 대한 정보 가져오기 (p. 389)

표준 예약 노드 캐시 유형

Amazon ElastiCache에서 표준 예약 인스턴스(RI)를 구매할 때는 예약 노드 인스턴스의 기간 동안 특정 캐시 노드 인스턴스 유형 및 AWS 리전에 대해 할인 요금을 이용하는 약정을 구매하는 것입니다. Amazon ElastiCache 예약 노드 인스턴스를 사용하려면 온디맨드 인스턴스와 똑같은 방법으로 새 ElastiCache 노드 인스턴스를 생성해야 합니다.

새 노드 인스턴스는 예약 노드 인스턴스의 사양과 정확히 일치해야 합니다. 새로운 노드 인스턴스의 사양이 계정의 기존 예약 노드 인스턴스와 일치하면 예약 인스턴스에 제공되는 할인 요금이 청구됩니다. 그렇지 않으면 노드 인스턴스에 대해 온디맨드 요금이 청구됩니다. 이 표준 RI는 R5 및 M5 인스턴스 패밀리부터 사용할 수 있습니다.

Note

다음에 논의되는 세 가지 제공 유형은 1년과 3년 단위로 사용할 수 있습니다.

제공 유형

선수금 없음 RI는 선결제 금액 없이 예약된 ElastiCache 인스턴스에 대한 액세스를 제공합니다. 선결제 없음 예약 ElastiCache 인스턴스는 사용되는지 여부와 상관없이 사용 기간 동안 매시간마다 할인된 시간당 요금 이 청구됩니다.

부분 선결제 RI의 경우 예약된 ElasticCache 인스턴스의 일부를 먼저 결제해야 합니다. 결제하지 않은 시간에 대해서는 사용 기간 동안 사용량에 상관없이 할인된 시간당 요금이 청구됩니다. 이 옵션은 다음 단원에서 설명할 이전 Heavy 사용률 옵션을 대신합니다.

전체 선결제 RI의 경우 RI 사용 기간이 시작될 때 전액 지불해야 합니다. 사용 시간과 관계없이 남은 기간 동안 다른 비용은 발생하지 않습니다.

이전 예약 노드 캐시 제공

세 가지의 이전 노드 예약 수준에는 — Heavy 사용률, Medium 사용률 및 Light 사용률이 있습니다. 어느 사용률 수준으로나 1년 또는 3년간 노드를 예약할 수 있습니다. 노드 유형, 사용률 수준 및 예약 기간에 따라 총비용이 정해집니다. 예약 노드를 구입하기 전에 다양한 모델을 비교하여 예약 노드가 회사에 줄 수 있는 절약효과를 확인하십시오.

특정 사용률 수준이나 기간으로 구입한 노드를 다른 사용률 수준이나 기간으로 바꿀 수 없습니다.

사용률 수준

Heavy 사용률 예약 노드는 용량 기준이 일관된 워크로드를 가능하게 하거나 되거나 상태가 꾸준한 워크로드를 실행합니다. Heavy 사용률 예약 노드는 높은 선납금 약정이 필요하지만 예약 노드 기간의 79% 이상을 실행하려는 경우 가장 크게 비용을 절감할 수 있습니다(온디맨드 요금의 70%까지). Heavy 사용률 예약 노드의경우, 일회성 요금을 지불합니다. 이는 노드가 실행 여부에 관계없이 사용 기간 동안 시간당 요금이 더 낮습니다.

Medium 사용률 예약 노드는 많은 기간 예약 노드를 이용할 계획이고 더 저렴한 일회성 요금을 원하거나 노드 사용을 종료할 때 노드 요금 지불을 중지할 수 있기를 원하는 경우 가장 적합한 옵션입니다. Medium 사용률 예약 노드는 예약 노드 기간의 40% 이상을 실행하려는 경우 더욱 비용 효율적인 옵션입니다. 이 옵션은 온 디맨드 가격의 최대 64%까지 절감할 수 있습니다. Medium 사용률 예약 노드를 사용하면 Light 사용률 예약 노드를 사용하는 것보다 약간 높은 일회성 요금을 지불하지만 노드를 실행할 때 시간당 사용 요금을 낮출수 있습니다.

Light 사용률 예약 노드는 매일 두세 시간 또는 일주일에 며칠 정도 실행하는 정기 작업에 적합합니다. Light 사용률 예약 노드를 사용하면 일회성 요금을 지불한 후 노드를 실행할 때 할인된 시간당 요금을 지불하게 됩니다. 노드가 예약된 노드 사용 기간의 17% 이상을 실행 중인 경우, 비용을 절감할 수 있습니다. 예약된 노드의 전체 사용 기간 동안 온디맨드 요금의 최대 56%까지 절감할 수 있습니다.

이전 예약 캐시 노드 상품

제공 유형	선납금	사용료	이점
Heavy 사용률	가장 높음	시간당 가장 낮은 요금. 예약 노드를 사용하고 있 는지 상관없이 전체 기간 에 적용됩니다.	예상되는 예약 노드 사용 률이 3년 약정 기준으로 79% 이상인 경우 전체 적인 비용이 가장 낮습니 다.
Medium 사용률	Medium	노드를 실행하는 매 시간 마다 시간당 사용 요금이 청구됩니다. 노드가 실행 되지 않을 때는 시간당 요금이 청구되지 않습니 다.	탄력적인 워크로드 또는 3년 약정 기간에 40% 이 상의 보통 사용량이 필요 한 경우에 적합합니다.
Light 사용률	가장 낮음	노드를 실행하는 매 시간 마다 시간당 사용 요금이 청구됩니다. 노드가 실행 되지 않을 때는 시간당 요금이 청구되지 않습니 다. 모든 제공 유형 중에 서 시간당 요금이 가장 높지만 예약 노드가 실행 되고 있을 때만 요금이 적용됩니다.	항상 실행할 계획이라면 전체 비용이 가장 높습니 다. 그러나, 이는 3년 약 정 기간의 약 15% 이상 예약 노드를 자주 사용하 지 않는 경우 전체 비용 이 가장 낮습니다.
온디맨드 사용(예약 노 드 없음)	없음	시간당 요금이 가장 높습 니다. 노드가 실행 중일 때마다 적용됩니다.	시간당 비용이 가장 높습 니다.

자세한 내용은 Amazon ElastiCache 요금을 참조하십시오.

예약 노드 상품에 대한 정보 가져오기

예약 노드를 구입하기 전에 사용 가능한 예약 노드 상품에 대한 정보를 확인할 수 있습니다.

다음 예제에서는 AWS Management 콘솔, AWS CLI 및 ElastiCache API를 사용하여 사용 가능한 예약 노드 상품의 요금과 정보를 확인하는 방법을 보여줍니다.

주제

- 예약 노드 상품에 대한 정보 가져오기 (콘솔) (p. 384)
- 예약 노드 상품에 대한 정보 가져오기 (AWS CLI) (p. 385)
- 예약 노드 상품에 대한 정보 가져오기 (ElastiCache API) (p. 386)

예약 노드 상품에 대한 정보 가져오기 (콘솔)

AWS Management 콘솔을 사용하여 사용 가능한 예약 클러스터 상품의 요금과 그 밖의 정보를 가져오려면 다음 절차를 따르십시오.

사용 가능한 예약 노드 상품에 대한 정보를 가져오려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 창에서 Reserved Cache Nodes(예약 캐시 노드)를 선택합니다.
- 3. [Purchase Reserved Cache Node]를 선택합니다.
- 4. 제품 설명에서 Redis를 선택합니다.
- 5. 사용 가능한 상품을 확인하려면 다음 3개 목록 중에서 선택합니다.
 - 캐시 노드 유형
 - 기간
 - Offering Type(제공 유형)

선택한 후 노드당 비용과 선택 항목의 총 비용이 [Purchase Reserved Cache Nodes] 마법사에 표시됩니다.

6. [Cancel]을 선택하면 이 노드를 구입하지 않으며 요금이 발생하지 않습니다.

예약 노드 상품에 대한 정보 가져오기 (AWS CLI)

사용 가능한 예약 노드 상품의 요금과 그 밖의 정보를 가져오려면 명령 프롬프트에 다음 명령을 입력합니다.

```
aws elasticache describe-reserved-cache-nodes-offerings
```

이 작업은 다음과 비슷한 출력을 생성합니다(JSON 형식).

```
"ReservedCacheNodesOfferings": [
          "OfferingType": "Heavy Utilization",
          "FixedPrice": 4328.0,
          "ReservedCacheNodesOfferingId": "0192caa9-daf2-4159-b1e5-a79bb1916695",
          "UsagePrice": 0.0,
          "RecurringCharges": [
                  "RecurringChargeAmount": 0.491,
                  "RecurringChargeFrequency": "Hourly"
          "ProductDescription": "memcached",
          "Duration": 31536000,
          "CacheNodeType": "cache.r3.4xlarge"
     },
****** some output omitted for brevity *******
     {
          "OfferingType": "Heavy Utilization",
          "FixedPrice": 4132.0,
          "ReservedCacheNodesOfferingId": "fb766e0a-79d7-4e8f-a780-a2a6ed5ed439",
          "UsagePrice": 0.0,
          "RecurringCharges": [
                  "RecurringChargeAmount": 0.182,
                  "RecurringChargeFrequency": "Hourly"
          "ProductDescription": "redis",
          "Duration": 94608000,
```

자세한 내용은 AWS CLI 참조의 describe-reserved-cache-nodes-offerings를 참조하십시오.

예약 노드 상품에 대한 정보 가져오기 (ElastiCache API)

사용 가능한 예약 노드 상품의 요금과 정보를 가져오려면 DescribeReservedCacheNodesOfferings 작업을 호출하십시오.

Example

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=DescribeReservedCacheNodesOfferings
    &Version=2014-12-01
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20141201T220302Z
    &X-Amz-Algorithm
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Signature<
```

이 호출은 다음과 비슷한 출력을 반환합니다.

```
<DescribeReservedCacheNodesOfferingsResponse xmlns="http://elasticache.us-</pre>
west-2.amazonaws.com/doc/2013-06-15/">
  <DescribeReservedCacheNodesOfferingsResult>
    <ReservedCacheNodesOfferings>
      <ReservedCacheNodesOffering>
        <Duration>31536000</Duration>
        <OfferingType>Medium Utilization</OfferingType>
        <CurrencyCode>USD</CurrencyCode>
        <RecurringCharges/>
        <FixedPrice>1820.0</FixedPrice>
        <ProductDescription>memcached</productDescription>
        <UsagePrice>0.368</UsagePrice>
        <ReservedCacheNodesOfferingId>438012d3-4052-4cc7-b2e3-8d3372e0e706/
ReservedCacheNodesOfferingId>
        <CacheNodeType>cache.m1.large</CacheNodeType>
      </ReservedCacheNodesOffering>
      <ReservedCacheNodesOffering>
      (...some output omitted for brevity...)
      </ReservedCacheNodesOffering>
    </ReservedCacheNodesOfferings>
  </DescribeReservedCacheNodesOfferingsResult>
  <ResponseMetadata>
    <RequestId>5e4ec40b-2978-11e1-9e6d-771388d6ed6b</RequestId>
  </ResponseMetadata>
</DescribeReservedCacheNodesOfferingsResponse>
```

자세한 내용은 ElastiCache API의 DescribeReservedCacheNodesOfferings를 참조하십시오.

예약 노드 구입

다음 예제에서는 AWS Management 콘솔, AWS CLI 및 ElastiCache API를 사용하여 예약 노드 상품을 구입하는 방법을 설명합니다.

Important

이 단원의 예제에 따라 예약할 수 없는 AWS 계정에 요금이 부과됩니다.

주제

- 예약 노드 구입 (콘솔) (p. 387)
- 예약 노드 구입 (AWS CLI) (p. 387)
- 예약 노드 구입 (ElastiCache API) (p. 388)

예약 노드 구입 (콘솔)

이 예제에서는 예약 노드 ID가 myreservationID인 특정 예약 노드 상품 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f를 구입하는 방법을 보여줍니다.

다음 절차에서는 AWS Management 콘솔을 사용하여 상품 ID로 예약 노드 상품을 구입합니다.

예약 노드를 구입하려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 목록에서 [Reserved Cache Nodes] 링크를 선택합니다.
- 3. [Purchase Reserved Cache Node] 버튼을 선택합니다.
- 4. [Product Description] 드롭다운 목록 상자에서 노드 유형을 선택합니다.
- 5. [Cache Node Class] 드롭다운 목록 상자에서 노드 클래스를 선택합니다.
- 6. [Term] 드롭다운 목록 상자에서 노드를 예약할 기간을 선택합니다.
- 7. 다음 중 하나를 수행하십시오.
 - [Offering Type] 드롭다운 목록 상자에서 제공 유형을 선택합니다.
 - [Reserved Cache Node ID] 텍스트 상자에 예약 노드 ID를 입력합니다.

Note

예약 캐시 노드 ID는 고객마다 고유한 식별자이며 이 예약을 추적하는 데 사용됩니다. 이 상자를 비워 두면 ElastiCache에서 자동으로 예약 식별자를 생성합니다.

8. [Next] 버튼을 선택합니다.

[Purchase Reserved Cache Node] 대화 상자에 선택한 예약 노드 속성에 대한 요약 내용과 지불 기한이 표시됩니다.

9. 계속하려면 [Yes, Purchase] 버튼을 선택하고 예약 노드를 구입합니다.

Important

[Yes, Purchase]를 선택하면 선택한 예약 노드에 요금이 부과됩니다. 요금이 부과되지 않도록 하려면 [Cancel]을 선택합니다.

예약 노드 구입 (AWS CLI)

다음 예제에서는 예약 노드 ID가 myreservationID인 특정 예약 클러스터 상품 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f를 구입하는 방법을 보여줍니다.

명령 프롬프트에서 다음 명령을 입력합니다.

Linux, macOS, or Unix의 경우:

aws elasticache purchase-reserved-cache-nodes-offering \

```
--reserved-cache-nodes-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f \
--reserved-cache-node-id myreservationID
```

Windows의 경우:

```
aws elasticache purchase-reserved-cache-nodes-offering ^
--reserved-cache-nodes-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f ^
--reserved-cache-node-id myreservationID
```

이 명령은 다음과 비슷한 출력을 반환합니다.

```
RESERVATION ReservationId Class Start Time Duration Fixed Price Usage Price Count State Description Offering Type RESERVATION myreservationid cache.ml.small 2013-12-19T00:30:23.247Z 1y 455.00 USD 0.092 USD 1 payment-pending memcached Medium Utilization
```

자세한 내용은 AWS CLI 참조의 purchase-reserved-cache-nodes-offering을 참조하십시오.

예약 노드 구입 (ElastiCache API)

다음 예제에서는 예약 클러스터 노드 ID가 myreservationID인 특정 예약 상품 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f를 구입하는 방법을 보여줍니다.

다음 파라미터와 함께 PurchaseReservedCacheNodesOffering 작업을 호출합니다.

- ReservedCacheNodesOfferingId = 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f
- ReservedCacheNodeID = myreservationID
- CacheNodeCount = 1

Example

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=PurchaseReservedCacheNodesOffering
    &ReservedCacheNodesOfferingId=649fd0c8-cf6d-47a0-bfa6-060f8e75e95f
    &ReservedCacheNodeID=myreservationID
    &CacheNodeCount=1
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20141201T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20141201T220302Z
    &X-Amz-Date=20141201T220302Z
    &X-Amz-SignedHeaders=Host
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Credential=<credential>
    &X-Amz-Signature=<signature>
```

이 호출은 다음과 비슷한 출력을 반환합니다.

자세한 내용은 ElastiCache API 참조의 PurchaseReservedCacheNodesOffering을 참조하십시오.

예약 노드에 대한 정보 가져오기

AWS Management 콘솔, AWS CLI 및 ElastiCache API를 사용하여 구입한 예약 노드에 대한 정보를 가져올 수 있습니다.

주제

- 예약 노드에 대한 정보 가져오기 (콘솔) (p. 389)
- 예약 노드에 대한 정보 가져오기 (AWS CLI) (p. 389)
- 예약 노드에 대한 정보 가져오기 (ElastiCache API) (p. 390)

예약 노드에 대한 정보 가져오기 (콘솔)

다음 절차에서는 AWS Management 콘솔을 사용하여 구입한 예약 노드에 대한 정보를 가져오는 방법을 설명합니다.

구입한 예약 노드에 대한 정보를 가져오려면

- 1. AWS Management 콘솔에 로그인하고 https://console.aws.amazon.com/elasticache/에서 ElastiCache 콘솔을 엽니다.
- 2. 탐색 목록에서 [Reserved Cache Nodes] 링크를 선택합니다.

계정의 예약 노드가 [Reserved Cache Nodes] 목록에 나타난다. 목록에서 예약 노드를 선택하면 콘솔 아 래쪽의 세부 정보 창에 예약 노드에 대한 자세한 정보가 표시됩니다.

예약 노드에 대한 정보 가져오기 (AWS CLI)

AWS 계정의 예약 노드에 대한 정보를 가져오려면 명령 프롬프트에 명령을 입력하십시오.

 $\hbox{aws elasticache describe-reserved-cache-nodes}\\$

이 작업은 다음과 비슷한 출력을 생성합니다(JSON 형식).

```
{
    "ReservedCacheNodeId": "myreservationid",
    "ReservedCacheNodesOfferingId": "649fd0c8-cf6d-47a0-bfa6-060f8e75e95f",
    "CacheNodeType": "cache.m1.small",
    "Duration": "31536000",
    "ProductDescription": "memcached",
```

```
"OfferingType": "Medium Utilization",
   "MaxRecords": 0
}
```

자세한 내용은 AWS CLI 참조의 describe--reserved-cache-nodes를 참조하십시오.

예약 노드에 대한 정보 가져오기 (ElastiCache API)

AWS 계정의 예약 노드에 대한 정보를 가져오려면 DescribeReservedCacheNodes 작업을 호출하십시오.

Example

```
https://elasticache.us-west-2.amazonaws.com/
    ?Action=DescribeReservedCacheNodes
    &Version=2014-12-01
    &SignatureVersion=4
    &SignatureMethod=HmacSHA256
    &Timestamp=20141201T220302Z
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Date=20141201T220302Z
    &X-Amz-DignedHeaders=Host
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Expires=20141201T220302Z
    &X-Amz-Signature=<signature>
```

이 호출은 다음과 비슷한 출력을 반환합니다.

```
<DescribeReservedCacheNodesResponse xmlns="http://elasticache.us-west-2.amazonaws.com/</pre>
doc/2013-06-15/">
 <DescribeReservedCacheNodesResult>
    <ReservedCacheNodes>
      <ReservedCacheNode>
        <OfferingType>Medium Utilization</OfferingType>
        <CurrencyCode>USD</CurrencyCode>
        <RecurringCharges/>
        <ProductDescription>memcached</productDescription>
        <ReservedCacheNodesOfferinqId>649fd0c8-cf6d-47a0-bfa6-060f8e75e95f/
ReservedCacheNodesOfferingId>
        <State>payment-failed</State>
        <ReservedCacheNodeId>myreservationid</ReservedCacheNodeId>
        <CacheNodeCount>1</CacheNodeCount>
        <StartTime>2010-12-15T00:25:14.131Z</StartTime>
        <Duration>31536000</Duration>
        <FixedPrice>227.5</FixedPrice>
        <UsagePrice>0.046</UsagePrice>
        <CacheNodeType>cache.m1.small</CacheNodeType>
      </ReservedCacheNode>
      <ReservedCacheNode>
      (...some output omitted for brevity...)
      </ReservedCacheNode>
    </ReservedCacheNodes>
  </DescribeReservedCacheNodesResult>
  <ResponseMetadata>
    <RequestId>23400d50-2978-11e1-9e6d-771388d6ed6b</RequestId>
 </ResponseMetadata>
</DescribeReservedCacheNodesResponse>
```

자세한 내용은 ElastiCache API 참조의 DescribeReservedCacheNodes를 참조하십시오.

Amazon ElastiCache 규정 준수 확인

타사 감사자는 여러 AWS 규정 준수 프로그램의 일환으로 Amazon ElastiCache의 보안 및 규정 준수를 평가합니다. 여기에는 SOC, PCI, FedRAMP, HIPAA 등이 포함됩니다.

규정 준수 프로그램 범위에 속하는 AWS 서비스의 목록은 규정 준수 프로그램 제공 범위 내 AWS 서비스 페이지에서 확인하십시오. 일반 정보는 AWS 규정 준수 프로그램을 참조하십시오.

AWS Artifact를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 AWS 아티팩트의 보고서 다운로드를 참조하십시오.

