

# RHA 소개와 사용 방법

2019.09.06

시스템엔지니어링파트 / alan.b

1. RHA (Redis High Availability) 소개
2. 사용 방법 및 주의 사항
3. 이슈 해결 사례
4. 마무리

## RHA (Redis High Availability) 소개

---

## RHA (Redis High Availability)



in-memory data structure store, used as a  
database, cache and message broker



in-memory data structure store, used as a database, cache and message broker

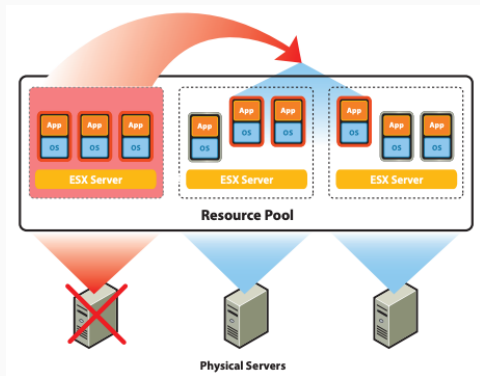
고가용성 Redis

- Single Redis
- Redis HA

- 노드 장애시 1분 이내에 복구
- 데이터 유실 (기본 설정)
- 메모리 12GB 이하 지원
- {name}.dakao.io:6379

# Single Redis

- 노드 장애시 1분 이내에 복구
- 데이터 유실 (기본 설정)
- 메모리 12GB 이하 지원
- {name}.dakao.io:6379

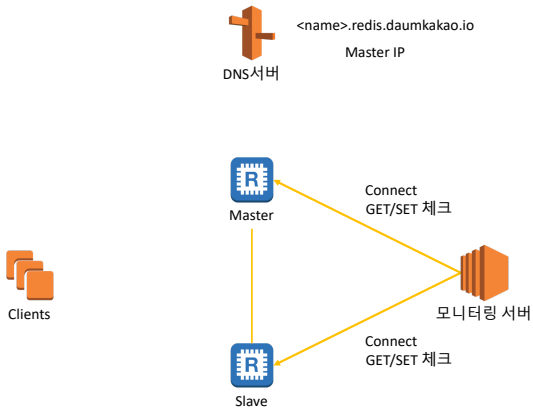


[https://www.vmware.com/pdf/ha\\_datasheet.pdf](https://www.vmware.com/pdf/ha_datasheet.pdf)

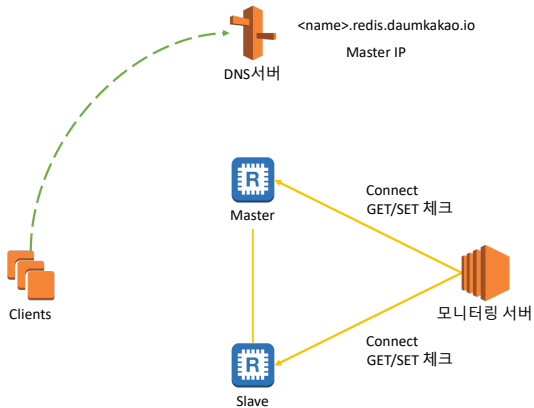
- 마스터 장애시 30초 이내에 페일오버
- 데이터 유지
- 메모리 48GB 이하 지원 (12GB 이하 샤딩 권장)
- Redis replication과 도메인 스위칭 기반
- {name}.redis.daumkakao.io:6379
- DNS cache 비활성 필요  
예) `-Dsun.net.inetaddr.ttl=0`



