

hswaw - Bugless #30

January boston-packets postgres fuckup cleanup, master tracker

03/07/2021 01:47 PM - implr

| | | |
|---|----------|-----------------|
| Status: | Assigned | |
| Priority: | Normal | |
| Assignee: | implr | |
| Category: | hscloud | |
| Description | | |
| On 2020-12-26, postgres was upgraded from 9.x to 12.x. The init script was not updated, so after a boston reboot at 2021-01-27, 9 started again. All data written between the 26th and 2021-01-27 is in the (currently inactive) 12. Affected databases: | | |
| <ul style="list-style-type: none">• kasownik - tentatively okay (members table), detailed transfers diff TBD• mail2sql• mailman, mailman-web #31• ood• owncloud, owncloud_sandbox FIXED in #32• pretix• spam | | |
| After all blocking issues are done, upgrade to 12.x and delete 9. | | |
| Related issues: | | |
| Related to hswaw - Bugless #31: pg fuckup: recover mailman database entries f... | | New |
| Related to hswaw - Bugless #32: pg fuckup: merge owncloud | | Resolved |

History

#1 - 03/07/2021 01:51 PM - implr

- Related to Bugless #31: pg fuckup: recover mailman database entries from pg12 added

#2 - 03/07/2021 01:52 PM - q3k

- Status changed from New to Assigned

- Assignee set to implr

#3 - 03/07/2021 01:56 PM - q3k

- Related to Bugless #32: pg fuckup: merge owncloud added

#4 - 03/13/2021 08:26 PM - q3k

[implr](#): did you figure out when TBD is? I'd like to start merging some stuff.

#5 - 03/14/2021 07:25 PM - implr

The last mtime in 12's pg_wal directory is Jan 27 21:16, so probably this. We could check logs for when the reboot exactly happened, but those have been rotated out already, so would have to check backups and meeeeh doesn't seem worth the effort

#6 - 03/14/2021 07:28 PM - implr

- Description updated

#7 - 03/14/2021 07:37 PM - implr

postgres 12 configured to listen on port +1 (5433), you can connect via unix socket with -U postgres -p 5433

#8 - 07/10/2021 03:23 PM - q3k

For purposes of finding any data that is present in PG12 but missing in PG9 due to rollback we need to also have some well-defined timestamps. These timestamps can be used to determine whether a dated object within PG12 but missing from PG9 is due to a creation-and-less and not having been since removed from PG9.

ie., if an object is found in PG12 it is assumed to have been lost due to rollback if it's missing from PG9 and its timestamp lies between the following two values:

```
>>> datetime(year=2021, month=1, day=25).timestamp()
1611529200.0
>>> datetime(year=2021, month=1, day=28).timestamp()
1611788400.0
```

Or, for your scripts:

```
incident_start = 1611529200 # when pg12 started to run
incident_end = 1611788400 # when we rolled back to pg9
```

I'm assuming these more conservatively than the above estimate. All cases of data lost due to the rollback will be manually verified before any action is taken, so it's better to have false positives than false negatives.

#9 - 07/12/2021 07:29 PM - implr

- *Description updated*

#10 - 07/04/2022 01:08 PM - q3k

- *Category set to hscLOUD*

#11 - 04/10/2023 09:18 PM - implr

9.6 was ancient, out of support, and thus removed. I did a 9.6 -> 14 pg_upgrade (this time taking care to kill 9.6 for good). The lost data in 12 is still there - just think "14" every time you see "9.6" in this issue.