Amazon ElastiCache 사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률 및 규정에 따라 결정됩니다. Amazon ElastiCache 사용 시 HIPAA, PCI 또는 Fedrmap와 같은 표준을 준수해야 하는 경우 다음과 같은 AWS 도움말 리소스를 활용하십시오.

- 보안 및 규정 준수 빠른 시작 안내서 이 배포 가이드에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수에 중점을 둔 기본 AWS 환경을 배포하기 위한 단계를 제공합니다.
- HIPAA 보안 및 규정 준수 기술 백서 설계 이 백서는 기업에서 AWS를 사용하여 HIPAA를 준수하는 애플리케이션을 만드는 방법을 설명합니다.
- AWS 규정 준수 리소스 이 워크북 및 안내서는 귀사의 산업 및 위치에 적용될 수 있습니다.
- AWS Config 이 AWS 서비스로 리소스 구성이 내부 관행, 업계 지침 및 규정을 준수하는 정도를 평가할 수 있습니다.
- AWS Security Hub 이 AWS 서비스는 보안 업계 표준 및 모범 사례 준수 여부를 확인하는 데 도움이 되는 AWS 내 보안 상태에 대한 포괄적인 관점을 제공합니다.

주제

• Redis용 ElastiCache 규정 준수 (p. 391)

Redis용 ElastiCache 규정 준수

이 단원에서는 Redis용 Amazon ElastiCache 사용 시 제공되는 규정 준수 요구 사항 및 제어를 확인할 수 있습니다.

주제

- 규정 준수 관련 셀프 서비스 보안 업데이트 (p. 391)
- Redis용 ElastiCache FedRAMP 규정 준수 (p. 392)
- HIPAA 자격 획득 (p. 393)
- Redis용 ElastiCache PCI DSS 준수 (p. 394)
- 새 규정 준수 클러스터 생성 및 시드 (p. 394)
- 추가 정보 (p. 395)

규정 준수 관련 셀프 서비스 보안 업데이트

ElastiCache는 콘솔, API, CLI를 통해 서비스 업데이트라는 셀프 서비스 소프트웨어 업데이트 기능을 제공합니다. 이 기능을 사용하여 Redis 클러스터에서 온디맨드 및 실시간으로 보안 업데이트를 관리할 수 있습니다. 이 기능을 사용하면 최신 필수 보안 수정 사항으로 Redis 클러스터를 업데이트할 때 비즈니스에 미치는 영향이 최소화되도록 제어할 수 있습니다.

보안 업데이트는 서비스 업데이트 기능을 통해 릴리스됩니다. 보안 업데이트의 업데이트 유형 값으로 지정됩니다. 서비스 업데이트에는 심각도와 Recommended Apply by Date(날짜별 권장 적용) 필드가 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Redis용 ElastiCache 규정 준수

Redis 클러스터의 규정 준수를 유지하기 위해서는 Recommended Apply by Date(날짜별 권장 적용)로 이용 가능한 업데이트를 적용해야 합니다. SLA Met 필드에는 Redis 클러스터의 규정 준수 상태가 반영됩니다.

Note

권장 일자까지 서비스 업데이트를 적용하지 않거나 서비스 업데이트가 만료될 경우에도 ElastiCache가 대신 업데이트를 적용하지 않습니다.

Redis 콘솔, 이메일, Amazon SNS, CloudWatch 이벤트 및 Personal Health Dashboard 등의 공지를 통해 Redis 클러스터에 적용 가능한 서비스 업데이트 알림을 받을 수 있습니다. 셀프 서비스 유지 관리에 대한 자세한 내용은 Amazon ElastiCache의 셀프 서비스 업데이트 (p. 412) 단원을 참조하십시오.

다음 리전에서는 CloudWatch 이벤트와 Personal Health Dashboard가 지원되지 않습니다.

- us-gov-west-1
- · us-gov-east-1
- · cn-north-1
- · cn-northwest-1

Redis용 ElastiCache FedRAMP 규정 준수

AWS FedRAMP 규정 준수 프로그램에는 Redis용 Amazon ElastiCache가 FedRAMP 권한 부여 서비스로 포함되어 있습니다. 연방 또는 상업 고객인 경우 서비스를 사용하여 중요 워크로드를 중간 영향 수준 이하의 데이터로 AWS 미국 동부 및 미국 서부에서 처리하고 저장할 수 있습니다. 서비스를 사용하여 중요 워크로드를 AWS GovCloud(미국) 리전의 승인 범위를 높은 영향 수준 이하의 데이터로 처리할 수 있습니다.

FedRAMP PMO 또는 AWS 영업 팀 계정 관리자를 통해 AWS FedRAMP 보안 패키지 액세스를 요청할 수 있으며, AWS Artifact에서 AWS Artifact를 통해 다운로드할 수 있습니다.

요구 사항

Redis용 ElastiCache 클러스터에서 FedRAMP 지원을 가능하도록 하려면 클러스터와 클러스터 내 노드에서 다음 요구 사항을 충족해야 합니다.

- 엔진 버전 요구 사항 FedRAMP 규정 준수 요건에 부합하려면 활성화 및 비활성화된 클러스터 모드 모두에 대해 클러스터는 Redis용 ElastiCache 3.2.6. 4.0.10 이상을 실행해야 합니다.
 - 활성화 및 비활성화된 클러스터 모드 모두에 대한 Redis용 ElastiCache 버전 3.2.6, 4.0.10 이상부터, 다음과 같은 추가적인 데이터 보안 기능도 활성화할 수 있습니다.
 - Redis용 ElastiCache 전송 중 데이터 암호화(TLS) (p. 300)
 - Redis용 ElastiCache에서 미사용 데이터 암호화 (p. 306)
 - Redis AUTH 명령을 사용하여 사용자 인증 (p. 313)
- 노드 유형 요구 사항 클러스터에서 최신 노드 유형인 M4, M5, T2, R4 또는 R5를 실행해야 합니다. 자세한 내용은 다음 자료를 참조하십시오.
 - 지원되는 노드 유형 (p. 54)
 - 노드 크기 선택 (p. 67)
- FIPS 엔드포인트 요구 사항 Redis용 ElastiCache은 다음 리전에서 사용할 수 있는 FIPS 엔드포인트로 생성할 수 있습니다.

리전 이름/리전	FIPS 엔드포인트
미국 동부(오하이오) 리전	elasticache-fips.us-
us-east-2	east-2.amazonaws.com

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Redis용 ElastiCache 규정 준수

리전 이름/리전	FIPS 엔드포인트	
미국 동부(버지니아 북부) 지역	elasticache-fips.us-	
us-east-1	east-1.amazonaws.com	
미국 서부(캘리포니아 북부) 리전	elasticache-fips.us-	
us-west-1	west-1.amazonaws.com	
미국 서부(오레곤) 지역	elasticache-fips.us-	
us-west-2	west-2.amazonaws.com	
AWS GovCloud(미국 서부)	elasticache-fips.us-gov-	
us-gov-west-1	west-1.amazonaws.com	

• 보안 업데이트 요구 사항 – Recommended Apply by Date(날짜별 권장 적용)을 통해 Redis 클러스터를 정 기적으로 업데이트해야 합니다. 비즈니스에 영향이 없도록 실시간 및 온디맨드로도 클러스터를 업데이트 할 수 있습니다. 자세한 내용은 Amazon ElastiCache의 셀프 서비스 업데이트 (p. 412) 단원을 참조하십 시오.

HIPAA 자격 획득

AWS HIPAA 규정 준수 프로그램에는 Redis용 Amazon ElastiCache가 HIPAA 적격 서비스로 포함되어 있습니다.

HIPAA를 준수하여 Redis용 ElastiCache를 사용하려면 AWS와 비즈니스 제휴 계약(BAA)을 체결해야 합니다. 또한 클러스터 내의 클러스터 및 노드는 다음과 같은 엔진 버전, 노드 유형 및 데이터 보안 요구 사항을 충족해야 합니다.

요구 사항

Redis용 ElastiCache 클러스터에서 HIPAA 지원을 가능하도록 하려면 클러스터와 클러스터 내 노드에서 다음 요구 사항을 충족해야 합니다.

- 엔진 버전 요구 사항 HIPAA 자격을 갖추려면 클러스터에서 다음 Redis용 ElastiCache 버전 중 하나를 실행해야 합니다.
 - Redis용 ElastiCache 버전 5.0.0(개선됨) (p. 40)
 - Redis용 ElastiCache 버전 4.0.10(개선됨) (p. 41)
 - Redis용 ElastiCache 버전 3.2.6(개선됨) (p. 42)
- 노드 유형 요구 사항 클러스터에서 최신 노드 유형인 M4, M5, T2, R4 또는 R5를 실행해야 합니다. 자세한 내용은 다음을 참조하십시오.
 - 지원되는 노드 유형 (p. 54)
 - 노드 크기 선택 (p. 67)
- 데이터 보안 요구 사항 클러스터에서 전송 중 데이터 암호화, 미사용 데이터 암호화 및 Redis AUTH를 활성화해야 합니다. 자세한 내용은 다음을 참조하십시오.
 - Redis용 ElastiCache 전송 중 데이터 암호화(TLS) (p. 300)
 - Redis용 ElastiCache에서 미사용 데이터 암호화 (p. 306)
 - Redis AUTH 명령을 사용하여 사용자 인증 (p. 313)
- 보안 업데이트 요구 사항 Recommended Apply by Date(날짜별 권장 적용)을 통해 보안 유형의 최신 서비스 업데이트로 Redis 클러스터를 업데이트해야 합니다. 비즈니스에 영향이 없도록 실시간 및 온디맨드로도 클러스터를 업데이트할 수 있습니다. 자세한 내용은 Amazon ElastiCache의 셀프 서비스 업데이트 (p. 412) 단원을 참조하십시오.

이러한 요구 사항을 구현함으로써 Redis용 ElastiCache를 사용하여 HIPAA 규정을 준수하면서 개인 건강 정보(PHI)를 저장, 처리 및 액세스할 수 있습니다.

AWS 클라우드 및 HIPAA 자격 획득에 대한 일반적인 내용은 다음 문서를 참조하십시오.

- HIPAA 규정 준수
- Amazon Web Services에서 HIPAA 보안 및 규정 준수에 따라 아키텍처 설계
- 보안 업데이트 요구 사항 Recommended Apply by Date(날짜별 권장 적용)을 통해 Redis 클러스터를 정 기적으로 업데이트해야 합니다. 비즈니스에 영향이 없도록 실시간 및 온디맨드로도 클러스터를 업데이트 할 수 있습니다. 자세한 내용은 Amazon ElastiCache의 셀프 서비스 업데이트 (p. 412) 단원을 참조하십 시오.

Redis용 ElastiCache PCI DSS 준수

AWS PCI DSS 규정 준수 프로그램에는 Redis용 Amazon ElastiCache가 PCI 규정 준수 서비스로 포함되어 있습니다. PCI DSS 3.2 규정 준수 패키지는 AWS Artifact를 통해 다운로드할 수 있습니다. 자세한 내용은 AWS PCI DSS 규정 준수 프로그램을 참조하십시오.

요구 사항

Redis용 ElastiCache 클러스터에서 PCI DSS 지원을 가능하도록 하려면 클러스터와 클러스터 내 노드에서 다음 요구 사항을 충족해야 합니다.

- 엔진 버전 요구 사항 활성화 및 비활성화된 클러스터 모드 모두에 대해 클러스터는 Redis용 ElastiCache 3.2.6, 4.0.10 이상을 실행해야 합니다.
- 노드 유형 요구 사항 클러스터에서 최신 노드 유형인 M4, M5, T2, R4 또는 R5를 실행해야 합니다. 자세한 내용은 다음을 참조하십시오.
 - 지원되는 노드 유형 (p. 54)
 - 노드 크기 선택 (p. 67)
- 보안 업데이트 요구 사항 Recommended Apply by Date(날짜별 권장 적용)을 통해 Redis 클러스터를 정 기적으로 업데이트해야 합니다. 비즈니스에 영향이 없도록 실시간 및 온디맨드로도 클러스터를 업데이트 할 수 있습니다. 자세한 내용은 Amazon ElastiCache의 셀프 서비스 업데이트 (p. 412) 단원을 참조하십 시오.

Redis용 ElastiCache는 또한 데이터 보안 제어를 제공하여 서비스 사용 시 고객 카드 소지자 데이터(CHD)와 같은 민감한 재무 데이터를 저장, 처리 및 전송할 때 클러스터를 보호합니다.

데이터 보안 옵션 – 자세한 내용은 다음을 참조하십시오.

- Redis용 ElastiCache 전송 중 데이터 암호화(TLS) (p. 300)
- Redis용 ElastiCache에서 미사용 데이터 암호화 (p. 306)
- Redis AUTH 명령을 사용하여 사용자 인증 (p. 313)

새 규정 준수 클러스터 생성 및 시드

규정 준수 클러스터를 생성하려면 새 클러스터를 생성하여 선택 사항이 원하는 규정 준수에 대한 요구 사항을 충족하도록 합니다. 이러한 요구 사항에는 엔진 버전, 노드 유형, 암호화 및 필요한 경우 FIPS 엔드포인트가 포함될 수 있습니다. 선택한 경우 생성하고 있는 것과 같이 기존 클러스터의 데이터를 사용해 새 규정 준수 클러스터를 시드할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.

- 클러스터 생성 (p. 65)
- Redis 복제 그룹을 처음부터 새로 생성 (p. 148)
- 외부에서 생성된 백업으로 새 클러스터 시드 (p. 211)

추가 정보

AWS 클라우드 규정 준수에 대한 일반적인 내용은 다음 문서를 참조하십시오.

- 규정 준수 관련 셀프 서비스 보안 업데이트 (p. 391)
- Amazon ElastiCache의 셀프 서비스 업데이트 (p. 412)
- AWS 클라우드 규정 준수
- 공동 책임 모델
- 규정 준수 프로그램 제공 AWS 범위 내 서비스
- AWS HIPAA 규정 준수 프로그램
- · Architecting for HIPAA Security and Compliance on Amazon Web Services
- AWS PCI DSS 규정 준수 프로그램

Amazon ElastiCache의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전에서는 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 지연 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크에 연결되어 있습니다. 가용 영역을 사용하면 중단 없이 가용 영역 간에 자동으로 장애 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 내용은 AWS 글로벌 인프라를 참조하십시오.

AWS 글로벌 인프라 외에 Amazon ElastiCache도 데이터 복원성과 백업 요구 사항을 지원하는 다양한 기능을 제공합니다.

주제

• 장애 완화 (p. 395)

장애 완화

Amazon ElastiCache 구현을 계획할 때는 장애가 애플리케이션 및 데이터에 미치는 영향을 최소화하도록 계획해야 합니다. 이 단원의 항목은 애플리케이션 및 데이터를 장애로부터 보호하기 위해 취할 수 있는 접근 방식을 다룹니다.

주제

- Redis 실행 시 장애 완화 (p. 395)
- 권장 사항 (p. 397)

Redis 실행 시 장애 완화

Redis 엔진을 실행할 때 노드 또는 가용 영역 장애의 영향을 최소화하기 위한 다음과 같은 옵션이 있습니다.

노드 장애 완화

Redis 노드 장애의 영향을 완화하기 위해 다음과 같은 옵션이 있습니다.

주제

- 장애 완화: Redis AOF(Append Only File) (p. 396)
- 장애 완화: Redis 복제 그룹 (p. 396)

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 장애 완화

장애 완화: Redis AOF(Append Only File)

Redis에 대해 AOF가 활성화되면 Redis 클러스터에 데이터가 작성될 때마다 해당하는 트랜잭션 레코드도 Redis AOF(Append Only File)에 작성됩니다. Redis 프로세스가 재시작되면 ElastiCache는 대체 클러스터를 생성하고 프로비저닝합니다. 그런 다음 클러스터에 대해 AOF를 실행하여 데이터로 다시 채울 수 있습니다.

클러스터 장애를 완화하기 위해 Redis AOF를 사용하면 다음과 같은 일부 단점이 있습니다.

• 시간이 많이 걸립니다.

클러스터를 생성 및 프로비저닝하는 데는 몇 분 정도가 걸릴 수 있습니다. AOF 크기에 따라, 클러스터에 대해 실행하는 경우 애플리케이션이 데이터를 위해 클러스터에 액세스할 수 없을 때 더 많은 시간이 추가됩니다. 이렇게 하면 애플리케이션 데이터베이스에 직접 접촉하도록 유도됩니다.

• AOF가 커질 수 있습니다.

클러스터에 대한 모든 쓰기는 트랜잭션 레코드에 작성되므로 AOF가 매우 커질 수 있으며, 해당 데이터세트에 대한 .rdb 파일보다 더 커질 수 있습니다. ElastiCache는 크기가 제한된 로컬 인스턴스 스토어에 의존하므로, AOF를 활성화하면 디스크 공간 부족 문제가 발생할 수 있습니다. 다중 AZ가 활성화된 복제 그룹을 사용하여 디스크 공간 부족 문제를 방지할 수 있습니다.

• AOF의 사용이 모든 장애 시나리오로부터 보호할 수 있는 것은 아닙니다.

예를 들어, 기본 물리적 서버에서 하드웨어 결함이 생겨 노드가 실패하면 ElastiCache가 다른 서버에 새로운 노드를 프로비저닝합니다. 이 경우 AOF를 사용할 수 없으며, 이를 사용하여 데이터를 복구할 수 없습니다.

자세한 내용은 Redis용 ElastiCache의 AOF(Append Only Files) (p. 218) 단원을 참조하십시오.

장애 완화: Redis 복제 그룹

Redis 복제 그룹은 애플리케이션이 읽고 쓸 수 있는 단일 기본 노드와 1~5개의 읽기 전용 복제본 노드로 구성되어 있습니다. 기본 노드에 데이터가 작성될 때마다 읽기 전용 복제본 노드에서도 비동기식으로 업데이트됩니다.

읽기 전용 복제본 장애의 경우

- 1. ElastiCache는 장애가 있는 읽기 전용 복제본을 감지합니다.
- 2. ElastiCache는 장애가 있는 노드를 오프라인 상태로 전환합니다.
- 3. ElastiCache는 동일한 AZ에서 대체 노드를 시작하고 프로비저닝합니다.
- 4. 새 노드가 기본 노드와 동기화됩니다.

이 기간 동안 애플리케이션에서는 다른 노드를 사용하여 계속 읽고 쓸 수 있습니다.

자동 장애 조치가 있는 Redis 다중 AZ

Redis 복제 그룹에서 자동 장애 조치가 있는 다중 AZ를 활성화할 수 있습니다. 자동 장애 조치가 있는 다중 AZ를 활성화했는지 여부와 상관없이 장애가 있는 기본 노드가 자동으로 감지되고 대체됩니다. 다중 AZ가 활성화되었는지 여부에 따라 이러한 작동 방식이 달라집니다.

자동 장애 조치가 있는 다중 AZ가 활성화된 경우

- 1. ElastiCache가 기본 노드 장애를 감지합니다.
- 2. ElastiCache가 복제 지연 시간이 가장 짧은 읽기 전용 복제본 노드를 기본 노드로 승격시킵니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 장애 완화

- 3. 다른 복제본이 새 기본 노드와 동기화됩니다.
- 4. ElastiCache가 장애가 있는 기본 노드의 AZ에서 읽기 전용 복제본을 실행합니다.
- 5. 새 노드가 새로 승격된 기본 노드와 동기화됩니다.

복제본 노드에 장애 조치하는 것은 일반적으로 새 기본 노드를 생성하고 프로비저닝하는 것보다 빠릅니다. 즉, 애플리케이션에서는 다중 AZ가 활성화되지 않은 경우보다 더 빠르게 기본 노드에 대한 쓰기를 재개할 수 있습니다.

자세한 내용은 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132) 단원을 참조하십시오.

자동 장애 조치가 있는 다중 AZ가 비활성화된 경우

- 1. ElastiCache가 기본 장애를 감지합니다.
- 2. ElastiCache가 기본 오프라인 상태로 전환합니다.
- 3. ElastiCache가 새 기본 노드를 생성하고 프로비저닝하여 장애가 있는 기본 노드를 대체합니다.
- 4. ElastiCache가 새 기본 노드를 기존 복제본 중 하나와 동기화합니다.
- 5. 동기화가 완료되면 새 노드가 클러스터의 기본 노드로 작동합니다.

이 프로세스 중 1단계에서 4단계까지 애플리케이션에서는 기본 노드에 쓸 수 없습니다. 그러나 애플리케이션에서는 복제본 노드에서 계속 읽을 수 있습니다.

추가 보호를 위해 서로 다른 AZ(가용 영역)의 복제 그룹에서 노드를 시작하는 것이 좋습니다. 이렇게 할 경우 AZ 장애는 해당 AZ의 노드에만 영향을 주며 다른 노드에는 영향을 주지 않습니다.