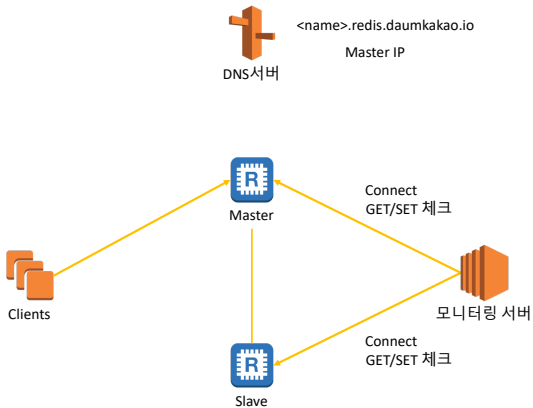
# Redis HA 동작 방식



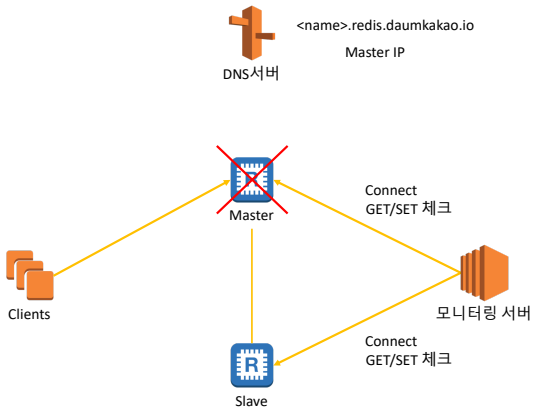
# Redis HA 동작 방식



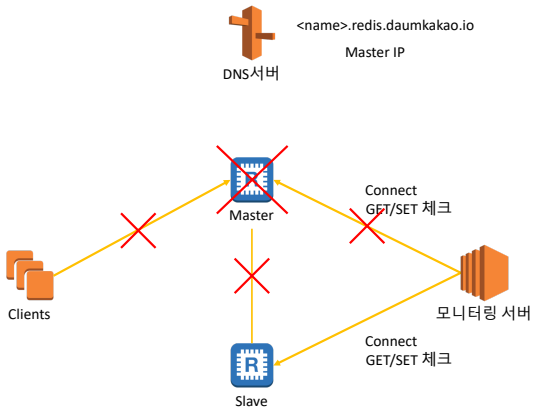
# Redis HA 동작 방식



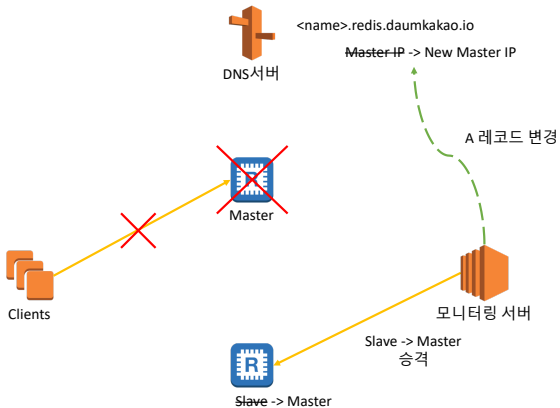
# Redis HA 동작 방식



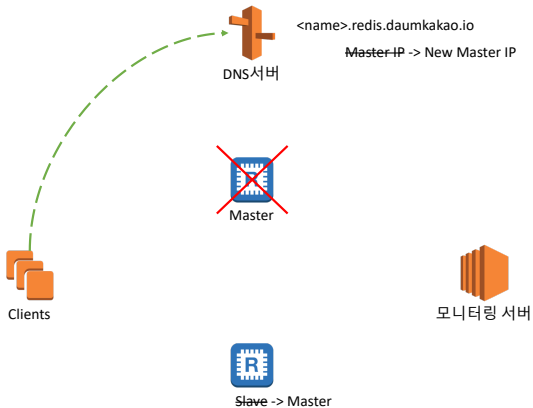
## Redis HA 동작 방식



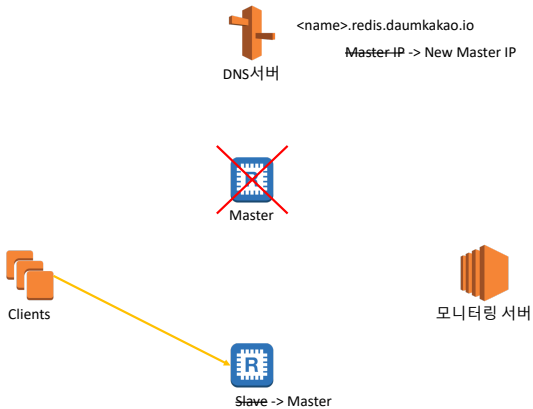
# Redis HA 동작 방식



# Redis HA 동작 방식

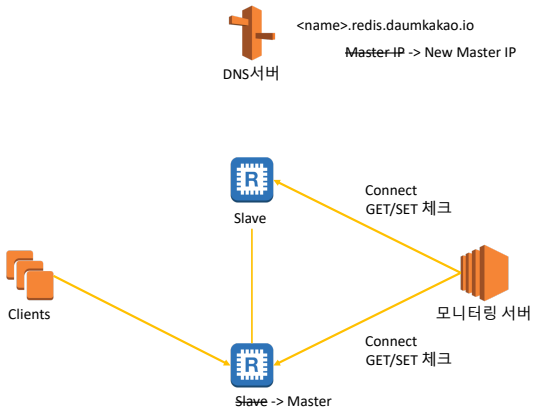


# Redis HA 동작 방식





# Redis HA 동작 방식



RHA 사용 현황 (2019-09-02 기준)

Single Redis

**200**

Single Redis

More info ↻

Redis HA

**570**

Redis HA

More info ↻

## 사용 방법 및 주의 사항

---

# Redis 구성 요청 및 문의

인프라 시스템 업무 아지트에 요청 및 문의    @@inmem

<https://kakao.agit.in/g/238059/wall>

The screenshot shows a Kakao Agit group interface. The header features the group name '인프라 시스템 업무 (요청 및 작업지원)' and a description '# 인프라 담당자 정보 lunatine.c (최민호) · 1524명'. Below the header are buttons for 'Web Push', 'Feeds', 'Joined Agits 7', and 'Group setting'. The main content area has a