

Redis Failed to start

Advanced key-value store.

Introducción

En algún momento nuestro servidor **Redis** falla, y deja de estar operativo. Reiniciamos pero no arranca y en su lugar muestra un error cuando hacemos un `sudo systemctl status redis-server`

```
jun 02 17:43:33 abkrim-nox systemd[1]: Failed to start Advanced key-value store.
```

Por más que lo intentamos no lo consigue. Veamos que podemos hacer.

Analizando el problema

Systemctl logs

```
journalctl -xeu redis-server.service
[+] The job identifier is 3699 and the job result is done.
jun 02 17:45:14 abkrim-nox systemd[1]: redis-server.service: Start request repeated too quickly.
jun 02 17:45:14 abkrim-nox systemd[1]: redis-server.service: Failed with result 'exit-code'.
[+] Subject: Unit failed
[+] Defined-By: systemd
[+] Support: http://www.ubuntu.com/support
[+]
[+] The unit redis-server.service has entered the 'failed' state with result 'exit-code'.
jun 02 17:45:14 abkrim-nox systemd[1]: Failed to start Advanced key-value store.
```

Bueno ya tenemos una pista pero viendo los logs (bendita bitácora) podemos obtener más información.

```
sudo tail -n100 /var/log/redis/redis-server.log
...
```

```
7471:M 02 Jun 2022 17:48:00.413 * DB loaded from base file appendonly.aof.66.base.rdb: 0.000 seconds
7471:M 02 Jun 2022 17:48:00.893 # Bad file format reading the append only file appendonly.aof.66.incr.aof:
make a backup of your AOF file, then use ./redis-check-aof --fix <filename.manifest>
```

Con esto vemos que nuestra configuración de redis esta configurado para usar una estrategia AOF (append-only file) para evitar perdidas de datos en caso de una terminación brusca (energía, kill - 9,..) que no permita la escritura de los datos activos a disco.

appendonly yes

Y además de esto, nuestro fichero AOF esta corrupto. Así pues hay que recuperarlo.

Reparando el fichero AOF

El comando general es `redis-check-aof -fix` ahora falta encontrar tanto el binario de la utilidad como el fichero.

En mi caso un Ubuntu 22.04 con redis instalado vía repositorio, siendo la versión 6.7

```
➤ sudo ls -liah /var/lib/redis
```

```
total 16K
```

```
17170880 4,0K drwxr-x--- 3 redis redis 4,0K jun  2 20:19 .
16777254 4,0K drwxr-xr-x 94 root  root  4,0K may 31 10:43 ..
17170910 4,0K drwxr-x--- 2 redis redis 4,0K jun  2 18:34 appendonlydir
17170564 4,0K -rw-rw---- 1 redis redis 1,6K jun  2 20:19 dump.rdb
```

```
➤ sudo ls -liah /var/lib/redis/appendonlydir
```

```
total 836K
```

```
17170910 4,0K drwxr-x--- 2 redis redis 4,0K jun  2 18:34 .
17170880 4,0K drwxr-x--- 3 redis redis 4,0K jun  2 20:19 ..
17171350 4,0K -rw-rw---- 1 redis redis 1,6K jun  2 18:34 appendonly.aof.67.base.rdb
17171130 820K -rw-r----- 1 redis redis 815K jun  2 20:21 appendonly.aof.67.incr.aof
17171294 4,0K -rw-r----- 1 redis redis  92 jun  2 18:34 appendonly.aof.manifest
```

Así pues el comando sería

```
sudo /usr/bin/redis-check-aof --fix /var/lib/redis/appendonlydir/appendonly.aof.manifest
```

```
Start checking Multi Part AOF
```

```
Start to check BASE AOF (RDB format).
[offset 0] Checking RDB file appendonly.aof.66.base.rdb
[offset 26] AUX FIELD redis-ver = '7.0.0'
[offset 40] AUX FIELD redis-bits = '64'
[offset 52] AUX FIELD ctime = '1653893946'
[offset 67] AUX FIELD used-mem = '5178616'
[offset 79] AUX FIELD aof-base = '1'
[offset 81] Selecting DB ID 0
[offset 3048] Checksum OK
[offset 3048] \o/ RDB looks OK! \o/
[info] 19 keys read
[info] 9 expires
[info] 9 already expired
RDB preamble is OK, proceeding with AOF tail...
AOF analyzed: filename=appendonly.aof.66.base.rdb, size=3048, ok_up_to=3048, ok_up_to_line=1, diff=0
BASE AOF appendonly.aof.66.base.rdb is valid
Start to check INCR files.
AOF appendonly.aof.66.incr.aof format error
AOF analyzed: filename=appendonly.aof.66.incr.aof, size=64241117, ok_up_to=64238121,
ok_up_to_line=4261202, diff=2996
This will shrink the AOF appendonly.aof.66.incr.aof from 64241117 bytes, with 2996 bytes, to 64238121 bytes
Continue? [y/N]: y
Successfully truncated AOF appendonly.aof.66.incr.aof
All AOF files and manifest are valid
```

Después de esto ya podremos iniciar redis

```
› sudo systemctl restart redis-server
› sudo systemctl status redis-server
● redis-server.service - Advanced key-value store
   Loaded: loaded (/lib/systemd/system/redis-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2022-06-02 17:56:19 CEST; 7min ago
     Docs: http://redis.io/documentation,
           man:redis-server(1)
  Main PID: 12969 (redis-server)
    Status: "Ready to accept connections"
     Tasks: 6 (limit: 38330)
    Memory: 3.9M
       CPU: 1.538s
   CGroup: /system.slice/redis-server.service
```

```
└─12969 "/usr/bin/redis-server 127.0.0.1:6379" "" "" "" "" "" "" "" ""
```

```
jun 02 17:56:18 abkrim-nox systemd[1]: Starting Advanced key-value store...
```

```
jun 02 17:56:19 abkrim-nox systemd[1]: Started Advanced key-value store.
```

Enlaces

- [Redis persistence](#)

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Revision #1

Created 2 June 2022 18:27:09 by Abkrim

Updated 2 June 2022 18:28:07 by Abkrim