자세한 내용은 고가용성을 위한 복제 그룹 사용 (p. 127) 단원을 참조하십시오.

가용 영역 장애 완화

가용 영역 장애의 영향을 완화하려면 가능한 한 많은 가용 영역에 노드를 배치합니다.

보유한 노드의 수와 상관없이 동일한 가용 영역에 노드가 모두 배치된 경우, 해당 AZ에 치명적인 장애가 발생하면 모든 캐시 데이터가 손실됩니다. 그러나 여러 AZ에 노드를 배치할 경우 AZ에 장애가 발생하면 해당 AZ에 있는 노드만 손실됩니다.

읽기 작업이 이제 더 적은 수의 노드에서 공유되므로 노드가 손실될 때마다 성능 저하를 경험할 수 있습니다. 노드가 대체될 때까지 이 성능 저하가 계속됩니다. Redis 노드에 데이터가 분할되지 않으므로 기본 노드가 손실될 경우에만 일부 데이터 손실이 발생할 위험이 있습니다.

Redis 노드의 가용 영역 지정에 대한 자세한 내용은 Redis(클러스터 모드 비활성화됨) 클러스터 생성 (콘솔) (p. 69) 단원을 참조하십시오.

리전 및 가용 영역에 대한 자세한 내용은 리전 및 가용 영역 선택 (p. 47) 단원을 참조하십시오.

권장 사항

계획해야 할 장애의 유형에는 개별 노드 장애와 광범위한 가용 영역 장애의 두 유형이 있습니다. 가장 좋은 장애 완화 계획은 두 유형의 장애를 모두 해결하는 것입니다.

장애의 영향 최소화

노드 장애의 영향을 최소화하려면 구현 시 각 샤드에서 여러 노드를 사용하고 여러 가용 영역에 노드를 분산 하는 것이 좋습니다.

Redis를 실행할 때는 기본 노드에 장애가 발생하는 경우 ElastiCache가 복제본으로 자동으로 장애 조치하도록 복제 그룹에 대해 다중 AZ를 활성화하는 것이 좋습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 인프라 보안

가용 영역 장애의 영향 최소화

가용 영역 장애의 영향을 최소화하려면 가능한 한 여러 개의 서로 다른 가용 영역에서 노드를 시작하는 것이 좋습니다. AZ에 노드를 균등하게 분산하면, 드물지만 AZ 장애가 발생할 경우 영향을 최소화합니다.

기타 주의 사항

Redis를 실행할 경우, 위 사항 이외에 클러스터의 정기적 백업을 예약하는 것이 좋습니다. 백업(스냅샷)은 장애 또는 손상이 발생할 경우 클러스터를 복원하는 데 사용할 수 있는 .rdb 파일을 생성합니다. 자세한 내용은 Redis용 ElastiCache 백업 및 복원 (p. 187) 단원을 참조하십시오.

AWS Elasticache의 인프라 보안

관리형 서비스인 AWS Elasticache는 Amazon Web Services: 보안 프로세스 개요 백서에 설명된 AWS 글로 벌 네트워크 보안 절차로 보호됩니다.

AWS에서 게시한 API 호출을 사용하여 네트워크를 통해 Elasticache에 액세스합니다. 클라이언트가 TLS(전송 계층 보안) 1.0 이상을 지원해야 합니다. TLS 1.2 이상을 권장합니다. 클라이언트는 Ephemeral Diffie-Hellman(DHE) 또는 Elliptic Curve Ephemeral Diffie-Hellman(ECDHE)과 같은 PFS(전달 완전 보안, Perfect Forward Secrecy)가 포함된 암호 제품군도 지원해야 합니다. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 관련된 보안 액세스 키를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

캐싱 전략 및 모범 사례

다음은 Amazon ElastiCache에 대한 권장 모범 사례입니다. 다음 모범 사례를 준수하면 클러스터의 성능과 안정성이 향상됩니다.

주제

- 캐싱 전략 (p. 399)
- 제한된 Redis 명령 (p. 403)
- 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)
- 예약된 메모리 관리 (p. 405)
- Redis AOF를 사용할 때 디스크 공간 부족 문제 완화 (p. 409)
- 모범 사례: 온라인 클러스터 크기 조정 (p. 410)
- 모범 사례: 유지 관리 중 가동 중지 최소화 (p. 411)

캐싱 전략

다음 항목에서는 캐시를 채우고 유지 관리하기 위한 전략을 확인할 수 있습니다.

캐시를 채우고 유지 관리하기 위해 구현하려는 전략은 캐싱되는 데이터의 유형과 해당 데이터에 대한 액세스 패턴에 따라 달라집니다. 예를 들어, 게임 사이트와 새 이야기 추세의 상위 10개 리더보드에 대해 동일한 전 략을 사용하려고 하지 않을 수 있습니다. 이 단원의 나머지 부분에서는 일반적인 캐시 유지 관리 전략, 이에 대한 장점 및 단점에 대해 살펴봅니다.

주제

- 지연 로딩 (p. 399)
- 라이트-스루 (p. 401)
- TTL 추가 (p. 402)
- 관련 주제 (p. 403)

지연 로딩

이름에서 알 수 있듯이 지연 로딩은 필요할 때에만 데이터를 캐시에 로드하는 캐싱 전략입니다. 다음 설명과 같이 작동합니다.

Amazon ElastiCache는 액세스하는 애플리케이션과 데이터 스토어(데이터베이스) 사이에 위치하는 인 메모리 키/값 저장소입니다. 애플리케이션에서 데이터를 요청할 때마다 ElastiCache 캐시에 먼저 요청합니다. 데이터가 캐시에 있으며 최신 상태인 경우 ElastiCache가 데이터를 애플리케이션에 반환합니다. 데이터가 캐시에 없거나 만료된 경우 애플리케이션에서 데이터 스토어에 데이터를 요청합니다. 데이터 스토어에서 데이터를 애플리케이션에 반환합니다. 다음으로 애플리케이션은 스토어에서 수신한 데이터를 캐시에 씁니다. 이렇게 하면 다음에 요청이 있을 때 데이터를 더 빨리 검색할 수 있습니다.

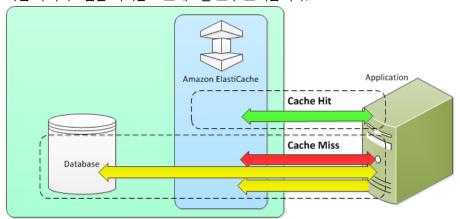
캐시 적중률은 데이터가 캐시에 있고 만료되지 않은 경우에 발생합니다.

- 1. 애플리케이션은 캐시에서 데이터를 요청합니다.
- 2. 캐시는 애플리케이션으로 데이터를 반환합니다.

캐시 누락은 데이터가 캐시에 없거나 만료된 경우에 발생합니다.

- 1. 애플리케이션은 캐시에서 데이터를 요청합니다.
- 2. 캐시에 데이터가 요청되지 않으므로 null을 반환합니다.
- 3. 애플리케이션은 데이터베이스에 데이터를 요청하고 수신합니다.
- 4. 애플리케이션은 새 데이터로 캐시를 업데이트합니다.

다음 다이어그램은 이러한 프로세스를 모두 보여줍니다.



지연 로딩의 장점 및 단점

지연 로딩의 장점은 다음과 같습니다.

• 요청된 데이터만 캐싱됩니다.

대부분의 데이터가 요청되지 않으므로 지연 로딩은 요청되지 않은 데이터가 있는 캐시를 채우지 않습니다.

• 노드 장애가 애플리케이션에 치명적인 영향을 주지 않습니다.

노드 장애가 발생하여 새로운 빈 노드로 대체될 경우 애플리케이션에서는 지연 시간 증가를 통해 계속 작동합니다. 새 노드에 대한 요청이 발생하면, 각 캐시가 누락될 때마다 데이터베이스 쿼리가 생성됩니다. 동시에 데이터 복사본이 캐시에 추가되어 이후의 요청이 캐시에서 검색됩니다.

지연 로딩의 단점은 다음과 같습니다.

- 캐시 누락 패널티가 있습니다. 각 캐시 누락은 세 개의 이동으로 나타납니다.
 - 1. 캐시에서 데이터에 대한 초기 요청
 - 2. 데이터에 대한 데이터베이스의 쿼리
 - 3. 캐시에 데이터 작성

이러한 누락으로 인해 애플리케이션으로 데이터를 가져오는 것이 눈에 띄게 지연될 수 있습니다.

• 기한 경과된 데이터

캐시 누락 시에만 데이터를 캐시에 쓰면, 캐시의 데이터가 기한이 경과할 수 있습니다. 이 결과는 데이터베이스에서 데이터가 변경될 때 캐시에 대한 업데이트가 없기 때문에 발생합니다. 이 문제를 해결하려면 라이트-스루 (p. 401) 및 TTL 추가 (p. 402) 전략을 사용할 수 있습니다.

지연 로딩 의사(pseudo) 예제

다음은 지연 로딩 로직의 의사(pseudo) 코드 예제입니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 라이트-스루

```
// ********************
// function that returns a customer's record.
// Attempts to retrieve the record from the cache.
// If it is retrieved, the record is returned to the application.
// If the record is not retrieved from the cache, it is
     retrieved from the database,
     added to the cache, and
     returned to the application
//
// *******
get_customer(customer_id)
   customer record = cache.get(customer id)
   if (customer_record == null)
       customer_record = db.query("SELECT * FROM Customers WHERE id == {0}", customer_id)
       cache.set(customer_id, customer_record)
   return customer_record
```

이 예제의 경우, 데이터를 갖는 애플리케이션 코드는 다음과 같습니다.

```
customer_record = get_customer(12345)
```

라이트-스루

라이트-스루 전략은 데이터베이스에 데이터를 작성할 때마다 데이터를 추가하거나 캐시의 데이터를 업데이 트합니다.

라이트-스루의 장점 및 단점

라이트-스루의 장점은 다음과 같습니다.

• 캐시의 데이터가 기한 경과되지 않습니다.

캐시의 데이터는 데이터베이스에 쓰일 때마다 업데이트되므로 항상 최신 상태입니다.

• 쓰기 패널티 vs. 읽기 패널티

모든 쓰기에는 다음 2개의 이동이 수반됩니다.

- 1. 캐시에 쓰기
- 2. 데이터베이스에 쓰기

이로 인해 프로세스에 지연 시간이 추가됩니다. 즉, 최종 사용자는 일반적으로 데이터를 검색할 때보다 데이터를 업데이트할 때 지연 시간에 더 관대합니다. 업데이트는 더 많이 작동하므로 오래 걸릴 수 있다는 고유한 생각이 있습니다.

라이트-스루의 단점은 다음과 같습니다.

• 누락된 데이터

노드 장애 또는 확장으로 인해 새 노드를 스핀업하면 데이터가 누락됩니다. 이 데이터는 데이터베이스에 추가되거나 업데이트될 때까지 계속 누락됩니다. 라이트-스루로 지연 로딩 (p. 399)을 구현하여 이 문제를 최소화할 수 있습니다.

• 캐시 이탈

대부분의 데이터는 절대 읽히지 않으며 이는 리소스 낭비입니다. TTL(Time to Live) 값을 추가 (p. 402)하면 낭비되는 공간을 최소화할 수 있습니다.

라이트-스루 의사(pseudo) 예제

다음은 라이트-스루 로직의 의사(pseudo) 코드 예제입니다.

```
// *****************************
// function that saves a customer's record.
// *******************************
save_customer(customer_id, values)

customer_record = db.query("UPDATE Customers WHERE id = {0}", customer_id, values)
cache.set(customer_id, customer_record)
return success
```

이 예제의 경우, 데이터를 갖는 애플리케이션 코드는 다음과 같습니다.

```
save_customer(12345,{"address":"123 Main"})
```

TTL 추가

지연 로딩은 기한 경과 데이터에 대해 허용되지만 빈 노드로 인해 실패하지 않습니다. 라이트-스루는 데이터를 항상 최신 상태로 유지하지만, 빈 노드로 인해 실패할 수 있으며 불필요한 데이터로 캐시를 채울 수 있습니다. 각 쓰기에 TTL(Time to Live) 값을 추가하면 각 전략의 이점을 얻을 수 있습니다. 동시에 추가 데이터로 캐시를 복잡하게 만들지 않을 수 있습니다.

TTL(Time To Live)은 키가 만료될 때까지의 시간(초)을 지정하는 정수 값입니다. Redis는 이 값에 대해 초 또는 밀리초를 지정할 수 있습니다. 애플리케이션에서 만료된 키를 읽으려고 하면 키가 없는 것으로 처리됩니다. 데이터베이스가 키에 대해 쿼리되고 캐시가 업데이트됩니다. 이는 값이 기한 경과가 아님을 보장하지 않습니다. 그러나 데이터가 너무 기간 경과되지 않도록 방지하며, 경우에 따라 캐시의 값이 데이터베이스에서 새로 고침되어야 합니다.

자세한 내용은 Redis set 명령 또는.

TTL 의사(pseudo) 예제

다음 코드는 TTL을 통한 라이트-스루 로직의 의사(pseudo) 코드 예제입니다.

다음은 TTL을 통한 지연 로딩 로직의 의사(pseudo) 코드 예제입니다.

```
// ****************************
// function that returns a customer's record.
// Attempts to retrieve the record from the cache.
// If it is retrieved, the record is returned to the application.
// If the record is not retrieved from the cache, it is
// retrieved from the database,
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 관련 주제

```
added to the cache, and
//
    returned to the application.
// The TTL value of 300 means that the record expires
// 300 seconds (5 minutes) after the set command
     and subsequent reads will have to query the database.
// **************
get_customer(customer_id)
   customer_record = cache.get(customer_id)
   if (customer_record != null)
       if (customer_record.TTL < 300)</pre>
           return customer_record
                                        // return the record and exit function
   // do this only if the record did not exist in the cache OR
       the TTL was >= 300, i.e., the record in the cache had expired.
   customer_record = db.query("SELECT * FROM Customers WHERE id = {0}", customer_id)
   cache.set(customer_id, customer_record, 300) // update the cache
   return customer record
                                        // return the newly retrieved record and exit
 function
```

이 예제의 경우, 데이터를 갖는 애플리케이션 코드는 다음과 같습니다.

```
save_customer(12345,{"address":"123 Main"})
```

```
customer_record = get_customer(12345)
```

관련 주제

- 인 메모리 데이터 스토어 (p. 2)
- 엔진 및 버전 선택
- Redis용 ElastiCache 클러스터 조정 (p. 219)

제한된 Redis 명령

관리형 서비스 환경을 제공하기 위해 ElastiCache는 고급 권한이 필요한 특정 캐시 엔진별 명령에 대한 액세 스를 제한합니다. Redis를 실행하는 캐시 클러스터에서는 다음 명령을 사용할 수 없습니다.

- bgrewriteaof
- bgsave
- · cluster addslot
- cluster delslot
- · cluster setslot
- · config
- debug
- migrate
- replicaof
- save
- slaveof
- shutdown
- sync

충분한 메모리를 확보하여 Redis 스냅샷 생성

버전 2.8.22 이상의 Redis 스냅샷 및 동기화

Redis 2.8.22에는 동기화 및 저장을 수행하는 동안 스왑 사용량 증가 없이 애플리케이션 사용에 추가 메모리를 할당할 수 있는 forkless 저장 프로세스가 도입되었습니다. 자세한 내용은 동기화 및 백업 구현 방법 (p. 143) 단원을 참조하십시오.

버전 2.8.22 이전의 Redis 스냅샷 및 동기화

Redis ElastiCache를 사용하여 작업할 때 다음과 같이 많은 경우에 Redis가 백그라운드 쓰기 명령을 호출합니다.

- 백업을 위해 스냅샷을 생성하는 경우
- 복제 그룹에서 기본을 사용하여 복제본을 동기화하는 경우
- Redis에 AOF(append-only file) 기능을 활성화하는 경우
- 복제본을 마스터로 승격하는 경우(기본/복제본 동기화 발생)

Redis에서 백그라운드 쓰기 프로세스를 실행할 때마다 사용 가능한 메모리가 충분해야 프로세스 오버헤드를 감당할 수 있습니다. 사용 가능한 메모리를 충분히 확보하지 못하면 프로세스가 실패합니다. 그러므로 Redis 클러스터를 생성할 때 메모리가 충분한 노드 인스턴스 유형을 선택해야 합니다.

백그라운드 쓰기 프로세스 및 메모리 사용량

백그라운드 쓰기 프로세스가 호출될 때마다 Redis에서 해당 프로세스를 포크합니다. Redis는 단일 스레드입니다. 포크 하나가 Redis .rdb 스냅샷 파일에서 데이터를 디스크에 계속 쓰고 나머지 포크가 모든 읽기 및 쓰기 작업을 처리합니다. 스냅샷이 특정 시점 스냅샷이 되도록 모든 데이터 업데이트 및 추가가 데이터 영역과 별도의 사용 가능한 메모리 영역에 기록됩니다.

사용 가능한 메모리가 충분하여 데이터가 계속 디스크에 쓰여지는 동안 모든 쓰기 작업을 기록할 수 있으면 메모리 부족 문제가 발생하지 않습니다. 다음과 같은 경우에 해당되며 메모리 부족 문제가 생기기 쉽습니다.

- 애플리케이션이 여러 쓰기 작업을 수행하여 새로운 데이터나 업데이트된 데이터를 저장하기 위해 사용 가능한 메모리가 대량으로 필요합니다.
- 새로운 데이터나 업데이트된 데이터를 쓸 사용 가능한 메모리가 거의 없습니다.
- 디스크에 계속 쓰기 위해 시간이 오래 걸리는 큰 데이터 세트가 있어 많은 쓰기 작업이 필요합니다.

다음 다이어그램에서는 백그라운드 쓰기 프로세스를 실행할 때의 메모리 사용량을 보여줍니다.

Memory use prior to a snapshot

Memory for data	Reserved/available memory				
Memory use during a snapshot—sufficient memory					
Memory for data	Memory used by background write operations Avail. memory		Additional memory needed for background		
Memory use during a snapshot—insufficien	write operations				
Memory for data	Memory used by background write operations				

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 백그라운드 쓰기를 실행할 때 메모리 부족 방지

백업이 성능에 미치는 영향에 대한 정보는 백업의 성능 영향 (p. 189) 단원을 참조하십시오.

Redis에서 스냅샷을 수행하는 방법에 대한 자세한 내용은 http://redis.io를 참조하십시오.

리전 및 가용 영역에 대한 자세한 내용은 리전 및 가용 영역 선택 (p. 47) 단원을 참조하십시오.

백그라운드 쓰기를 실행할 때 메모리 부족 방지

BGSAVE 또는 BGREWRITEAOF와 같은 백그라운드 쓰기 프로세스를 호출할 때마다 프로세스가 실패하지 않도록 하려면 프로세스 중에 쓰기 작업에 사용되는 것보다 많은 메모리가 있어야 합니다. 최악의 시나리오는 백그라운드 쓰기 작업 중에 모든 Redis 레코드가 업데이트되고 일부 새로운 레코드가 캐시에 추가되는 것입니다. 따라서 2.8.22 이전의 Redis 버전에서는 reserved-memory-percent를 50(50%)으로 설정하고 Redis 버전 2.8.22 이상에서는 25(25%)로 설정하는 것이 좋습니다.

maxmemory 값은 데이터 및 작업 오버헤드에 사용할 수 있는 메모리를 나타냅니다. 기본 파라미터 그룹에서 reserved-memory 파라미터를 수정할 수 없으므로 클러스터의 사용자 지정 파라미터 그룹을 생성해야 합니다. reserved-memory의 기본값은 0이며 Redis가 데이터에 모든 maxmemory를 소비할 수 있어 백그라운드 쓰기 프로세스와 같은 다른 용도로 사용할 수 있는 메모리가 거의 없을 수 있음을 의미합니다. 노드 인스턴스 유형에 따른 maxmemory 값은 Redis 노드 유형별 파라미터 (p. 295) 단원을 참조하십시오.

reserved-memory 파라미터를 사용하여 상자에서 Redis가 사용하는 메모리 양을 줄일 수도 있습니다.

ElastiCache의 Redis 관련 파라미터에 대한 자세한 내용은 Redis 특정 파라미터 (p. 274) 단원을 참조하십시 오.

파라미터 그룹 생성 및 수정에 대한 정보는 파라미터 그룹 생성 (p. 262) 및 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

예약된 메모리 관리

예약된 메모리는 비데이터 사용을 위해 구분된 메모리입니다. 백업 또는 장애 조치를 수행할 때 클러스터의 데이터가 .rdb 파일에 작성되는 동안 Redis에서는 사용 가능한 메모리를 사용하여 클러스터에 쓰기 작업을 기록합니다. 모든 쓰기에 사용 가능한 메모리를 충분히 확보하지 못하면 프로세스가 실패합니다. 아래에서 Redis용 ElastiCache에 대한 예약된 메모리를 관리하기 위한 옵션과 이러한 옵션을 적용하는 방법에 대한 정보를 찾을 수 있습니다.

주제

- 필요한 예약된 메모리의 양 (p. 405)
- 예약된 메모리를 관리하기 위한 파라미터 (p. 406)
- 예약된 메모리 관리 파라미터 지정 (p. 408)

필요한 예약된 메모리의 양

Redis 2.8.22 이전 버전을 실행하고 있는 경우 Redis 2.8.22 이후 버전을 실행하는 것보다 백업 및 장애 조치를 위해 더 많은 메모리를 예약해야 합니다. 이러한 요구 사항은 Redis용 ElastiCache가 백업 프로세스를 구현하는 방법이 다르기 때문입니다. 경험상, 2.8.22 이전 버전의 Redis 오버헤드에 대해서는 노드 유형 maxmemory 값의 절반을 예약하고 Redis 버전 2.8.22 이후에 대해서는 1/4을 예약합니다.

자세한 내용은 다음 자료를 참조하십시오.

- 충분한 메모리를 확보하여 Redis 스냅샷 생성 (p. 404)
- 동기화 및 백업 구현 방법 (p. 143)

예약된 메모리를 관리하기 위한 파라미터

2017년 3월 16일부터 Redis용 Amazon ElastiCache에서는 Redis 메모리 관리를 위해 상호 배타적인 파라미터 두 개, 즉 reserved-memory와 reserved-memory-percent를 제공합니다. 이 두 파라미터는 Redis배포의 일부가 아닙니다.

ElastiCache 고객이 된 시점에 따라, 이러한 파라미터 중 하나 또는 다른 파라미터가 기본 메모리 관리 파라미터입니다. 새 Redis 클러스터 또는 복제 그룹을 생성하고 기본 파라미터 그룹을 사용할 때 이 파라미터가적용됩니다.

- 2017년 3월 16일 이전에 시작한 고객의 경우 기본 파라미터 그룹을 사용하여 Redis 클러스터 또는 복제 그룹을 생성할 때 메모리 관리 파라미터가 reserved-memory입니다. 이 경우 0바이트의 메모리가 예약 됩니다.
- 2017년 3월 16일 이후에 시작된 고객의 경우 기본 파라미터 그룹을 사용하여 Redis 클러스터 또는 복제 그룹을 생성할 때 메모리 관리 파라미터는 reserved-memory-percent입니다. 이 경우 노드 maxmemory 값의 25%가 비데이터 용도로 예약됩니다.

두 개의 Redis 메모리 관리 파라미터를 읽은 후에는 기본이 아니거나 기본이 아닌 값을 가진 파라미터를 사용하는 것이 좋습니다. 이 경우, 다른 예약된 메모리 관리 파라미터로 변경할 수 있습니다.

이 파라미터의 값을 변경하려면 사용자 정의 파라미터 그룹을 생성하고 기본 설정 메모리 관리 파라미터 및 값을 사용하도록 수정할 수 있습니다. 그다음부터는 Redis 클러스터나 복제 그룹을 새로 생성할 때마다 이 사용자 지정 파라미터 그룹을 사용할 수 있습니다. 기존 클러스터나 복제 그룹의 경우 사용자 지정 파라미터 그룹을 사용하도록 수정할 수 있습니다.

자세한 내용은 다음 자료를 참조하십시오.

- 예약된 메모리 관리 파라미터 지정 (p. 408)
- 파라미터 그룹 생성 (p. 262)
- 파라미터 그룹 수정 (p. 271)
- ElastiCache 클러스터 수정 (p. 87)
- 복제 그룹 수정 (p. 165)

reserved-memory 파라미터

2017년 3월 16일 이전에는 모든 Redis용 ElastiCache 예약된 메모리 관리가 reserved-memory 파라미터 를 사용하여 수행되었습니다. reserved-memory의 기본값은 0입니다. 이 기본값은 Redis 오버헤드에 대해 메모리를 예약하지 않으며, Redis가 모든 노드의 메모리를 데이터로 사용하는 것을 허용합니다.

백업 및 장애 조치를 위해 충분한 메모리를 사용할 수 있도록 reserved-memory를 변경할 경우 사용자 지정 파라미터 그룹을 생성해야 합니다. 이 사용자 지정 파라미터 그룹에서 reserved-memory를 클러스터에 대해 실행 중인 Redis 버전과 클러스터의 노드 유형에 적합한 값으로 설정합니다. 자세한 내용은 필요한 예약된 메모리의 양 (p. 405) 단원을 참조하십시오.

Redis용 ElastiCache 파라미터 reserved-memory는 Redis용 ElastiCache에 고유하며, Redis 배포의 일부가 아닙니다.

다음 절차는 reserved-memory를 사용하여 Redis 클러스터의 메모리를 관리하는 방법을 보여줍니다.

reserved-memory를 사용하여 메모리를 예약하려면

1. 실행 중인 엔진 버전과 일치하는 파라미터 그룹 패밀리를 지정하는 사용자 지정 파라미터 그룹을 생성합니다. 예를 들어, redis2.8 파라미터 그룹 패밀리를 지정합니다. 자세한 내용은 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

aws elasticache create-cache-parameter-group \

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 예약된 메모리를 관리하기 위한 파라미터