navigation bar with 'All', 'Members', 'Photos', 'Files', 'Schedule', 'Notes', 'Requests', and 'See description'. Below this is a 'Template' section with various request buttons. The 'Redis 구성 요청' button is highlighted with a red border.

**인프라 시스템 업무 (요청 및 작업지원)**  
# 인프라 담당자 정보  
lunatine.c (최민호) · 1524명

Web Push Feeds Joined Agits 7 Group setting

All Members Photos Files Schedule Notes Requests See description

Template Message Schedule Request Poll Manage Templates

서버 장비 요청 장비반납 서버작업 기타요청 일반 도메인 요청 VIP요청 장비변경

krane 기타 요청 Kafka 요청 ElasticSearch 클러스터 요청 Krane 생성 요청 Krane 삭제 요청

Krane 쿼터 증설 요청 **Redis 구성 요청** Krane 소유권이전 RabbitMQ 요청 Arcus / Memcached 구성 요청

GPU Cloud 신청 IPv6 사내 Test SSID 신청 HTTP Content Caching nGrinder [요청] Kage DB Backup

## Redis 구성 요청 템플릿

- 요청 서비스
- Redis 메모리 (3, 6, 12, 24, 36, 48 GB 중 선택)
- 용도   예) 캐시, Pub/Sub
- Redis 이름  
    ⇒ 주소 결정
- 노드 장애시 데이터 유실 허용 여부  
    ⇒ Single Redis 또는 Redis HA 결정
- 클라이언트 호스트명  
    ⇒ 데이터센터 위치 결정
- Eviction policy (기본값 allkeys-lru)