```
--cache-parameter-group-name redis28-m3xl \
--description "Redis 2.8.x for m3.xlarge node type" \
--cache-parameter-group-family redis2.8
```

- 2. Redis 오버헤드를 위해 예약해야 하는 메모리의 바이트 수를 계산합니다. Redis 노드 유형별 파라미터 (p. 295)에서 노드 유형에 대한 maxmemory 값을 찾을 수 있습니다.
- 3. reserved-memory 파라미터가 이전 단계에서 계산된 바이트 수가 되도록 사용자 지정 파라미터 그룹을 수정합니다. 다음 AWS CLI 예제에서는 2.8.22 이전 버전의 Redis를 실행하고 있으며 노드의 maxmemory 절반을 예약한다고 가정합니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

```
aws elasticache modify-cache-parameter-group \
    --cache-parameter-group-name redis28-m3xl \
    --parameter-name-values "ParameterName=reserved-memory, ParameterValue=7130316800"
```

각 노드 유형에는 다른 maxmemory 값이 있으므로 사용할 각 노드 유형에 대해 별도의 사용자 지정 파라 미터 그룹이 필요합니다. 따라서 각 노드 유형에는 reserved-memory에 대해 서로 다른 값이 필요합니다.

4. Redis 클러스터나 복제 그룹을 수정하여 사용자 지정 파라미터 그룹을 사용하도록 합니다.

다음 CLI 예제에서는 my-redis-cluster 클러스터를 수정하여 즉시 시작되는 사용자 지정 파라미터 그룹 redis28-m3x1을 사용하도록 합니다. 자세한 내용은 ElastiCache 클러스터 수정 (p. 87) 단원을 참조하십시오.

```
aws elasticache modify-cache-cluster \
    --cache-cluster-id my-redis-cluster \
    --cache-parameter-group-name redis28-m3xl \
    --apply-immediately
```

다음 CLI 예제에서는 복제 그룹 my-redis-repl-grp를 수정하여 즉시 시작되는 사용자 지정 파라미터 그룹 redis28-m3x1을 사용하도록 합니다. 자세한 내용은 복제 그룹 수정 (p. 165) 단원을 참조하십시오.

```
aws elasticache modify-replication-group \
    --replication-group-id my-redis-repl-grp \
    --cache-parameter-group-name redis28-m3xl \
    --apply-immediately
```

reserved-memory-percent 파라미터

2017년 3월 16일에 Amazon ElastiCache는 reserved-memory-percent 파라미터를 도입했으며, Redis 용 ElastiCache의 모든 버전에 사용할 수 있도록 했습니다. reserved-memory-percent의 목적은 모든 클러스터에 대해 예약된 메모리 관리를 간소화하는 것입니다. 노드 유형과 상관없이 클러스터의 예약된 메모리를 관리하기 위해 각 파라미터 그룹 패밀리에 대해 단일 파라미터 그룹(예: redis2.8)을 보유할 수 있도록하여 이를 수행합니다. reserved-memory-percent에 대한 기본값은 25(25%)입니다.

Redis용 ElastiCache 파라미터 reserved-memory-percent는 Redis용 ElastiCache에 고유하며, Redis 배포의 일부가 아닙니다.

reserved-memory-percent를 사용하여 메모리를 예약하려면

reserved-memory-percent를 사용하여 Redis용 ElastiCache 클러스터에 대한 메모리를 관리하려면 다음 중 하나를 수행하십시오.

• Redis 2.8.22 이상을 실행 중이면 클러스터에 기본 파라미터 그룹을 할당합니다. 기본 25%가 적절합니다. 그렇지 않은 경우 다음 설명된 단계에 따라 값을 변경합니다. • Redis 2.8.22 이전 버전을 실행하고 있는 경우 reserved-memory-percent의 기본 25%보다 더 많은 메모리를 예약해야 할 수 있습니다. 이렇게 하려면 다음 절차를 사용하십시오.

reserved-memory-percent의 백분율 값을 변경하려면

1. 실행 중인 엔진 버전과 일치하는 파라미터 그룹 패밀리를 지정하는 사용자 지정 파라미터 그룹을 생성합니다. 예를 들어, redis2.8 파라미터 그룹 패밀리를 지정합니다. 기본 파라미터 그룹을 수정할 수 없으므로 사용자 지정 파라미터 그룹이 필요합니다. 자세한 내용은 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

```
aws elasticache create-cache-parameter-group \
    --cache-parameter-group-name redis28-50 \
    --description "Redis 2.8.x 50% reserved" \
    --cache-parameter-group-family redis2.8
```

reserved-memory-percent는 노드의 maxmemory%로 메모리를 예약하므로 각 노드 유형에 대해 사용자 지정 파라미터 그룹이 필요하지 않습니다.

2. reserved-memory-percent가 50(50%)이 되도록 사용자 지정 파라미터 그룹을 수정합니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

```
aws elasticache modify-cache-parameter-group \
    --cache-parameter-group-name redis28-50 \
    --parameter-name-values "ParameterName=reserved-memory-percent, ParameterValue=50"
```

3. Redis 2.8.22 이전 버전을 실행하는 Redis 클러스터나 복제 그룹에 대해 이 사용자 지정 파라미터 그룹을 사용합니다.

다음 CLI 예제에서는 Redis 클러스터 my-redis-cluster를 수정하여 즉시 시작되는 사용자 지정 파라미터 그룹 redis28-50을 사용하도록 합니다. 자세한 내용은 ElastiCache 클러스터 수정 (p. 87) 단원을 참조하십시오.

```
aws elasticache modify-cache-cluster \
    --cache-cluster-id my-redis-cluster \
    --cache-parameter-group-name redis28-50 \
    --apply-immediately
```

다음 CLI 예제에서는 Redis 복제 그룹 my-redis-repl-grp를 수정하여 즉시 시작되는 사용자 지정 파라미터 그룹 redis28-50을 사용하도록 합니다. 자세한 내용은 복제 그룹 수정 (p. 165) 단원을 참조 하십시오.

```
aws elasticache modify-replication-group \
    --replication-group-id my-redis-repl-grp \
    --cache-parameter-group-name redis28-50 \
    --apply-immediately
```

예약된 메모리 관리 파라미터 지정

2017년 3월 16일 현재 ElastiCache 고객이었던 사용자의 경우 예약된 메모리 관리 기본 파라미터는 예약된 메모리가 영(0) 바이트인 reserved-memory입니다. 2017년 3월 16일을 지난 시점에 ElastiCache 고객이 된 사용자의 경우 예약된 메모리 관리 기본 파라미터는 노드의 메모리 중 25%가 예약된 reserved-memory-percent입니다. 이것은 Redis용 ElastiCache 클러스터 또는 복제 그룹을 생성한 시점에 상관없이 적용됩니다. 그러나 예약된 메모리 관리 파라미터를 AWS CLI 또는 ElastiCache API를 사용해 변경할 수 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Redis AOF를 사용할 때 디스크 공간 부족 문제 완화

파라미터 reserved-memory 및 reserved-memory-percent는 함께 사용할 수 없습니다. 파라미터 그룹에는 항상 한 개의 파라미터가 있으며 둘 다 있을 수 없습니다. 파라미터 그룹을 수정하여 파라미터 그룹에서 예약된 메모리 관리에 사용할 파라미터를 변경할 수 있습니다. 기본 파라미터 그룹을 수정할 수 없으므로 파라미터 그룹은 사용자 지정 파라미터 그룹이어야 합니다. 자세한 내용은 파라미터 그룹 생성 (p. 262) 단원을 참조하십시오.

reserved-memory-percent를 지정하려면

reserved-memory-percent를 예약된 메모리 관리 파라미터로 사용하려면 modify-cache-parameter-group 명령어를 사용하여 사용자 지정 파라미터 그룹을 수정해야 합니다. reserved-memory-percent 파라미터를 사용하여 parameter-name-values 및 해당 값을 지정합니다.

다음 CLI 예제는 예약된 메모리를 관리하기 위해 reserved-memory-percent를 사용하도록 사용자 지정 파라미터 그룹 redis32-cluster-on을 수정합니다. 예약된 메모리 관리를 위해 파라미터 그룹에서 ParameterName 파라미터를 사용할 수 있도록 ParameterValue에 값을 지정해야 합니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

```
aws elasticache modify-cache-parameter-group \
--cache-parameter-group-name redis32-cluster-on \
--parameter-name-values "ParameterName=reserved-memory-percent, ParameterValue=25"
```

reserved-memory를 지정하려면

reserved-memory를 예약된 메모리 관리 파라미터로 사용하려면 modify-cache-parameter-group 명령어를 사용하여 사용자 지정 파라미터 그룹을 수정해야 합니다. reserved-memory 파라미터를 사용하 여 parameter-name-values 및 해당 값을 지정합니다.

다음 CLI 예제는 예약된 메모리를 관리하기 위해 reserved-memory를 사용하도록 사용자 지정 파라미터 그룹 redis32-m3x1을 수정합니다. 예약된 메모리 관리를 위해 파라미터 그룹에서 ParameterName 파라미터를 사용할 수 있도록 ParameterValue에 값을 지정해야 합니다. 엔진 버전이 2.8.22보다 더 새로운 버전이므로 값을 cache.m3.xlarge의 maxmemory 중 25퍼센트에 해당하는 3565158400으로 설정했습니다. 자세한 내용은 파라미터 그룹 수정 (p. 271) 단원을 참조하십시오.

```
aws elasticache modify-cache-parameter-group \
    --cache-parameter-group-name redis32-m3xl \
    --parameter-name-values "ParameterName=reserved-memory, ParameterValue=3565158400"
```

Redis AOF를 사용할 때 디스크 공간 부족 문제 완화

Amazon ElastiCache 구현을 계획할 때는 장애가 미치는 영향을 최소화하도록 계획해야 합니다.

복구 시나리오에서 AOF 파일이 유용하므로 AOF를 활성화합니다. 노드가 다시 시작되거나 서비스가 충돌할 경우 Redis가 AOF 파일에서 업데이트를 재생하여 다시 시작이나 충돌로 인해 손실된 데이터를 복구합니다.

Warning

AOF가 모든 실패 시나리오로부터 보호할 수 있는 것은 아닙니다. 예를 들어, 기본 물리적 서버에서 하드웨어 결함이 생겨 노드가 실패하면 ElastiCache가 다른 서버에 새로운 노드를 프로비저닝합니다. 이 경우 AOF 파일을 더 이상 사용할 수 없으며 이 파일을 사용하여 데이터를 복구할 수 없습니다. 따라서 Redis가 콜드 캐시로 다시 시작됩니다.

내결함성 향상을 위한 Redis 다중 AZ 활성화

데이터 손실을 방지하기 위해 AOF를 활성화하려는 경우 AOF 대신 다중 AZ를 활성화하여 복제 그룹을 사용해 보십시오. Redis 복제 그룹을 사용할 때 복제본이 실패하면 기본 클러스터를 사용하여 자동으로 대체 및동기화됩니다. Redis 복제 그룹에 다중 AZ가 활성화되고 기본이 실패하면 읽기 전용 복제본으로 장애 조치

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 모범 사례: 온라인 클러스터 크기 조정

됩니다. 일반적으로 이 기능은 AOF 파일에서 기본을 다시 빌드하는 것보다 훨씬 빠릅니다. 안정성과 복구 속도를 높이려면 읽기 전용 복제본 하나 이상이 여러 가용 영역에 있는 복제 그룹을 생성하고 AOF를 사용하는 대신 다중 AZ를 활성화하는 것이 좋습니다. 이 시나리오에서는 AOF가 필요하지 않으므로 ElastiCache가 다중 AZ 복제 그룹에서 AOF를 비활성화합니다.

자세한 내용은 다음 항목을 참조하십시오.

- 장애 완화 (p. 395)
- 고가용성을 위한 복제 그룹 사용 (p. 127)
- 작동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132)

모범 사례: 온라인 클러스터 크기 조정

리샤딩에는 클러스터에(서) 샤드 또는 노드를 추가 및 제거하고 샤드 간에 키 공간을 재분산하는 작업이 수반됩니다. 따라서 클러스터에 대한 로드, 메모리 사용률 및 데이터의 전체 크기 등과 같은 여러 가지 요인이 리샤딩 작업에 영향을 미칩니다. 최상의 성능을 구현하기 위해서는 균일한 워크로드 패턴 분산을 위한 전반적인 클러스터 모범 사례를 따르는 것이 좋습니다. 또한 다음 단계를 수행하는 것이 좋습니다.

리샤딩을 시작하기 전에 다음 작업을 수행하는 것이 좋습니다.

- 애플리케이션 테스트 가능한 경우 준비 환경에서 리샤딩 중 애플리케이션 동작을 테스트합니다.
- 확장 문제는 초기에 알리기 리샤딩은 컴퓨팅 집약적인 작업입니다. 이 때문에 리샤딩 중에는 CPU 사용률을 멀티 코어 인스턴스의 경우 80% 미만으로, 싱글 코어 인스턴스의 경우에는 50% 미만으로 유지하는 것이 좋습니다. 애플리케이션에서 확장 문제 관찰을 시작하기 전에 Redis용 ElastiCache 측정치를 모니터링한 다음 리샤딩을 시작합니다. CPUUtilization, NetworkBytesIn, NetworkBytesOut, CurrConnections, NewConnections, FreeableMemory, SwapUsage 및 BytesUsedForCache 측정치를 추적하면 유용합니다.
- 확장 전 충분한 여유 메모리를 사용할 수 있는지 확인 확장하는 경우 샤드에서 제거하려는 사용 중인 메모리의 1.5배가 넘는 여유 메모리가 샤드에 있는지 확인합니다.
- 사용량이 적은 시간에 리샤딩 시작 이 사례를 적용하면 리샤딩 작업 중 클라이언트에서 지연 시간과 처리량에 미치는 영향을 줄 일 수 있습니다. 또한 슬롯 재분산에 더 많은 리소스를 사용할 수 있기 때문에 리샤딩을 더욱 빠르게 완료할 수 있습니다.
- 클라이언트 제한 시간 동작 검토 일부 클라이언트에서는 온라인 클러스터 크기 조정 중 지연 시간이 더 길어지는 경우를 관찰할 수 있습니다. 제한 시간을 좀 더 길게 설정해 클라이언트 라이브러리를 구성하면 서버에 대한 로드가 더욱 큰 상황에서도 시스템에서 연결할 시간을 가질 수 있습니다. 일부 경우에 서버에 대해 많은 수의 연결을 열 수 있습니다. 이 경우 지수 백오프를 추가해 로직을 다시 연결합니다. 이렇게 하면 동시에 서버에 연결하는 새로운 연결이 급격하게 증가하는 것을 방지할 수 있습니다.

리샤딩 중에 다음 작업을 수행하는 것이 좋습니다.

- 비용이 많이 드는 명령 피하기 컴퓨팅 및 I/O 집약적인 작업(예: KEYS 및 SMEMBERS 명령)을 실행하지 않습니다. 이러한 작업은 클러스터에 대한 부하를 늘리고 클러스터의 성능에 영향을 미치기 때문에 이러한 접근 방식을 채택하는 것이 좋습니다. 대신 SCAN 및 SSCAN 명령을 사용합니다.
- Lua 모범 사례 따르기 오래 실행되는 Lua 스크립트의 사용을 피하고 항상 사전에 Lua 스크립트에 사용 되는 키를 선언합니다. 이렇게 하면 Lua 스크립트가 슬롯 간 명령을 사용하지 않습니다. Lua 스크립트에 사용되는 키가 동일한 슬롯에 속해 있는지 확인하십시오.

리샤딩 후에는 다음 내용에 유념하십시오.

- 대상 샤드에 사용 가능한 메모리가 부족한 경우에는 부분적인 확장만 가능했을 수 있습니다. 이러한 결과 가 발생하면 필요한 경우 사용 가능한 메모리를 검토한 후 작업을 다시 시도하십시오.
- 항목이 많은 슬롯은 마이그레이션되지 않습니다. 특히, 직렬화 후 크기가 256MB 이상인 항목이 있는 슬롯은 마이그레이션되지 않습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 모범 사례: 유지 관리 중 가동 중지 최소화

• BRPOPLPUSH 명령은 마이그레이션 중인 슬롯에서 실행되는 경우 지원되지 않습니다. FLUSHALL 및 FLUSHDB 명령은 리샤딩 작업 중 Lua 스크립트 내에서 지원되지 않습니다.

모범 사례: 유지 관리 중 가동 중지 최소화

클러스터 모드 구성은 관리 또는 비관리 작업 중에도 최고의 가용성을 제공합니다. 클러스터 탐색 엔드포인 트에 연결되는 클러스터 모드 지원 클라이언트를 사용하는 것이 좋습니다. 클러스터 모드가 비활성화된 경우 모든 쓰기 작업에 기본 엔드포인트를 사용하는 것이 좋습니다.

읽기 활동의 경우 애플리케이션은 클러스터의 어떤 노드에도 연결할 수 있습니다. 기본 엔드포인트와 달리, 노드 엔드포인트는 특정 엔드포인트로 확인됩니다. 복제본을 추가하거나 삭제하는 것과 같이 클러스터를 변 경하면 애플리케이션에서 노드 엔드포인트를 업데이트해야 합니다.

클러스터에 자동 장애 조치가 활성화되어 있는 경우, 기본 노드가 변경될 수 있습니다. 따라서, 애플리케이션은 노드의 역할을 확인하고 모든 읽기 엔드포인트를 업데이트해야 합니다. 이렇게 하면 기본에 큰 로드가 발생하지 않도록 하는 데 도움이 됩니다. 자동 장애 조치가 비활성화된 경우 노드의 역할이 변경되지 않습니다. 그러나 자동 장애 조치가 활성화된 클러스터에 비해 관리되거나 관리되지 않는 작업의 가동 중지 시간이 더 깁니다.

읽기 요청을 읽기 전용 복제본으로만 보내지 마십시오. 읽기 요청을 읽기 전용 복제복으로만 보내도록 클라이언트를 구성할 경우, 최소 두 개 이상의 읽기 전용 복제본을 마련하여 유지 관리 중 읽기 중단이 일어나지 않게 하십시오.

Amazon ElastiCache의 셀프 서비스 업데이트

Amazon ElastiCache는 Redis 클러스터 및 노드 집합을 자동으로 모니터링하여 서비스 업데이트가 제공되면 이를 적용합니다. 이는 일반적으로 ElastiCache가 이 업데이트를 적용할 수 있도록 사전 정의된 유지 관리 기간을 설정해야 한다는 의미이며, 이 기간이 지나치게 엄격하여 비즈니스 흐름을 제한할 수도 있습니다. 셀프서비스 업데이트를 통해 적용할 업데이트와 적용 시기를 제어할 수 있습니다. 선택한 Redis 클러스터에 대한 업데이트 진행 상황을 실시간으로 모니터링할 수도 있습니다. 비즈니스 요구 사항에 따라 기존 노드 및 클러스터에 대한 업데이트를 중단하기로 할 수 있습니다. 서비스 업데이트가 만료되기 전까지 언제든 적용되도록 새로운 클러스터 집합(일부 업데이트된 클러스터 포함)을 선택할 수 있습니다.

서비스 업데이트 관리

ElastiCache 서비스 업데이트는 정기적으로 릴리스됩니다. 서비스 업데이트에 대해 검증된 클러스터가 하나이상 있는 경우 이메일, SNS, Personal Health Dashboard(PHD) 및 Amazon CloudWatch 이벤트를 통해 알림을 받게 됩니다. 업데이트는 ElastiCache 콘솔의 서비스 업데이트 페이지에도 표시됩니다. 이 대시보드 보기를 이용하면 ElastiCache Redis 집합에 대한 모든 서비스 업데이트와 상태를 볼 수 있습니다. 서비스 업데이트를 위해 Redis 집합을 검토할 경우 감사 로그를 이용할 수 있습니다.

Note

이 로그를 사용하면 규정 준수를 위해 집합을 검토할 때 중요함을 증명할 수 있습니다. 자세한 내용은 규정 준수 관련 셀프 서비스 보안 업데이트 (p. 391) 단원을 참조하십시오.

그러나, 추천과 관계 없이 업데이트 적용 시기를 제어할 수 있습니다. Redis 클러스터가 항상 현행 보안 패치로 최신 상태를 유지할 수 있도록 최소한 보안 유형의 업데이트는 모두 적용할 것을 강력하게 권장합니다. 모든 Redis 클러스터의 최신 상태를 확인하려면 서비스 업데이트 상태를 선택하십시오. 이 보기는 업데이트를 적용할 수 없는 클러스터도 보여줍니다. 또한 Redis 클러스터에 적용한 업데이트가 예상 업데이트 시간을 초과하여 비즈니스 흐름을 방해할 수 있습니다. 이 경우 이를 중지하고 비즈니스 요구에 더 적합한 시점에 다시적용할 수 있습니다.

자세한 내용은 Amazon ElastiCache 유지 관리 도움말 페이지를 참조하십시오.

다음 단원에서는 다음과 같은 옵션에 대해 상세히 알아봅니다.

주제

- 셀프 서비스 업데이트 적용 (p. 412)
- 셀프 서비스 업데이트 중지 (p. 418)

셀프 서비스 업데이트 적용

업데이트가 이용 가능 상태일 때부터 만료됨 상태가 될 때까지 Redis 집합에 서비스 업데이트를 적용할 수 있습니다. 보안 유형의 서비스 업데이트는 축적됩니다. 즉, 만료되지 않은 상태로 아직 적용되지 않은 모든 업데이트가 최신 업데이트에 포함됩니다.

Note

날짜별 권장 적용의 기한이 지났더라도 사용 가능 상태인 서비스 업데이트만 적용할 수 있습니다.

Redis 집합의 검토와 Redis 클러스터에 적용 가능한 서비스별 업데이트의 적용에 관한 자세한 정보는 Redis용 콘솔을 사용한 서비스 업데이트 적용 (p. 413) 단원을 참조하십시오.

집합 내 하나 이상의 Redis 클러스터에 새로운 서비스 업데이트가 제공될 경우, ElastiCache 콘솔, API 또는 AWS CLI를 사용하여 이 업데이트를 적용할 수 있습니다. 다음 단원에서는 업데이트 적용에 사용할 수 있는 옵션에 대해 설명합니다.

콘솔을 사용한 서비스 업데이트 적용

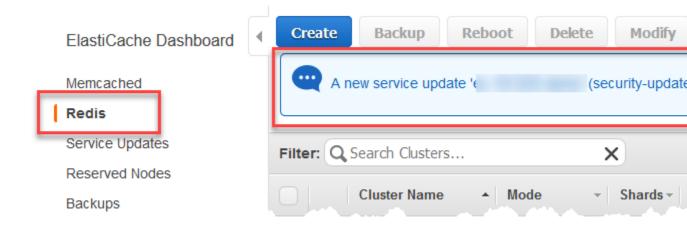
다음의 콘솔 옵션 중 하나를 사용하여 서비스 업데이트를 적용할 수 있습니다. ElastiCache는 서로 다른 두 가지 관점을 통해 업데이트의 적용 방법과 시기를 판단할 수 있도록 지원합니다.

주제

- Redis용 콘솔을 사용한 서비스 업데이트 적용 (p. 413)
- 서비스 업데이트 목록을 사용한 서비스 업데이트 적용 (p. 415)

Redis용 콘솔을 사용한 서비스 업데이트 적용

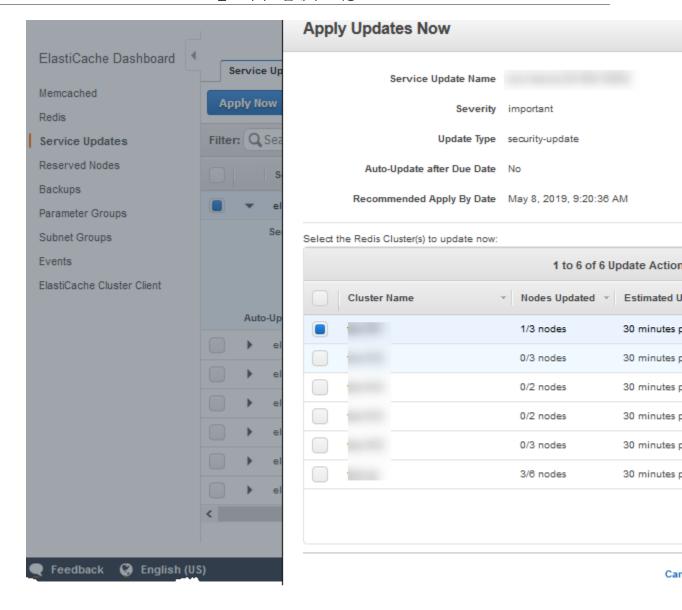
개별 Redis 클러스터의 업데이트 상태를 검토하려면 이 옵션을 선택한 다음, 해당 서비스 업데이트에 대해 적용, 보기 또는 중지를 선택하십시오. 서비스 업데이트가 제공되면 콘솔의 Redis 페이지 상단에 다음과 같이 배너가 표시됩니다.



지금 적용을 선택하면 다음과 같이 해당 워크플로 내의 적용 가능한 모든 클러스터 또는 하위 집합에 서비스 업데이트를 적용하기로 선택할 수 있습니다.

Note

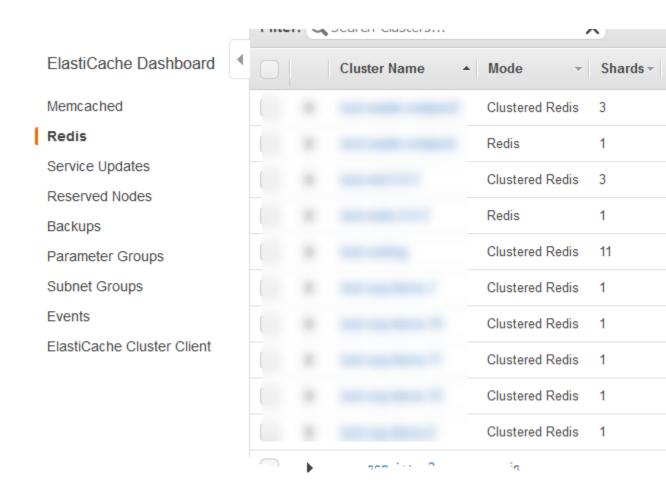
무시를 선택하면 해당 콘솔 세션 중에는 콘솔에 배너가 더 이상 표시되지 않습니다. 그러나, 세션을 새로 고침한 후에는 배너가 다시 나타납니다.



지금 업데이트 적용 페이지에 대한 다음 사항에 유의하십시오.

- Auto-Update after Due Date(기한 이후 자동 업데이트): 셀프 서비스 업데이트 만료 전에 적용하지 않을 경우, 업데이트되지 않은 모든 클러스터 또는 개별 노드가 다음 번 누적 업데이트 출시 전까지 규정 미준수 상태로 남아 있게 됩니다 ElastiCache는 사용자를 대신해 서비스 업데이트를 자동 적용하지 않습니다.
- Redis 클러스터의 업데이트된 노드 비율과 예상 업데이트 소요 시간을 통해 유지 관리 일정을 계획할 수 있습니다. 서비스 업데이트가 비즈니스 흐름의 예상 소요 시간 제약을 초과한 경우, 업데이트를 중단하고 나중에 다시 적용할 수 있습니다. 자세한 내용은 셀프 서비스 업데이트 중지 (p. 418) 단원을 참조하십시오.
- 서비스 업데이트를 가능한 모든 Redis 클러스터에 적용하기로 할 경우, 확인을 선택하십시오. 이를 선택하면, 서비스 업데이트의 상태를 모니터링할 수 있는 서비스 업데이트 페이지를 확인할 수 있습니다.
- 취소를 선택하면 다음과 같은 다른 옵션을 알아 볼 수 있습니다.

ElastiCache 대시보드에서 각 Redis 클러스터의 업데이트 상태를 확인할 수 있습니다. 업데이트 상태를 사용하여 사용 가능한 서비스 업데이트에 대한 클러스터의 규정 준수 상태를 확인할 수 있습니다.



업데이트 상태는 다음 중 하나를 표시합니다.

- 업데이트 가능: 이 클러스터에 업데이트를 적용할 수 있습니다.
- 진행 중: 업데이트가 이 클러스터에 적용되고 있으므로 예상 업데이트 소요 시간에 이 클러스터를 사용할수 없는 상태입니다.
- 중지 중: 진행 중인 업데이트가 완료 전에 중지되었습니다.
- 중지됨: 업데이트가 종료되었습니다.

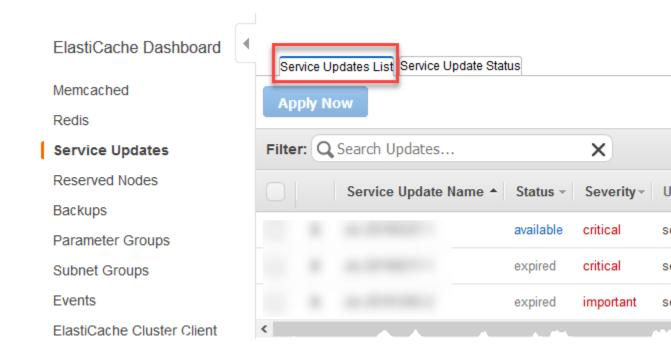
Note

Redis 클러스터에 대해 진행 중인 업데이트를 중지하면 일부 노드는 업데이트되고 다른 노드는 업데이트되지 않습니다. 중지 중 프로세스는 이미 업데이트된 노드의 변경 사항을 되돌리지 않습 니다. 아직 사용 가능 상태인 업데이트는 편의에 따라 만료됨 상태가 되기 전까지 노드에 다시 적 용할 수 있습니다.

• 최신 상태: 업데이트가 적용되었으며 클러스터가 규정을 준수합니다. 규정 준수에 대한 자세한 정보는 규정 준수 관련 셀프 서비스 보안 업데이트 (p. 391)를 참조하십시오.

서비스 업데이트 목록을 사용한 서비스 업데이트 적용

다른 관련 정보와 함께 개별 서비스 업데이트 목록과 상태를 확인하려면 서비스 업데이트 목록 탭을 선택하십시오.



서비스 업데이트 목록에는 다음이 포함되어 있습니다.

- 서비스 업데이트 이름: 서비스 업데이트의 고유 식별자입니다.
- 상태: 업데이트의 상태로, 다음 중 하나에 해당합니다.
 - 사용 가능: 업데이트를 필요한 Redis 클러스터에 사용할 수 있습니다.
 - 완료: 업데이트가 적용되었으며 모든 Redis 클러스터는 규정을 준수합니다. (자세한 내용은 규정 준수 관련 셀프 서비스 보안 업데이트 (p. 391) 단원을 참조하십시오.)
 - 취소됨: 업데이트가 취소되었으며 더 이상 필요하지 않습니다.
 - 만료됨: 업데이트를 더 이상 적용할 수 없습니다.
- 심각도: 업데이트 적용의 우선 순위를 나타냅니다.
 - 심각: 즉시 적용하는 것이 좋습니다(14일 이내).
 - 중요: 비즈니스 흐름 상 가능한 한 빨리 적용하는 것이 좋습니다(30일 이내).
 - 중간: 비즈니스 흐름 상 가능한 한 빨리 적용하는 것이 좋습니다(60일 이내)
 - 낮음: 비즈니스 흐름 상 가능한 한 빨리 적용하는 것이 좋습니다(90일 이내)
- 업데이트 유형: 이 버전에는 보안 업데이트만 지원됩니다.
- 릴리스 날짜: 업데이트가 릴리스되어 Redis 집합에 적용할 수 있게 된 날짜입니다.
- Recommended Apply By Date(날짜별 권장 적용): 업데이트를 적용해야 하는 ElastiCache 지침 일자입니다.

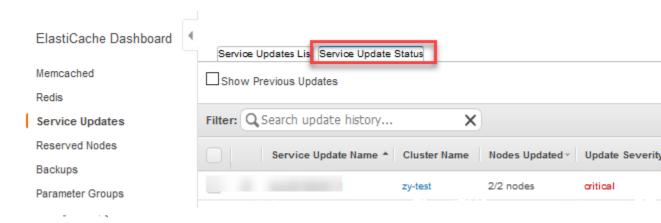
개별 업데이트를 선택하면 다음과 같이 추가적인 세부 정보가 제공됩니다.

- 업데이트 설명: 서비스 업데이트에 관한 세부 정보를 제공합니다.
- 업데이트 만료 일자: 서비스 업데이트가 만료되어 더 이상 제공되지 않는 날짜를 의미합니다. 만료 일자 이 전에 적용되지 않은 모든 업데이트는 다음 업데이트로 누적됩니다.

Important

보안 유형의 업데이트는 비즈니스 흐름 상 가능한 한 빨리 적용하는 것이 좋습니다. 이렇게 하면 항상 최신 보안 패치로 Redis 클러스터를 최신 상태로 유지하고 규정을 준수할 수 있습니다. 자세한 내용은 규정 준수 관련 셀프 서비스 보안 업데이트 (p. 391) 단원을 참조하십시오.

해당하는 Redis 클러스터와 관련된 개별 서비스 업데이트 목록을 검토하려면 Service Update Status(서비스 업데이트 상태) 탭을 선택하십시오.



Service Updates Status(서비스 업데이트 상태) 목록을 확인할 때는 다음 사항을 참고하십시오.

- 서비스 업데이트 이름: 서비스 업데이트에 관한 상세 정보를 제공합니다.
- 클러스터 이름: 업데이트 대상인 Redis 클러스터의 목록입니다.
- 업데이트된 노드: 업데이트되었거나 특정 서비스 업데이트를 여전히 사용할 수 있는 특정 클러스터 내 개별 노드의 비율입니다.
- 업데이트 심각도: 업데이트 적용의 우선 순위를 나타냅니다.
 - 심각: 즉시 적용하는 것이 좋습니다(14일 이내).
 - 중요: 비즈니스 흐름 상 가능한 한 빨리 적용하는 것이 좋습니다(30일 이내).
 - 중간: 비즈니스 흐름 상 가능한 한 빨리 적용하는 것이 좋습니다(60일 이내)
 - 낮음: 비즈니스 흐름 상 가능한 한 빨리 적용하는 것이 좋습니다(90일 이내)
- 업데이트 유형: 이 버전에는 보안 업데이트만 지원됩니다.
- Service Update Status(서비스 업데이트 상태): 업데이트의 상태로, 다음 중 하나에 해당합니다.
 - 사용 가능: 업데이트를 필요한 Redis 클러스터에 사용할 수 있습니다.
 - 완료: 업데이트가 적용되었으며 모든 Redis 클러스터는 규정을 준수합니다.
 - 취소됨: 업데이트가 취소되었으며 더 이상 필요하지 않습니다.
 - 만료됨: 업데이트를 더 이상 적용할 수 없습니다.
- Service Update SLA Met(서비스 업데이트 SLA 충족됨): 클러스터가 규정을 준수하는지 여부를 반영합니다.
 - 예: 제공되는 모든 업데이트가 날짜별 적용까지 이 클러스터와 사용 가능한 노드에 적용되었습니다.
 - 아니요: 서비스 업데이트가 하나 이상의 노드에 성공적으로 적용되었을 수 있으나 클러스터 내의 다른 노드는 여전히 사용 가능 상태입니다. 이는 보통 서비스 업데이트가 적용되었다가 중지된 경우에 나타납 니다.

Note

클러스터에 대한 서비스 업데이트 진행을 중지할 경우에도 이미 업데이트된 노드는 완료 상태가 됩니다. 진행 중 또는 중지 중 상태인 노드는 중지됨 상태가 되고 Service Update SLA Met(서비스 업데이트 SLA 충족됨) 상태가 아니요로 변경됩니다.

- 해당 사항 없음: 권장되는 날짜별 적용 이후에 복제 그룹이 생성되었습니다.
- Cluster Status Modified Date(클러스터 상태 수정 날짜): 서비스 업데이트로 인해 클러스터가 수정된 최근 날짜입니다.

Note

Show Previous Updates(이전 업데이트 보기) 확인란을 선택하면 더 이상 제공되지 않는 이전 업데 이트의 목록이 표시됩니다.

AWS CLI를 사용한 서비스 업데이트 적용

서비스 업데이트가 제공된다는 알림을 받은 후 AWS CLI를 사용하여 해당 업데이트를 검사하고 적용할 수 있습니다.

• 제공되는 서비스 업데이트의 설명을 검색하려면:

aws elasticache describe-service-updates --service-update-status available

자세한 정보는 DescribeServiceUpdates를 참조하십시오.

• not-applied 또는 stopped 상태인 업데이트 작업을 검토하려면:

aws elasticache describe-update-actions --service-update-name sample-service-update --update --update action-status not-applied stopped

자세한 내용은 DescribeUpdateActions를 참조하십시오.

• 복제본 그룹 목록에 서비스 업데이트를 적용하려면:

aws elasticache batch-apply-update-action --service-update-name sample-service-update --replication-group-ids my-replication-group-1 my-replication-group-2

자세한 정보는 BatchApplyUpdateAction을 참조하십시오.

셀프 서비스 업데이트 중지

업데이트를 진행 중인 Redis 클러스터에 예상치 못한 서지가 발생하거나, 업데이트에 소요되는 시간이 지나 치게 길어 비즈니스 흐름 피크 타임에 방해가 되는 경우 업데이트를 중지할 수 있습니다.

중지 중 작업은 이들 노드와 아직 업데이트되지 않은 노드에 대한 모든 업데이트를 즉시 중지합니다. 진행 중 상태인 노드는 완료될 때까지 계속됩니다. 그러나, 업데이트 사용 가능 상태인 동일한 클러스터 내의 다른 노 드에 대한 업데이트는 중단되며 중지 중 상태로 변경됩니다.

중지 중 워크플로가 완료되면 중지 중 상태인 노드는 중지됨 상태가 됩니다. 업데이트의 워크플로에 따라 일부 클러스터의 노드는 업데이트되지 않습니다. 다른 클러스터에는 업데이트된 노드와 여전히 업데이트 사용가능 상태인 노드가 포함될 수 있습니다.

비즈니스 흐름 상 가능할 때 업데이트 프로세스를 완료할 수 있습니다. 이 경우, 업데이트를 완료할 해당 클러스터를 선택한 다음 지금 적용을 선택하십시오. 자세한 내용은 Redis용 콘솔을 사용한 서비스 업데이트 적용 (p. 413) 단원을 참조하십시오.

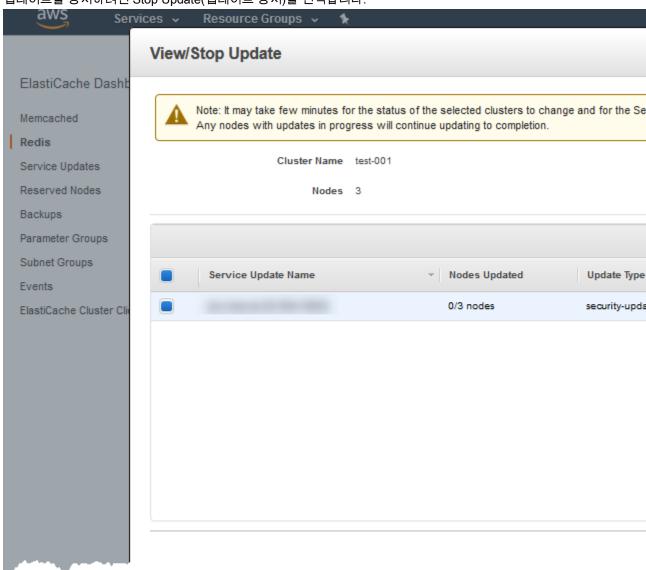
콘솔을 사용한 서비스 업데이트 중지

Redis 콘솔을 사용해 서비스 업데이트를 중지할 수 있습니다. 다음에서는 이 작업을 수행하는 방법을 보여줍니다.

• 선택한 Redis 클러스터에 서비스 업데이트가 진행된 후 ElastiCache 콘솔의 Redis 대시보드 상단에 View/ Stop Update(업데이트 보기/중지) 탭이 나타납니다.

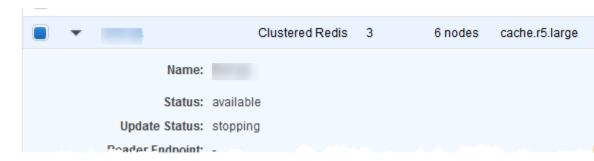


• 업데이트를 중지하려면 Stop Update(업데이트 중지)를 선택합니다.



• 업데이트를 중지할 경우 Redis 클러스터를 선택하고 상태를 확인하십시오. 다음과 같이 중지 중 상태로 바뀌었다가 최종적으로 중지됨 상태가 됩니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 셀프 서비스 업데이트 중지



AWS CLI를 사용한 서비스 업데이트 중지

AWS CLI를 사용해 서비스 업데이트를 중지할 수 있습니다. 다음 코드 예제에서는 이를 수행하는 방법을 보여 줍니다.

aws elasticache batch-stop-update-action --service-update-name sample-service-update --replication-group-ids my-replication-group-1 my-replication-group-2

자세한 정보는 BatchStopUpdateAction을 참조하십시오.

참조

이 단원의 본 주제에서는 Amazon ElastiCache API 및 AWS CLI의 ElastiCache 섹션 관련 작업을 다룹니다. 또한 여기에는 일반적인 오류 메시지와 서비스 알림에 관한 설명이 포함되어 있습니다.

- ElastiCache API 사용 (p. 421)
- ElastiCache API 참조
- AWS CLI 참조의 ElastiCache 단원
- Amazon ElastiCache 오류 메시지 (p. 431)
- 알림 (p. 432)

ElastiCache API 사용

이 단원에서는 ElastiCache 작업 사용 방법 및 구현에 관해 작업 중심으로 설명합니다. 이러한 작업 설명 전체를 보려면 Amazon ElastiCache API Reference 단원을 참조하십시오.

주제

- Query API 사용 (p. 421)
- 사용 가능한 라이브러리 (p. 423)
- 애플리케이션 문제 해결 (p. 424)
- AWS CloudTrail을 사용하여 Amazon ElastiCache API 호출 로깅 (p. 424)

Query API 사용

쿼리 매개 변수

HTTP 쿼리 기반 요청은 GET 또는 POST와 같은 HTTP 동사와 Action 쿼리 매개 변수를 사용하는 HTTP 요청입니다.

각 쿼리 요청은 인증 및 작업을 처리할 수 있도록 일부 공통 파라미터를 포함해야 합니다.

일부 작업은 파라미터의 목록을 허용합니다. 이러한 목록은 param.n 표기법을 사용하여 지정됩니다. n의 값은 1부터 시작하는 정수입니다.

쿼리 요청 인증

HTTPS를 통해서만 쿼리 요청을 보낼 수 있으며 모든 쿼리 요청에는 서명이 포함되어야 합니다. 이 단원에서는 서명을 작성하는 방법을 설명합니다. 아래 절차에 설명된 방법은 서명 버전 4라고 합니다.

다음은 AWS에 대한 요청을 인증하는 데 사용되는 기본 단계입니다. 이 경우 사용자가 AWS에 등록되어 있으며 액세스 키 ID 및 보안 액세스 키를 가지고 있다고 가정합니다.

쿼리 인증 절차

- 1. 발신자가 AWS에 대한 요청을 구성합니다.
- 2. 발신자가 이 항목의 다음 단원에 정의된 방법으로 SHA-1 해시 기능을 사용하는 HMAC(Hash-based Message Authentication Code)에 대한 키 해싱인 요청 서명을 계산합니다.
- 3. 요청의 발신자가 요청 데이터, 서명 및 액세스 키 ID(사용된 보안 액세스 키의 키 식별자)를 AWS로 보냅니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 Query API 사용

- 4. AWS는 액세스 키 ID를 사용하여 보안 액세스 키를 찾습니다.
- 5. AWS는 요청의 서명 계산에 사용된 동일한 알고리즘을 사용하여 요청 데이터 및 보안 액세스 키에서 서명을 생성합니다.
- 6. 서명이 일치하는 경우 요청이 인증되는 것으로 간주됩니다. 서명이 일치하지 않는 경우 요청이 삭제되고 AWS는 오류를 반환합니다.

Note

Timestamp 매개 변수가 요청에 포함된 경우 요청에 대해 계산된 서명은 그 매개 변수 값보다 15분후에 만료됩니다.

Expires 매개 변수가 요청에 포함된 경우 그 서명은 Expires 매개 변수에 의해 지정된 시간에 만료됩니다.

요청 서명을 계산하려면

- 1. 정규화된 쿼리 문자열을 만듭니다. 이 절차의 뒷부분에서 필요합니다.
 - a. UTF-8 쿼리 문자열 구성 요소를 매개 변수 이름의 일반 바이트 순서로 정렬합니다. 이 매개 변수는 GET URI 또는 POST 요청 본문(Content-Type이 application/x-www-form-urlencoded일 경우)의 내용이 사용될 수 있습니다.
 - b. 다음 규칙에 따라 매개 변수 이름과 값을 URL 인코딩합니다.
 - i. RFC 3986에 정의된 예약되지 않은 모든 문자는 URL 인코딩하지 않습니다. 이러한 예약되지 않은 문자는 A ~ Z, a ~ z, 0 ~ 9, 하이픈(-), 밑줄(_), 마침표(.) 및 물결표(~)입니다.
 - ii. %XY와 같이 모든 기타 문자를 퍼센트 인코딩합니다(여기서 X 및 Y는 16진 문자 0 ~ 9 및 대문자 A ~ F).
 - iii. 확장된 UTF-8 문자는 %XY%ZA... 형식으로 퍼센트 인코딩합니다.
 - iv. 공백 문자는 %20(일반 인코딩 구조인 +가 아님)으로 퍼센트 인코딩합니다.
 - c. 매개 변수 값이 비어있는 경우에도 인코딩된 매개 변수 이름을 인코딩된 매개 변수 값과 등호(=) (ASCII 문자 61)로 구별합니다.
 - d. 앰퍼샌드(&)(ASCII 코드 38)로 이름-값 쌍을 구별합니다.
- 2. 다음의 의사(pseudo) 문법("\n"은 ASCII 줄 바꿈을 나타냄)에 따라 서명할 문자열을 만듭니다.