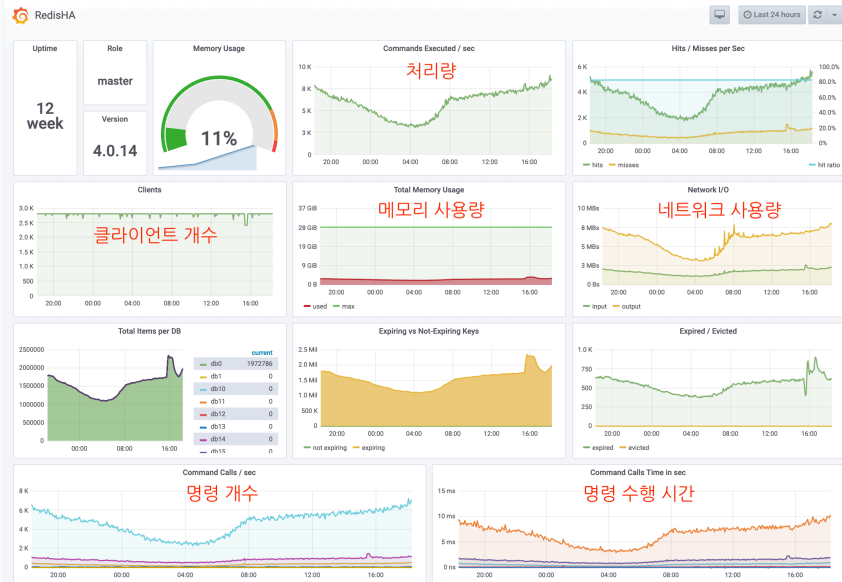
## Redis 구성 요청 템플릿

- 요청 서비스
- Redis 메모리 (3, 6, 12, 24, 36, 48 GB 중 선택)
- 용도    예) 캐시, Pub/Sub
- Redis 이름
  - ⇒ 주소 결정
- 노드 장애시 데이터 유실 허용 여부
  - ⇒ Single Redis 또는 Redis HA 결정
- 클라이언트 호스트명
  - ⇒ 데이터센터 위치 결정
- Eviction policy (기본값 allkeys-lru)

⇒ Redis 구성해서 전달

- 주소
- 대시보드

# 대시보드 (데이터 60일 보관)



### Single Redis

- 메모리 12GB 이하 지원
- 노드 장애시 데이터 유실 (기본 설정)

### Redis HA

- 메모리 48GB 이하 지원 (12GB 샤딩 권장)
- 페일오버시 데이터 유지 (유실 가능성)
- DNS cache 비활성 필요  
예) `-Dsun.net.inetaddr.ttl=0`
- KEYS, FLUSHALL, FLUSHDB 명령 비활성



## 주의 사항

### Single Redis

- 메모리 12GB 이하 지원
- 노드 장애시 데이터 유실 (기본 설정)

### Redis HA

- 메모리 48GB 이하 지원 (12GB 샤딩 권장)
- 페일오버시 데이터 유지 (유실 가능성)
- DNS cache 비활성 필요  
예) `-Dsun.net.inetaddr.ttl=0`
- KEYS, FLUSHALL, FLUSHDB 명령 비활성

### 설정 변경 금지

- 암호 변경 X
- CONFIG SET 명령 사용 X
- 설정 변경 필요하면 문의

## 이슈 해결 사례

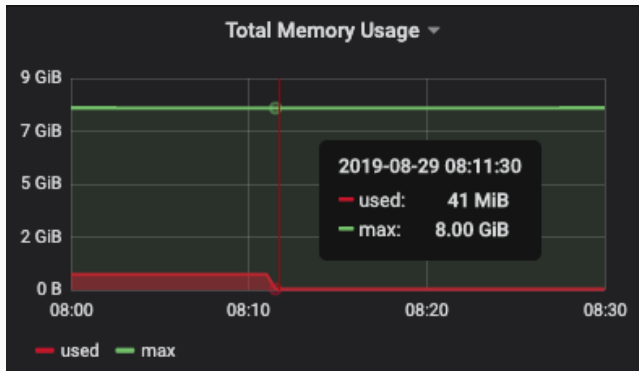
---

- Redis는 한 번에 하나의 요청을 처리 (일부 I/O 제외)
- KEYS, FLUSHALL, FLUSHDB 명령 비활성
- KEYS 대신 SCAN 명령 사용
- SCAN 명령 COUNT 옵션 사용 주의  
예) `SCAN 0 MATCH *PATTERN* COUNT 100000000`  
실행 시간 225446 ms  $\Rightarrow$  페일오버
- DEL 대신 UNLINK 명령 사용 (Redis 4.0 이상)

- 사용자 요청: DEL 명령
- Expired: TTL 만료
- Evicted: maxmemory 초과

## 데이터 삭제로 인한 응답 지연

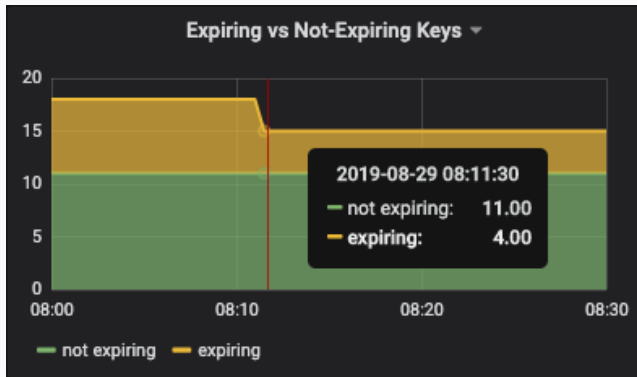
- 사용자 요청: DEL 명령
- Expired: TTL 만료
- Evicted: maxmemory 초과



<https://kakao.agit.in/g/300032888/wall/323123781>

## 데이터 삭제로 인한 응답 지연

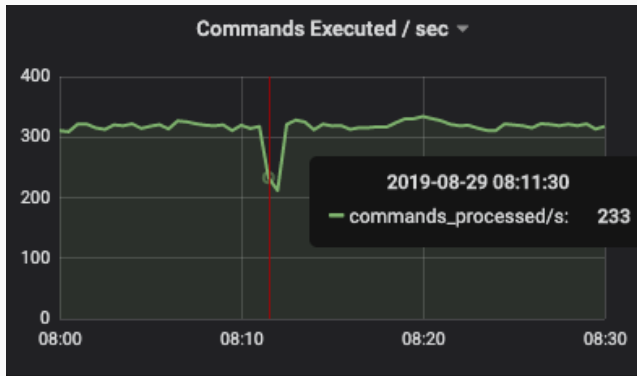
- 사용자 요청: DEL 명령
- Expired: TTL 만료
- Evicted: maxmemory 초과



<https://kakao.agit.in/g/300032888/wall/323123781>

## 데이터 삭제로 인한 응답 지연

- 사용자 요청: DEL 명령
- Expired: TTL 만료
- Evicted: maxmemory 초과



<https://kakao.agit.in/g/300032888/wall/323123781>

- 사용자 요청: DEL 명령
- Expired: TTL 만료
- Evicted: maxmemory 초과
- DEL 대신 UNLINK 명령 사용 (Redis 4.0 이상)
- DEL blocking 이슈  
⇒ Redis 4.0 Lazy freeing 사용



- 왜 사용하나요?

TCP Connection 재사용

⇒ TCP Connection 수립 비용 ↓

⇒ 응답 시간 ↓

- 왜 사용하나요?

TCP Connection 재사용

⇒ TCP Connection 수립 비용 ↓

⇒ 응답 시간 ↓

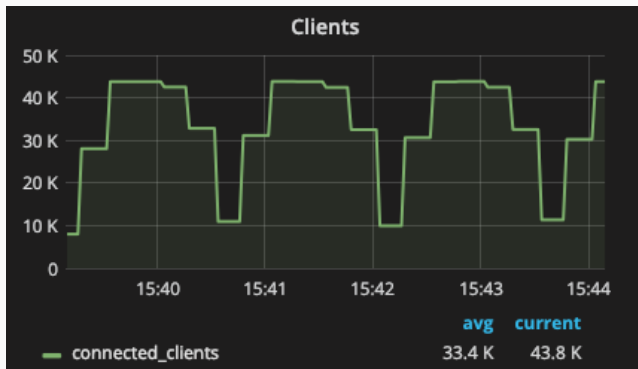
- 사용하지 않는다면?