```
StringToSign = HTTPVerb + "\n" +
ValueOfHostHeaderInLowercase + "\n" +
HTTPRequestURI + "\n" +
CanonicalizedQueryString <from the preceding step>
```

HTTPRequestURI 구성 요소는 URI의 HTTP 절대 경로 구성 요소이고 쿼리 문자열은 포함하지는 않습니다. HTTPRequestURI가 비어있는 경우 슬래시(/)를 사용합니다.

3. 사용자의 보안 액세스 키를 키로, SHA256 또는 SHA1을 해시 알고리즘으로 하여 방금 만든 문자열로 RFC 2104 호환 HMAC를 계산합니다.

자세한 내용은 https://www.ietf.org/rfc/rfc2104.txt를 참조하십시오.

- 4. 결과 값을 base64로 변환합니다.
- 5. 요청에서 Signature 매개 변수 값을 값으로 포함합니다.

예를 들어, 다음은 샘플 요청입니다(줄 바꿈이 명확성을 위해 추가됨).

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheClusters
&CacheClusterIdentifier=myCacheCluster
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 사용 가능한 라이브러리

&SignatureMethod=HmacSHA256 &SignatureVersion=4 &Version=2014-12-01

이전 쿼리 문자열의 경우 다음 문자열을 통해 HMAC 서명을 계산합니다.

```
elasticache.amazonaws.com\n
   Action=DescribeCacheClusters
   &CacheClusterIdentifier=myCacheCluster
    &SignatureMethod=HmacSHA256
   &SignatureVersion=4
    &Version=2014-12-01
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Credential=AKIADQKE4SARGYLE%2F20140523%2Fus-west-2%2Felasticache%2Faws4_request
    &X-Amz-Date=20141201T223649Z
   &X-Amz-SignedHeaders=content-type%3Bhost%3Buser-agent%3Bx-amz-content-sha256%3Bx-amz-
date
        content-type:
       \verb|host:elasticache.us-west-2.amazonaws.com||\\
       user-agent:CacheServicesAPICommand_Client
    x-amz-content-sha256:
    x-amz-date:
```

이 결과는 다음의 서명된 요청입니다.

서명 프로세스 및 요청 서명 계산에 대한 자세한 내용은 서명 버전 4 서명 프로세스 항목과 그 하위 항목을 참조하십시오.

사용 가능한 라이브러리

AWS는 Query API 대신 언어별 API를 사용하여 애플리케이션을 구축하려는 소프트웨어 개발자를 위한 소프트웨어 개발 키트(SDK)를 제공합니다. 이러한 SDK는 보다 쉽게 시작하도록 요청 인증, 요청 재시도 및 오류처리 같은 기본 기능(API에는 포함되지 않음)을 제공합니다. SDK 및 추가 리소스는 다음 프로그래밍 언어에 대해 제공됩니다.

- Java
- Windows 및 .NET
- PHP
- Python
- Ruby

다른 언어에 대한 자세한 내용은 샘플 코드 및 라이브러리를 참조하십시오.

애플리케이션 문제 해결

ElastiCache는 ElastiCache API와 상호 작용하는 동안 발생하는 문제를 해결할 때 도움이 되도록 구체적이고 서술적인 오류를 제공합니다.

오류 검색

일반적으로 사용자는 시간을 소비하여 결과를 처리하기 전에 애플리케이션이 먼저 해당 요청으로 오류가 발생되는지 여부를 확인하려고 합니다. 오류 발생 여부를 확인하는 가장 쉬운 방법은 ElastiCache API의 응답에서 Error 노드를 찾는 것입니다.

XPath 구문은 Error 노드의 발생뿐만 아니라 오류 코드 및 메시지를 쉽게 검색할 수 있는 간단한 방법을 제공합니다. 다음 코드 조각에서는 요청 중에 오류가 발생했는지 여부를 파악하기 위해 Perl 및 XML::Xpath 모듈을 사용합니다. 오류가 발생되면 코드는 응답에 첫 번째 오류 코드와 메시지를 인쇄합니다.

```
use XML::XPath;
my $xp = XML::XPath->new(xml =>$response);
if ( $xp->find("//Error") )
{print "There was an error processing your request:\n", " Error code: ",
$xp->findvalue("//Error[1]/Code"), "\n", " ",
$xp->findvalue("//Error[1]/Message"), "\n\n"; }
```

문제 해결 틴

다음 절차를 통해 ElastiCache API의 문제를 진단하고 해결하는 것이 좋습니다.

• ElastiCache가 올바르게 실행되는지 확인합니다.

이렇게 하려면, 브라우저 창을 열고 ElastiCache 서비스(https://elasticache.amazonaws.com 등)에 쿼리 요 청을 제출하면 됩니다. MissingAuthenticationTokenException 또는 500 내부 서버 오류가 발생하는 경우 서비스가 사용 가능하고 요청에 응답하는지 확인할 수 있습니다.

• 요청 구조 확인.

각 ElastiCache 작업에 대한 참조 페이지는 ElastiCache API 참조에 있습니다. 파라미터를 올바르게 사용하고 있는지 여부를 다시 확인합니다. 어떤 문제가 발생할 수 있을 지에 대해 미리 알아보려면 샘플 요청이나 사용자 시나리오를 살펴보고 이러한 샘플이 유사한 작업을 하고 있는지 확인하십시오.

• 포럼 확인.

ElastiCache에는 그 과정에서 다른 사람들이 경험한 문제에 대한 해결책을 검색할 수 있는 토론 포럼이 있습니다. 포럼을 보려면 다음 사이트를 참조하십시오.

https://forums.aws.amazon.com/ .

AWS CloudTrail을 사용하여 Amazon ElastiCache API 호출 로깅

Amazon ElastiCache는 Amazon ElastiCache의 사용자, 역할 또는 AWS 서비스가 수행한 작업에 대한 레코드를 제공하는 서비스인 AWS CloudTrail과 통합됩니다. CloudTrail은 Amazon ElastiCache 콘솔의 호출 및 Amazon ElastiCache API 코드 호출 등 Amazon ElastiCache에 대한 모든 API 작업 호출을 이벤트로 캡처합니다. 추적을 생성하면 Amazon ElastiCache에 대한 이벤트를 비롯하여 CloudTrail 이벤트를 Amazon S3 버킷으로 지속적으로 배포할 수 있습니다. 추적을 구성하지 않은 경우 Event history(이벤트 기록)에서 CloudTrail 콘솔의 최신 이벤트를 볼 수도 있습니다. CloudTrail에서 수집하는 정보를 사용하여 Amazon ElastiCache에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 AWS CloudTrail을 사용하여

Amazon ElastiCache API 호출 로깅

CloudTrail에 대한 자세한 내용은 AWS CloudTrail User Guide를 참조하십시오.

CloudTrail의 Amazon ElastiCache 정보

CloudTrail은 계정 생성 시 AWS 계정에서 활성화됩니다. Amazon ElastiCache에서 활동이 수행되면 해당 활동은 Event history(이벤트 기록)에서 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트에 기록됩니다. AWS 계정에서 최신 이벤트를 확인, 검색 및 다운로드할 수 있습니다. 자세한 내용은 CloudTrail 이벤트 기록에서 이벤트 보기를 참조하십시오.

Amazon ElastiCache에 대한 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려는 경우 추적을 생성합니다. 추적은 CloudTrail이 Amazon S3 버킷으로 로그 파일을 전송할 수 있도록 합니다. 콘솔에서 추적을 생성하면 기본적으로 모든 리전에 추적이 적용됩니다. 추적은 AWS 파티션에 있는 모든 리전의 이벤트를 로깅하고 지정된 Amazon S3 버킷으로 로그 파일을 전송합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.

- 추적 생성 개요
- CloudTrail 지원 서비스 및 통합
- CloudTrail에 대한 Amazon SNS 알림 구성
- 여러 리전에서 CloudTrail 로그 파일 받기 및 여러 계정에서 CloudTrail 로그 파일 받기

모든 Amazon ElastiCache 작업이 CloudTrail에서 로깅되고 ElastiCache API Reference에 문서화됩니다. 예를 들어, CreateCacheCluster, DescribeCacheCluster 및 ModifyCacheCluster 작업을 호출하면 CloudTrail 로그 파일에 항목이 생성됩니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 자격 증명 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트로 했는지 아니면 IAM 사용자 자격 증명으로 했는지 여부
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 CloudTrail userIdentity 요소를 참조하십시오.

Amazon ElastiCache 로그 파일 항목 이해

추적은 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 제공할 수 있도록 해 주는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함됩니다. 이벤트는 어떤 소스로부터의 단일 요청을 나타내며 요청된 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 들어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 호출의 주문 스택 추적이 아니기 때문에 특정 순서로 표시되지 않습니다.

다음은 CreateCacheCluster 작업을 보여 주는 CloudTrail 로그 항목이 나타낸 예제입니다.

```
{
   "eventVersion":"1.01",
   "userIdentity":{
        "type":"IAMUser",
        "principalId":"EXAMPLEEXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/elasticache-allow",
        "accountId":"123456789012",
        "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
        "userName":"elasticache-allow"
},
   "eventTime":"2014-12-01T22:00:35Z",
   "eventSource":"elasticache.amazonaws.com",
   "eventName":"CreateCacheCluster",
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 AWS CloudTrail을 사용하여 Amazon ElastiCache API 호출 로깅

```
"awsRegion":"us-west-2",
    "sourceIPAddress": "192.0.2.01",
    "userAgent": "Amazon CLI/ElastiCache 1.10 API 2014-12-01",
    "requestParameters":{
        "numCacheNodes":2,
        "cacheClusterId": "test-memcached",
        "engine": "memcached",
        "aZMode": "cross-az",
        "cacheNodeType": "cache.m1.small"
    "responseElements":{
        "engine": "memcached",
        "clientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "cacheParameterGroup":{
            "cacheParameterGroupName": "default.memcached1.4",
            "cacheNodeIdsToReboot":{
            "parameterApplyStatus": "in-sync"
        "preferredAvailabilityZone": "Multiple",
        "numCacheNodes":2,
        "cacheNodeType":"cache.m1.small",
        "cacheClusterStatus": "creating",
        "autoMinorVersionUpgrade":true,
        "preferredMaintenanceWindow": "thu: 05:00-thu: 06:00",
        "cacheClusterId": "test-memcached",
        "engineVersion": "1.4.14",
        "cacheSecurityGroups":[
                "status": "active",
                "cacheSecurityGroupName": "default"
        "pendingModifiedValues":{
    },
    "requestID": "104f30b3-3548-11e4-b7b8-6d79ffe84edd",
    "eventID": "92762127-7a68-42ce-8787-927d2174cde1"
}
```

다음은 DescribeCacheCluster 작업을 보여 주는 CloudTrail 로그 항목이 나타낸 예제입니다. 모든 Amazon ElastiCache Describe 호출(Describe*)의 경우 ResponseElements 단원이 제거되며 null으로 표시됩니다.

```
{
    "eventVersion":"1.01",
    "userIdentity":{
        "type": "IAMUser",
        "principalId": "EXAMPLEEXAMPLEEXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/elasticache-allow",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "elasticache-allow"
   },
    "eventTime": "2014-12-01T22:01:00Z",
    "eventSource": "elasticache.amazonaws.com",
    "eventName": "DescribeCacheClusters",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.01",
    "userAgent": "Amazon CLI/ElastiCache 1.10 API 2014-12-01",
    "requestParameters":{
        "showCacheNodeInfo":false,
        "maxRecords":100
```

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 AWS CloudTrail을 사용하여 Amazon ElastiCache API 호출 로깅

```
},
    "responseElements":null,
    "requestID":"1f0b5031-3548-11e4-9376-c1d979ba565a",
    "eventID":"a58572a8-e81b-4100-8e00-1797ed19d172"
}
```

다음 예제는 ModifyCacheCluster 작업을 기록하는 CloudTrail 로그 항목을 보여줍니다.

```
"eventVersion":"1.01",
    "userIdentity":{
        "type": "IAMUser",
        "principalId": "EXAMPLEEXAMPLEEXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/elasticache-allow",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "elasticache-allow"
   },
    "eventTime": "2014-12-01T22:32:21Z",
    "eventSource": "elasticache.amazonaws.com",
    "eventName": "ModifyCacheCluster",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.01",
    "userAgent": "Amazon CLI/ElastiCache 1.10 API 2014-12-01",
    "requestParameters":{
        "applyImmediately":true,
        "numCacheNodes":3,
        "cacheClusterId": "test-memcached"
    "responseElements":{
        "engine": "memcached",
        "clientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "cacheParameterGroup":{
            "cacheParameterGroupName": "default.memcached1.4",
            "cacheNodeIdsToReboot":{
            "parameterApplyStatus": "in-sync"
        },
        "cacheClusterCreateTime": "Dec 1, 2014 10:16:06 PM",
        "preferredAvailabilityZone": "Multiple",
        "numCacheNodes":2,
        "cacheNodeType": "cache.m1.small",
        "cacheClusterStatus": "modifying",
        "autoMinorVersionUpgrade":true,
        "preferredMaintenanceWindow": "thu: 05:00-thu: 06:00",
        "cacheClusterId": "test-memcached",
        "engineVersion": "1.4.14",
        "cacheSecurityGroups":[
            {
                "status": "active",
                "cacheSecurityGroupName": "default"
        ٦,
        "configurationEndpoint":{
            "address": "test-memcached.example.cfg.use1prod.cache.amazonaws.com",
            "port":11211
        "pendingModifiedValues":{
            "numCacheNodes":3
        }
    "requestID": "807f4bc3-354c-11e4-9376-c1d979ba565a",
    "eventID": "e9163565-376f-4223-96e9-9f50528da645"
```

}

ElastiCache 명령줄 인터페이스 설정

이 섹션은 명령행 도구 실행을 위한 필수 조건, 명령행 도구를 구할 수 있는 위치, 도구 및 환경 설정 방법을 설명하고 도구 사용의 몇몇 일반적인 예를 포함하고 있습니다.

ElastiCache용 AWS CLI로 이동하는 경우에만 이 주제의 지침을 따르십시오.

Important

Amazon ElastiCache 명령줄 인터페이스(CLI)는 API 버전 2014-09-30 이후의 ElastiCache 개선 사항을 지원하지 않습니다. 명령줄에서 최신 ElastiCache 기능을 사용하려면 AWS 명령줄 인터페이스를 사용합니다.

주제

- 사전 요구사항 (p. 428)
- 명령행 도구 얻기 (p. 429)
- 도구 설정 (p. 429)
- 도구에 대한 자격 증명 제공 (p. 430)
- 환경 변수 (p. 431)

사전 요구사항

이 문서는 Linux/UNIX 또는 Windows 환경에서 작업할 수 있음을 가정합니다. 또한 Amazon ElastiCache 명령줄 도구는 UNIX 기반 환경인 Mac OS X에서도 작동하지만 이 설명서에는 특정 Mac OS X 지침이 포함되어 있지 않습니다.

하나의 규칙으로서 모든 명령줄 텍스트 앞에 일반적인 PROMPT> 명령줄 프롬프트가 나옵니다. 머신의 실제명령행 프롬프트는 다를 수 있습니다. 또한 Linux/UNIX 고유 명령을 표시하기 위해서는 \$ 를, Windows 고유명령에 대해서는 C:\> 을 사용합니다. 명령의 결과인 출력 예는 접두사 없이 그 후에 즉시 표시됩니다.

Java 런타임 환경

이 설명서에 사용된 명령행 도구를 실행하려면 Java 버전 5 이상이 있어야 합니다. JRE 또는 JDK 설치가 허용됩니다. Linux/UNIX 및 Windows를 포함한 다양한 플랫폼 용도의 JRE를 살펴보고 다운로드하려면 Java SE Downloads를 참조하십시오.

Java Home 변수 설정

명령행 도구는 Java 런타임을 찾기 위해 환경 변수(JAVA_HOME)를 사용합니다. 이 환경 변수는 실행 가능한 bin(Linux 및 UNIX) 또는 java(Windows) 실행 파일을 차례로 포함하고 있는 java.exe이라는 하위 디렉토리가 있는 디렉토리의 전체 경로로 설정되어야 합니다.

Java Home 변수를 설정하려면

- 1. Java Home 변수를 설정합니다.
 - Linux 및 UNIX에서 다음 명령을 입력합니다.

\$ export JAVA_HOME=<PATH>

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 명령행 도구 얻기

• Windows에서 다음 명령을 입력합니다.

```
C:\> set JAVA_HOME=<PATH>
```

- 2. \$JAVA HOME/bin/java -version을 실행하고 출력을 확인하여 경로 설정을 확인합니다.
 - Linux/UNIX에서 다음과 유사한 출력을 확인할 수 있습니다.

```
$ $JAVA_HOME/bin/java -version
java version "1.6.0_23"
Java(TM) SE Runtime Environment (build 1.6.0_23-b05)
Java HotSpot(TM) Client VM (build 19.0-b09, mixed mode, sharing)
```

• Windows에서 다음과 유사한 출력을 확인할 수 있습니다.

```
C:\> %JAVA_HOME%\bin\java -version
java version "1.6.0_23"
Java(TM) SE Runtime Environment (build 1.6.0_23-b05)
Java HotSpot(TM) Client VM (build 19.0-b09, mixed mode, sharing)
```

명령행 도구 얻기

명령줄 도구는 ElastiCache 개발자 도구 웹 사이트에서 ZIP 파일로 제공합니다. 이러한 도구는 Java로 작성 되었으며 Windows 2000/XP/Vista/Windows 7, Linux/UNIX 및 Mac OSX에 대한 셸 스크립트를 포함하고 있습니다. ZIP 파일은 자체 포함되어 있고 설치가 필요 없으므로 간단히 해당 Zip 파일을 다운로드하여 로컬 머신의 디렉토리에 압축을 풉니다.

도구 설정

명령행 도구는 지원 라이브러리를 찾기 위해 환경 변수(AWS_ELASTICACHE_HOME)를 사용합니다. 이 환경 변수를 먼저 설정해야 도구를 사용할 수 있습니다. 환경 변수를 명령행 도구의 압축을 푼 디렉토리 경로로 설정합니다. 이 디렉토리 이름은 ElastiCacheCli-A.B.nnnn(A, B 및 n은 버전/릴리스 번호)이며 bin 및 lib라는하위 디렉토리를 포함하고 있습니다.

AWS_ELASTICACHE_HOME 환경 변수를 설정하려면

- 명령행 창을 열고 다음 명령 중 하나를 입력하여 AWS ELASTICACHE HOME 환경 변수를 설정합니다.
 - Linux 및 UNIX에서 다음 명령을 입력합니다.

```
$ export AWS_ELASTICACHE_HOME=<path-to-tools>
```

• Windows에서 다음 명령을 입력합니다.

```
C:\> set AWS_ELASTICACHE_HOME=<path-to-tools>
```

도구 사용을 좀 더 쉽게 하려면 도구의 BIN 디렉토리를 시스템 경로에 추가하는 것이 좋습니다. 이 설명서의 나머지 부분은 BIN 디렉토리가 시스템 경로에 있음을 가정합니다.

도구의 BIN 디렉토리를 시스템 경로에 추가하려면

다음 명령을 입력하여 도구의 BIN 디렉토리를 시스템 경로에 추가합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 도구에 대한 자격 증명 제공

• Linux 및 UNIX에서 다음 명령을 입력합니다.

\$ export PATH=\$PATH:\$AWS_ELASTICACHE_HOME/bin

• Windows에서 다음 명령을 입력합니다.

C:\> set PATH=%PATH%;%AWS_ELASTICACHE_HOME%\bin

Note

Windows 환경 변수는 명령 창을 닫으면 재설정됩니다. 이러한 변수를 영구적으로 설정할 수 있습니다. 자세한 내용은 현재 Windows 버전의 설명서를 참조하십시오.

Note

경로에 공백이 포함된 경우 다음 예에서처럼 큰따옴표로 묶어야 합니다. "C:\Program Files\Java"

도구에 대한 자격 증명 제공

명령행 도구는 AWS 계정과 함께 제공되는 AWS 액세스 키 및 보안 액세스 키가 필요합니다. 명령행을 사용 하거나 로컬 시스템에 위치하는 자격 증명 파일에서 이러한 키를 얻을 수 있습니다.

배포 파일에는 개별 정보를 사용하여 수정해야 하는 템플릿 파일인 \${AWS_ELASTICACHE_HOME}/credential-file-path.template가 포함되어 있습니다. 다음은 템플릿 파일의 콘텐츠입니다.

AWSAccessKeyId=<Write your AWS access ID>
AWSSecretKey=<Write your AWS secret key>

Important

UNIX에서는 자격 증명 파일의 소유자로 권한을 제한합니다.

\$ chmod 600 <the file created above>

자격 증명 파일 설정을 사용하여 AWS_CREDENTIAL_FILE 환경 변수를 설정해야 ElastiCache 도구가 정보를 찾을 수 있습니다.

AWS CREDENTIAL FILE 환경 변수를 설정하려면

- 1. 환경 변수 설정:
 - Linux 및 UNIX에서는 다음 명령을 사용하여 변수를 업데이트합니다.

\$ export AWS_CREDENTIAL_FILE=<the file created above>

• Windows에서는 다음 명령을 사용하여 변수를 설정합니다.

C:\> set AWS_CREDENTIAL_FILE=<the file created above>

2. 설정한 것이 제대로 작동하는지 확인하려면 다음 명령을 실행합니다.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 환경 변수

elasticache --help

모든 ElastiCache 명령에 대한 사용법 페이지가 표시됩니다.

환경 변수

환경 변수는 스크립팅, 기본값 구성 또는 기본값 임시 재정의에 유용할 수 있습니다.

AWS_CREDENTIAL_FILE 환경 변수뿐만 아니라 ElastiCache 명령줄 인터페이스와 함께 포함된 API 도구는 대부분 다음 변수를 지원하지 않습니다.

- EC2 REGION 사용할 AWS 리전입니다.
- AWS_ELASTICACHE_URL 서비스 호출에 사용할 URL입니다. EC2_REGION이 지정되어 있거나 -- region 파라미터가 전달되는 경우 다른 리전 엔드포인트를 지정할 필요가 없습니다.

다음 예제에서는 EC2_REGION이라는 환경 변수를 설정하여 API 도구에서 사용하는 리전을 구성하는 방법을 보여줍니다.

Linux, OS X 또는 Unix

\$ export EC2_REGION=us-west-1

Windows

\$ set EC2_REGION=us-west-1

Amazon ElastiCache 오류 메시지

다음 오류 메시지가 Amazon ElastiCache에 의해 반환됩니다. ElastiCache, 기타 AWS 서비스 또는 Redis에 의해 반환되는 다른 오류 메시지를 받을 수 있습니다. ElastiCache가 아닌 소스의 오류 메시지에 대한 설명은 오류 메시지를 생성하는 소스의 설명서를 참조하십시오.

- 클러스터 노드 할당량 초과 (p. 431)
- 고객의 노드 할당량 초과 (p. 432)
- 수동 스냅샷 할당량 초과 (p. 432)
- 부족한 캐시 클러스터 용량 (p. 432)

오류 메시지: 클러스터 노드 할당량이 초과되었습니다. 각 클러스터는 이 리전에서 최대 %n개의 노드를 가질 수 있습니다.

원인: 클러스터를 생성 또는 수정하려고 시도하여 클러스터에 %n개가 넘는 노드가 있습니다.

솔루션: 클러스터에 %n개가 넘는 노드가 있지 않도록 요청을 변경하십시오. 또는 %n 이상의 노드가 필요한 경우 Amazon ElastiCache 노드 요청 양식을 사용하여 요청하십시오.

자세한 내용은 Amazon Web Services 일반 참조의 Amazon ElastiCache 한계를 참조하십시오.

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서 알림

오류 메시지: 고객 노드 할당량이 초과되었습니다. 이 리전에서 최대 %n개의 노드가 있을 수 있습니다. 또는이 리전에서 이미 %s개의 노드 할당량에 도달했습니다.

원인: 클러스터를 생성 또는 수정하려고 시도하여 계정에 이 리전의 모든 클러스터에 대해 %n개가 넘는 노드가 있습니다.

솔루션: 이 계정에 대한 모든 클러스터의 리전에 있는 총 노드가 %n개를 초과하지 않도록 요청을 변경하십시오. 또는 %n 이상의 노드가 필요한 경우 Amazon ElastiCache 노드 요청 양식을 사용하여 요청하십시오.

자세한 내용은 Amazon Web Services 일반 참조의 Amazon ElastiCache 한계를 참조하십시오.

오류 메시지: 24시간 내에 수행된 이 클러스터에 대한 수동 스냅샷의 최대 수가 도달했습니다. 개의 할당량에 도달했거나24시간 내에 수행된 이 노드에 대한 수동 스냅샷의 최대 수가 %n개의 할당량에 도달했습니다.

원인: 24시간의 기간 내에 허용된 수동 스냅샷의 최대 수에 이미 도달했을 때 클러스터의 수동 스냅샷을 수행하려고 시도했습니다.

솔루션: 24시간을 기다린 후 클러스터의 다른 수동 스냅샷을 시도하십시오. 또는 지금 수동 스냅샷을 수행해야 하는 경우 클러스터의 다른 노드와 같이 동일한 데이터가 있는 다른 노드의 스냅샷을 수행하십시오.

오류 메시지: InsufficientCacheClusterCapacity

원인: 현재 AWS에 요청에 대한 서비스를 제공할 수 있을 만큼 온디맨드 용량이 충분하지 않습니다.

솔루션:

- 몇 분 정도 기다린 후 다시 요청을 제출합니다. 용량은 자주 변할 수 있습니다.
- 노드 또는 샤드(노드 그룹) 수가 줄어든 새 요청을 제출하십시오. 예를 들어 단일 요청을 통해 노드 15 개를 시작하는 경우 노드 5개의 요청 3개 또는 노드 1개 대신 요청 15개를 시도합니다.
- 클러스터를 시작하고 있는 경우 가용 영역을 지정하지 않고 새 요청을 제출하십시오.
- 클러스터를 시작하고 있는 경우 이후의 단계에서 확장할 수 있는 다른 노드 유형을 사용하여 새 요청을 제출하십시오. 자세한 내용은 Redis용 ElastiCache 클러스터 조정 (p. 219) 단원을 참조하십시오.

알림

이 주제에서는 관심을 가질 수 있는 ElastiCache 알림을 다룹니다. 알림은 대부분 일시적이며, 솔루션을 찾아 구현할 때까지만 지속되는 상황이나 이벤트입니다. 알림에는 일반적으로 시작 날짜와 해결 날짜가 있으며, 그 이후에는 알림이 더 이상 관련되지 않습니다. 알림은 사용자과 관련이 있거나 관련이 없을 수 있습니다. 클러스터의 성능을 향상하는 구현 지침을 따르는 것이 좋습니다.

알림은 새로운 또는 향상된 ElastiCache 기능을 소개하지 않습니다.

일반 ElastiCache 알림

현재 엔진별로 분류되지 않은 미해결 ElastiCache 알림은 없습니다.

Redis용 ElastiCache 관련 알림

현재는 미해결 Redis용 ElastiCache 알림이 없습니다.

Redis용 ElastiCache 설명서 기록

• API 버전: 2015-02-02

• 설명서 최종 업데이트: 2020년 3월 16일

다음 표에서는 2018년 3월 이후 Redis용 ElastiCache 사용 설명서의 각 릴리스에서 변경된 중요 사항에 대해 설명합니다. 이 설명서에 대한 업데이트 알림을 받으려면 RSS 피드를 구독하면 됩니다.

최신 Redis용 ElastiCache 업데이트

update-history-change	update-history-description	update-history-date
이제 Amazon ElastiCache에서는 T3 표준 캐시 노드가 지원됩니다. (p. 433)	이제 Amazon ElastiCache에서 차세대 범용 순간 확장 가능 T3 표준 캐시 노드를 실행할 수 있습니다. Amazon EC2의 T3 표준 인스턴스는 기준 수준의 CPU 성능과 더불어 누적된 크레딧이 소진될 때까지 언제든 CPU 사용량을 순간 확장할 수 있는 기능을 제공합니다. 자세한 내용은 지원되는 노드 유형 단원을 참조하십시오.	November 12, 2019
ElastiCache 표준 예약 인스턴스 제품: 부분 선결제, 전체 선결제, 선결제 없음. (p. 433)	예약 인스턴스는 인스턴스 유형 및 AWS 리전에 따라 1년 또는 3 년 기간으로 Amazon ElastiCache 인스턴스를 예약할 수 있는 유연 성을 제공합니다. 자세한 내용은 예약 노드로 비용 관리를 참조하 십시오.	January 18, 2019
사용 설명서 재구성 (p. 433)	단일 ElastiCache 사용 설명서 가재구성되어 Redis용 사용 설명서 (Redis용 ElastiCache 사용 설명서)와 Memcached용 사용 설명서 (ElastiCache for Memcached 사용 설명서)가 별도로 존재합니다. AWS CLI 명령 참조: elasticache 단원과 Amazon ElastiCache API 참조의 문서 구조는 그대로 유지됩니다.	April 20, 2018

다음 표에서는 2018년 3월 이전의 Redis용 ElastiCache 사용 설명서에 적용되는 주요 변경 사항에 대해 설명합니다.