DNS cache 비활성

⇒ DNS 질의 ↑ 실패 확률 ↑

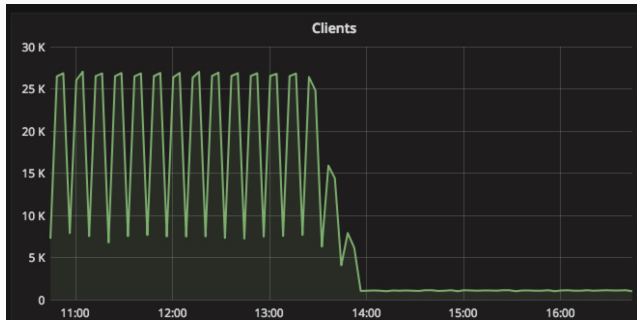
⇒ DNS 서버 부담 ↑

- minIdle=1000
- 클라이언트 30-40개
- 불필요한 TCP Connection ↑
- DNS 질의 실패 빈도 ↑



## JedisPoolConfig minIdle

- `minIdle=1000` → 20
- 클라이언트 30-40개
- 불필요한 TCP Connection ↓
- DNS 질의 실패 빈도 ↓



## JedisPoolConfig 기본 설정

```
1 package redis.clients.jedis;
2
3 import org.apache.commons.pool2.impl.GenericObjectPoolConfig;
4
5 public class JedisPoolConfig extends GenericObjectPoolConfig {
6     public JedisPoolConfig() {
7         // defaults to make your life with connection pool easier :)
8         setTestWhileIdle(true);
9         setMinEvictableIdleTimeMillis(60000);
10        setTimeBetweenEvictionRunsMillis(30000);
11        setNumTestsPerEvictionRun(-1);
12    }
13 }
```

▶ JedisPoolConfig.java

60초 이상 지속된 유휴 연결을 30초마다 검사해서 끊습니다.

### Redis Persistence

- RDB: 백업, 스냅샷
- AOF: Append-only, log rewrite

### Redis Persistence

- RDB: 백업, 스냅샷
- AOF: Append-only, log rewrite

역할	RDB	AOF
Master	disable	disable
Slave	disable	enable → disable

## 고가용성 O, 지속성 X

### Redis Persistence

- RDB: 백업, 스냅샷
- AOF: Append-only, log rewrite

역할	RDB	AOF
Master	disable	disable
Slave	disable	enable → disable

고가용성 O, 지속성 X (기본 설정)

지속성 필요하면 문의해 주세요.



## Swap 사용으로 인한 응답 지연

maxmemory 이상 메모리 사용  $\implies$  Swap

- Memory allocation
- Fragmentation
- Full resynchronization

## Swap 사용으로 인한 응답 지연

maxmemory 이상 메모리 사용  $\implies$  Swap

- Memory allocation
- Fragmentation
- Full resynchronization

$$\frac{\text{maxmemory}}{\text{system memory}} = 75\%$$

Redis 사용 메모리 크기만 고려해 주세요.  
시스템 메모리 사양은 알맞게 결정됩니다.

## Redis 4.0.2 crashed by signal: 11 when dataset size reaches 200+ GB #4493



**Closed** astappiev opened this issue on 30 Nov 2017 · 6 comments



astappiev commented on 30 Nov 2017



The crash occurs every time when the size of instance reach 200+ GB (usually between 210-220GB), it doesn't depend on the amount of data (dbsize).

Also, I tried to split the data into two parts and both parts were successfully processed, so it also not related to certain data. The data consists of (sha-1 hash (20 bytes) -> bytes from multiple long values (8-24 bytes)) pairs.

Also, previously, I tried with Redis 3.2.3 and the same crash occurred (if it can help, [crash report on Redis 3.2.3](#))

Memory test no errors found.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

▶ [antirez/redis#4493](#)

int overflow 문제 ⇒ 해당 버전 업그레이드

마무리

---

## RHA 소개 마무리

- RHA (Redis High Availability): 고가용성 Redis 제공
- 인프라 시스템 업무 아지트 @@inmem 문의
- DNS cache 비활성 필요
- Redis 싱글 스레드 기반
- Connection Pool 사용
- 지속성 필요하면 문의

- Q: Redis Cluster 지원 하나요?

A: 아쉽지만 아직 우선 순위가 낮습니다.

- Q: Dev용 Redis도 지원 하나요?

A: 올해 말 지원 예정

질문해 주세요.

@@inmem

경청해 주셔서 감사합니다.

설문조사 부탁드립니다.