변경 사항	설명	변경 날짜
아시아 태평양(오 사카-로컬) 지원.	ElastiCache에서 아시아 태평양(오사카-로컬) 리전에 대한 지원이 추가되었습니다. 아시아 태평양(오사카-로컬) 리전은 현재 단일 가용 영역을 지원하고 있으며 초대를 통해서만 이루어집니다. 자세한 내용은 다음 단원을 참조하십시오.	2018년 2월 12일

변경 사항	설명	변경 날짜
	지원되는 리전 및 엔드포인트 (p. 47)지원되는 노드 유형 (p. 54)	
EU(파리) 지원.	ElastiCache에서 EU(파리) 리전에 대한 지원이 추가되 었습니다. 자세한 내용은 다음 단원을 참조하십시오.	2017년 12월 18일
	지원되는 리전 및 엔드포인트 (p. 47)지원되는 노드 유형 (p. 54)	
중국(닝샤) 리전 지 원	Amazon ElastiCache에서 중국(닝샤) 리전 에 대한 지원 이 추가되었습니다. 자세한 내용은 다음 자료를 참조하 십시오.	2017년 12월 11일
	지원되는 리전 및 엔드포인트 (p. 47)지원되는 노드 유형 (p. 54)	
서비스 연결 역할 지원	ElastiCache의 이번 릴리스에서는 SLR(서비스 연결 역할)에 대한 지원이 추가되었습니다. 자세한 내용은 다음 자료를 참조하십시오.	2017년 12월 7일
	 Amazon ElastiCache에 서비스 연결 역할 사용 (p. 349) 권한 설정(신규 ElastiCache 사용자에게만 해당) (p. 21) 	
R4 노드 유형 지원	이 버전의 ElastiCache에서는 ElastiCache에서 지원하는 모든 AWS 리전에서 R4 노드 유형에 대한 지원이 추가되었습니다. R4 노드 유형을 온디맨드 또는 예약 캐시 노드로 구입할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.	2017년 11월 20일
	지원되는 노드 유형 (p. 54)Redis 노드 유형별 파라미터 (p. 295)	
Redis용 ElastiCache 3.2.10 및 온라인 샤딩 지 원	Amazon은 Redis용 ElastiCache Redis용 ElastiCache 3.2.10에 대한 지원을 추가했습니다. Redis용 ElastiCache에서는 온라인 클러스터 크기 조정을 도입해 클러스터에서 샤드를 추가 또는 제거하고 동시에 수신되는 I/O 요청을 계속해서 처리합니다. 자세한 내용은다음 자료를 참조하십시오.	2017년 11월 9일
	 모범 사례: 온라인 클러스터 크기 조정 (p. 410) Redis(클러스터 모드 활성화됨)를 위한 온라인 리샤 딩 및 샤드 재분배 (p. 243) 	
HIPAA 자격 획득	클러스터에서 암호화가 활성화된 경우 Redis용 ElastiCache 버전 3.2.6은 HIPAA 자격 획득 인증을 받 았습니다. 자세한 내용은 다음 자료를 참조하십시오.	2017년 11월 2일
	HIPAA 자격 획득 (p. 393)Amazon ElastiCache의 데이터 보안 (p. 299)	

변경 사항	설명	변경 날짜
Redis용 ElastiCache 3.2.6 및 암호화 지원	ElastiCache에서 다음 두 가지 암호화 기능이 포함된 Redis용 ElastiCache 3.2.6에 대한 지원이 추가되었습니다. • 전송 중 데이터 암호화는 클러스터 내 노드 사이 또는 클러스터와 애플리케이션 사이와 같이 전송 중인 경우 항상 데이터를 암호화합니다. • 미사용 데이터 암호화는 동기화 및 백업 작업 중 디스크 내 데이터를 암호화합니다. 자세한 내용은 다음 자료를 참조하십시오. • Amazon ElastiCache의 데이터 보안 (p. 299) • 지원되는 Redis용 ElastiCache 버전 (p. 37)	2017년 10월 25일
연결 패턴 항목	Amazon VPC에서 ElastiCache 클러스터에 액세스할 수 있는 여러 가지 패턴을 설명하는 항목이 ElastiCache 설명서에 추가되었습니다. 자세한 내용은 ElastiCache 사용 설명서의 Amazon VPC에 있는 ElastiCache 클러스터에 액세스하기 위한 액세스 패턴 (p. 320) 단원을 참조하십시오.	2017년 4월 24일
자동 장애 조치 테 스트 지원	복제를 지원하는 Redis 클러스터에서 자동 장애 조치를 테스트할 수 있는 지원이 ElastiCache에 추가되었습니 다. 자세한 내용은 다음 자료를 참조하십시오. • 자동 장애 조치가 있는 다중 AZ 테스트 (p. 139)(출처: ElastiCache 사용 설명서). • ElastiCache API 참조의 TestFailover. • AWS CLI 참조의 test-failover.	2017년 4월 4일
Redis 복원 기능 향 상	클러스터 크기 조정과 더불어 향상된 Redis 백업 및 복원 기능이 ElastiCache에 추가되었습니다. 이 기능은 백업 생성에 사용되는 클러스터와 샤드 수가 다른 클러스터에 백업을 복원하도록 지원합니다. (API 및 CLI의 경우이 기능은 서로 다른 샤드 수가 아닌 서로 다른 노드 그룹 수를 복원할 수 있습니다.) 이 업데이트는 서로 다른 Redis 슬롯 구성도 지원합니다. 자세한 내용은 선택적으로 클러스터 크기를 조정하여 백업에서 복원 (p. 208) 단원을 참조하십시오.	2017년 3월 15일
새로운 Redis 메모 리 관리 파라미터	예약된 메모리를 쉽게 관리하는 새 Redis 파라미터 reserved-memory-percent가 ElastiCache에 추가 되었습니다. 모든 버전의 Redis용 ElastiCache에서 이 파라미터를 사용할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오. • 예약된 메모리 관리 (p. 405) • Redis 3.2.4의 새 파라미터 (p. 282)	2017년 3월 15일

변경 사항	설명	변경 날짜
EU 서부(런던) 리 전 지원	EU(런던) 리전에 대한 지원이 ElastiCache에 추가되었습니다. 현재는 노드 유형 T2 및 M4만 지원됩니다. 자세한 내용은 다음 자료를 참조하십시오.	2016년 13월 12일
	 지원되는 리전 및 엔드포인트 (p. 47) 지원되는 노드 유형 (p. 54)	
캐나다(몬트리올) 리전 지원	캐나다(몬트리올) 리전에 대한 지원이 ElastiCache에 추가되었습니다. 현재는 이 AWS 리전에서 노드 유형 M4및 T2만 지원됩니다. 자세한 내용은 다음 단원을 참조하십시오.	2016년 12월 8일
	지원되는 리전 및 엔드포인트 (p. 47)지원되는 노드 유형 (p. 54)	
M4 및 R3 노드 유 형 지원	남아메리카(상파울루) 리전의 M4 및 M4 노드 유형 과 중국(북경) 지역의 M4 노드 유형에 대한 지원이 ElastiCache에 추가되었습니다. 자세한 내용은 다음 자료를 참조하십시오.	2016년 11월 1일
	 지원되는 리전 및 엔드포인트 (p. 47) 지원되는 노드 유형 (p. 54)	
미국 동부 2(오하이 오) 리전 지원	M4, T2 및 R3 노드 유형을 통한 미국 동부(오하이오) 리전(us-east-2) 지원이 ElastiCache에 추가되었습니다. 자세한 내용은 다음 자료를 참조하십시오.	2016년 10월 17일
	지원되는 리전 및 엔드포인트 (p. 47)지원되는 노드 유형 (p. 54)	
Redis 클러스터 지 원	Redis 클러스터(향상)에 대한 지원이 ElastiCache에 추가되었습니다. Redis 클러스터를 사용하는 고객은 최대 15개의 샤드(노드 그룹)에 데이터를 분할할 수 있습니다. 각 샤드는 샤드당 읽기 전용 복제본이 최대 5개인 복제를 지원합니다. Redis 클러스터 자동 장애 조치 시간은 이전 버전의 약 1/4입니다.	2016년 10월 12일
	이 릴리스에는 업계의 용도에 맞게 용어를 사용하도록 다시 고안된 관리 콘솔에 포함되어 있습니다.	
	자세한 내용은 다음 자료를 참조하십시오.	
	 Memcached와 Redis 비교 Redis용 ElastiCache 구성 요소 및 기능 (p. 8) — 노 드, 샤드, 클러스터 및 복제에 대한 단원을 참조하십 시오. 	
	• Redis용 ElastiCache 용어 (p. 14)	

변경 사항	설명	변경 날짜
M4 노드 유형 지원	ElastiCache에서 지원하는 대부분의 리전에서 M4 노드유형 패밀리에 대한 지원이 ElastiCache에 추가되었습니다. M4 노드 유형을 온디맨드 또는 예약 캐시 노드로구입할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오. • 지원되는 노드 유형 (p. 54) • Redis 노드 유형별 파라미터 (p. 295)	2016년 8월 3일
뭄바이 리전 지원	아시아 태평양(뭄바이) 리전에 대한 지원이 ElastiCache에 추가되었습니다. 자세한 내용은 다음 자료를 참조하십시오. • 지원되는 노드 유형 (p. 54) • Redis 노드 유형별 파라미터 (p. 295)	2016년 6월 27일
스냅샷 내보내기	ElastiCache 외부에서 액세스할 수 있도록 Redis 스냅샷을 내보낼 수 있는 기능이 ElastiCache에 추가되었습니다. 자세한 내용은 다음 자료를 참조하십시오. • 백업 내보내기 (p. 201)(출처: Amazon ElastiCache 사용 설명서) • Amazon ElastiCache API 참조의 CopySnapshot	2016년 5월 26일
노드 유형 확장	Redis 노드 유형을 확장하는 기능이 ElastiCache에 추 가되었습니다. 자세한 내용은 Redis용 ElastiCache 클 러스터 조정 (p. 219) 단원을 참조하십시오.	2016년 3월 24일
간편한 엔진 업그레 이드	Redis 캐시 엔진을 쉽게 업그레이드하는 기능이 ElastiCache에 추가되었습니다. 자세한 내용은 엔진 버 전 업그레이드 (p. 45) 단원을 참조하십시오.	2016년 3월 22일
R3 노드 유형 지원	중국(베이징) 리전 및 남아메리카(상파울루) 리전의 R3 노드 유형에 대한 지원이 ElastiCache에 추가되었습니 다. 자세한 내용은 지원되는 노드 유형 (p. 54) 단원을 참 조하십시오.	2016년 3월 16일
Lambda 함수 를 사용하여 ElastiCache 액세 스	Amazon VPC에서 ElastiCache에 액세스하기 위해 Lambda 함수를 구성할 수 있는 자습서가 추가되었습니 다. 자세한 내용은 ElastiCache 자습서 및 동영상 (p. 28) 단원을 참조하십시오.	2016년 2월 12일
Redis 2.8.24 지원	버전 2.8.23부터 추가된 Redis 개선 사항을 포함하여 Redis 버전 2.8.24에 대한 지원이 ElastiCache에 추가되었습니다. 개선 사항에는 버그 수정 및 불량 메모리 액세스 주소의 로깅에 대한 지원이 포함됩니다. 자세한 내용은 다음 자료를 참조하십시오. Redis용 ElastiCache 버전 2.8.24(개선됨) (p. 43) Redis 2.8 출시 정보	2016년 1월 20일
아시아 태평양(서 울) 리전 지원	t2, m3 및 r3 노드 유형을 통한 아시아 태평양(서울)(apnortheast-2) 리전 지원이 ElastiCache에 추가되었습니다.	2016년 1월 6일

변경 사항	설명	변경 날짜
Amazon ElastiCache 콘솔 변경.	새로운 Redis 버전에서는 사용자 환경이 더욱 향상되고 안정적이므로 Redis 버전 2.6.13, 2.8.6 및 2.8.19가 더 이상 ElastiCache 관리 콘솔에 표시되지 않습니다. 기타 옵션과 자세한 내용은 지원되는 Redis용 ElastiCache 버전 (p. 37) 단원을 참조하십시오.	2015년 12월 15일
Redis 2.8.23 지원	버전 2.8.22부터 추가된 Redis 개선 사항을 포함하여 Redis 버전 2.8.23에 대한 지원이 ElastiCache에 추가되었습니다. 개선 사항에는 버그 수정과 새 파라미터 close-on-slave-write에 대한 지원도 포함됩니다. 이 파라미터가 활성화되면 읽기 전용 복제본에 쓰려고시도하는 클라이언트를 연결 해제합니다. 자세한 내용은 Redis용 ElastiCache 버전 2.8.23(개선됨) (p. 43) 단원을 참조하십시오.	2015년 11월 13일
Redis 2.8.22 지원	버전 2.8.21 이후 ElastiCache에 추가된 기능 향상 및 개선 사항과 함께 Redis 버전 2.8.22에 대한 지원이 ElastiCache에 추가되었습니다. 개선 사항에는 다음이 포함됩니다. • 사용 가능한 메모리가 부족하여 forked 저장이 실패할 수 있는 경우 저장을 성공하게 해주는 forkless 저장 프로세스 구현 • 추가 CloudWatch 지표 — SaveInProgress 및 ReplicationBytes. • 부분적 동기화를 사용하기 위해 이제 모든 클러스터에 Redis 파라미터 repl-backlog-size 적용 전체 변경 사항 목록과 자세한 내용은 Redis용 ElastiCache 버전 2.8.22(개선됨) (p. 44) 단원을 참조하십시오. 이 설명서 릴리스에서는 설명서가 재구성되고 ElastiCache 명령줄 인터페이스 설명서가 제거되었습니다. 명령줄에 대해서는 ElastiCache용 AWS 명령줄을 참조하십시오.	2015년 9월 28일
Redis 2.8.21 지원	버전 2.8.19 이후의 Redis 개선 사항을 포함하여 Redis 버전 2.8.21에 대한 지원이 ElastiCache에 추가되었습 니다. 이 Redis 릴리스에는 여러 버그 수정이 포함되어 있습니다. 자세한 내용은 Redis 2.8 출시 정보를 참조하 십시오.	2015년 7월 29일
새로운 주제: AWS 외부에서 ElastiCache 액세 스	AWS 외부에서 ElastiCache 리소스에 액세스하는 방법에 대한 새로운 항목이 추가되었습니다. 자세한 내용은 AWS 외부에서 ElastiCache 리소스 액세스 (p. 107) 단원을 참조하십시오.	2015년 7월 9일

변경 사항	설명	변경 날짜
노드 대체 메시지 추가	ElastiCache에 노드 대체와 관련된 세 가지 메시지(ElastiCache:NodeReplacementScheduled, ElastiCache:NodeReplacementRescheduled 및 ElastiCache:NodeReplacementCanceled)가 추가되었습니다.	2015년 6월 11일
	이에 대한 자세한 내용과 노드 대체를 예약한 경우 취할수 있는 조치는 ElastiCache의 이벤트 알림 및 Amazon SNS (p. 372) 단원을 참조하십시오.	
Redis 버전 2.8.19 지원.	버전 2.8.6 이후의 Redis 개선 사항을 포함하여 Redis 버전 2.8.19에 대한 지원이 ElastiCache에 추가되었습 니다. 여기에는 다음에 대한 지원이 포함되어 있습니다.	2015년 3월 11일
	 Redis 명령 PFADD, PFCOUNT 및 PFMERGE를 사용하는 HyperLogLog 데이터 구조 새로운 명령 ZRANGEBYLEX, ZLEXCOUNT 및 ZREMRANGEBYLEX를 사용하는 사전식 범위 쿼리 다양한 버그 수정이 추가되어 백그라운드 저장 (bgsave) 하위 프로세스가 예기치 않게 중료될 경우 	
	마스터 SYNC를 실패하게 하여 기본 노드에서 복제본 노드로 기한이 지난 데이터를 전송하지 않습니다.	
	structure: the HyperLogLog를 참조하십시오.	
	PFADD, PFCOUNT 및 PFMERGE에 대한 자세한 내용 은 Redis Documentation을 참조하고 HyperLogLog를 클릭하십시오.	
비용 할당 태그 지 원	비용 할당 태그에 대한 지원이 ElastiCache에 추가되었습니다. 자세한 내용은 비용 할당 태그를 사용한 비용 모니터링 (p. 375) 단원을 참조하십시오.	2015년 2월 9일
AWS GovCloud (US-West) 리전 지 원	AWS GovCloud (US-West)(us-gov-west-1) 리전에 대한 지원이 ElastiCache에 추가되었습니다.	2015년 1월 29일
유럽(프랑크푸르 트) 리전 지원	유럽(프랑크푸르트)(eu-central-1) 리전에 대한 지원이 ElastiCache에 추가되었습ㄴ다.	2015년 1월 19일
Redis 복제 그룹의 자동 장애 조치가 지원되는 다중 AZ	기본 노드에서 Redis 복제 그룹의 읽기 전용 복제 본으로 자동 장애 조치되는 다중 AZ에 대한 지원이 ElastiCache에 추가되었습니다. ElastiCache가 복 제 그룹의 상태를 모니터링합니다. 기본 노드가 실패 하면 ElastiCache에서 자동으로 복제본을 기본 노드 로 승격시켜 복제본을 대체합니다. 자세한 내용은 작 동 중지 시간 최소화: 자동 장애 조치가 포함된 다중 AZ (p. 132) 단원을 참조하십시오.	2014년 10월 24일
AWS CloudTrailAPI 호출 의 로깅 지원	AWS CloudTrail을 사용한 모든 ElastiCache API 호출 로깅에 대한 지원이 ElastiCache에 추가되었습니다. 자세한 내용은 AWS CloudTrail을 사용하여 Amazon ElastiCache API 호출 로깅 (p. 424) 단원을 참조하십시 오.	2014년 9월 15일

변경 사항	설명	변경 날짜
새로운 인스턴스 크 기 지원	추가 범용(T2) 인스턴스에 대한 지원이 ElastiCache에 추가되었습니다. 자세한 내용은 파라미터 그룹을 사용해 엔진 파라미터 구성 (p. 260) 단원을 참조하십시오.	2014년 9월 11일
새로운 인스턴스 크 기 지원	추가 범용(M3) 인스턴스 및 메모리 최적화(R3) 인스턴스에 대한 지원이 ElastiCache에 추가되었습니다. 자세한 내용은 파라미터 그룹을 사용해 엔진 파라미터 구성 (p. 260) 단원을 참조하십시오.	2014년 7월 1일
Redis 클러스터의 백업 및 복원	이 릴리스에서는 ElastiCache를 사용하여 고객이 Redis 클러스터 스냅샷을 생성하고 이 스냅샷을 사용하여 새로운 클러스터를 만들 수 있습니다. 백업은 특정한 순간의 클러스터 복사본이며 클러스터 메타데이터와 Redis 캐시의 모든 데이터로 이루어져 있습니다. 백 업은 Amazon S3에 저장되며 고객은 스냅샷의 데이 터를 언제든지 새로운 클러스터에 복원할 수 있습니 다. 자세한 내용은 Redis용 ElastiCache 백업 및 복원 (p. 187) 단원을 참조하십시오.	2014년 4월 24일
Redis 2.8.6	ElastiCache는 Redis 2.6.13을 비롯해 Redis 2.8.6을 지원합니다. Redis 2.8.6을 사용하면 부분적인 재동기화와 언제든지 사용 가능해야 하고 사용자가 정의하는 최소 읽기 전용 복제본 수가 지원되므로 고객은 읽기 전용 복제본의 복원성과 내결함성을 개선할 수 있습니다. 또한 Redis 2.8.6에서는 게시 및 구독이 완벽하게 지원되므로 서버에서 발생하는 이벤트에 대해 고객이 알림을 받을 수 있습니다.	2014년 3월 13일
Redis 캐시 엔진	ElastiCache에서는 Memcached 외에 Redis 캐시 엔진 소프트웨어를 제공합니다. 현재 Redis를 사용하는 고 객은 Redis 스냅샷 파일의 기존 데이터를 사용하여 새 ElastiCache Redis 캐시 클러스터를 "시드"하고 관리형 ElastiCache환경으로 쉽게 마이그레이션할 수 있습니 다.	2013년 9월 3일
	이제 ElastiCache API는 Redis 복제 기능을 지원하기 위해 복제 그룹을 지원합니다. 기본 Redis 캐시 노드를 사용하여 복제 그룹을 생성하고 기본 노드의 캐시 데이터와 자동으로 동기화를 유지하는 하나 이상의 읽기 전용복제본 노드를 추가할 수 있습니다. 읽기 집중형 애플리케이션의 경우 읽기 전용 복제본으로 로드를 덜어 기본노드의 로드를 줄일 수 있습니다. 또한 읽기 전용 복제본은 기본 캐시 노드 실패 시 데이터 손실을 방지할 수있습니다.	
기본 Amazon Virtual Private Cloud(VPC) 지원	이 릴리스에서 ElastiCache는 Amazon Virtual Private Cloud(VPC)와 완전히 통합되었습니다. 새로운 고객의 경우 캐시 클러스터가 기본적으로 Amazon VPC에 생성됩니다. 자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.	2013년 1월 8일

Redis용 Amazon ElastiCache Redis용 ElastiCache 사용 설명서

변경 사항	설명	변경 날짜
Amazon Virtual Private Cloud(VPC) 지원	이 릴리스에서 ElastiCache 클러스터를 Amazon Virtual Private Cloud(VPC)에서 실행할 수 있습니다. 기본적으로 새로운 고객의 캐시 클러스터가 자동으로 Amazon VPC에 생성됩니다. 기존 고객은 자신의 속도로 Amazon VPC로 마이그레이션할 수 있습니다. 자세한 내용은 Amazon VPC 및 ElastiCache 보안 (p. 316) 단원을 참조하십시오.	2012년 20월 12일
새로운 캐시 노드 유형	이 릴리스에서는 캐시 노드 유형 4개를 추가로 제공합 니다.	2012년 11월 13일
예약 캐시 노드	이 릴리스는 예약된 캐시 노드에 대한 지원을 추가합니 다.	2012년 4월 5일
새 가이드	이 설명서는 Amazon ElastiCache 사용 설명서의 첫 번째 배포 버전입니다.	2011년 8월 22일

AWS Glossary

For the latest AWS terminology, see the AWS Glossary in the AWS General Reference.