

After upgrading to 1.0.0, postgres-0 is in CrashLoopBackOff state

Issue

- I upgraded my environment from 0.6.1 to 1.0.0 and now postgres-0 is in CrashLoopBackOff state:

```
[user@element2 element-enterprise-installer-1.0.0]$ kubectl get pods -n element-onprem
...
postgres-0                0/1    CrashLoopBackOff   6 (36s
ago)    6m44s
...
```

- Running `kubectl logs -n element-onprem postgres-0` gives me:

```
initdb: error: directory "/var/lib/postgresql/data" exists but is not empty
If you want to create a new database system, either remove or empty
the directory "/var/lib/postgresql/data" or run initdb
with an argument other than "/var/lib/postgresql/data".
```

Environment

- Element Enterprise Installer 1.0.0
- Existing 0.6.1 installation
- Using the installer's built in postgresql database

Resolution

To fix this issue, first read the root cause and issue sections and double check that this is your issue. The resolution is to delete the sts, pvc, and pv for postgres, the empty `data` directory and then re-run the installer. These steps **WILL destroy any existing Postgresql data**, which in the ephemeral case (that this issue describes) is none.

To find where the data directory is, run:

```
kubectl describe pv postgres | grep -i path
```

This will show output similar to:

```
StorageClass:    microk8s-hostpath
Type:            HostPath (bare host directory volume)
Path:           /mnt/data/synapse-postgres
HostPathType:
```

From here, we can see that `/mnt/data/synapse-postgres` is where postgres is trying to initiate the database. Let's take a look at that directory:

```
[user@element2 element-enterprise-installer-1.0.0]$ sudo ls -l /mnt/data/synapse-postgres/
total 0
drwx-----. 2 systemd-coredump input 6 Apr 26 15:13 data
[user@element2 element-enterprise-installer-1.0.0]$ sudo ls -l /mnt/data/synapse-postgres/data
total 0
```

As you can see, we have the `data` directory and it is empty. Make a note of this directory for later.

Now we need to remove the pvc and the pv. If you really do have just an empty `data` directory, there is no need to make a backup. If you have more than `data` in your postgres pv path, you will want to **STOP AND MAKE A BACKUP OF THAT PATH'S CONTENTS**.

Now, to delete the PVC, you will need two terminals. In one terminal, you will run:

```
kubectl delete pvc -n element-onprem postgres
```

You will notice that this command just sits there waiting once run. In another terminal, run this command:

```
kubectl delete pod -n element-onprem postgres-0
```

As soon as the pod is deleted, you should notice that the `kubectl delete pvc` command also completes. At this point, we need to now delete the pv:

```
kubectl delete pv -n element-onprem postgres
```

Now it is time to remove the sts for postgres:

```
kubectl delete sts -n element-onprem postgres
```

Remove the `data` directory:

```
sudo rm -r /mnt/data/synapse-postgres/data
```

Now re-run the installer. Once the installer is re-run, you should have a working postgresql. You should notice a running pod in `kubectl get pods -n element-onprem`:

```
postgres-0                1/1      Running    0                2m11s
```

and your `/mnt/data/synapse-postgres` directory should have entries similar to:

```
drwx-----. 6 systemd-coredump input    54 May  6 10:14 base
drwx-----. 2 systemd-coredump input  4096 May  6 10:15 global
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_commit_ts
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_dynshmem
-rw-----. 1 systemd-coredump input  4782 May  6 10:14 pg_hba.conf
-rw-----. 1 systemd-coredump input  1636 May  6 10:14 pg_ident.conf
drwx-----. 4 systemd-coredump input     68 May  6 10:14 pg_logical
drwx-----. 4 systemd-coredump input    36 May  6 10:14 pg_multixact
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_notify
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_replslot
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_serial
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_snapshots
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_stat
drwx-----. 2 systemd-coredump input    63 May  6 10:15 pg_stat_tmp
drwx-----. 2 systemd-coredump input    18 May  6 10:14 pg_subtrans
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_tblspc
drwx-----. 2 systemd-coredump input     6 May  6 10:14 pg_twophase
-rw-----. 1 systemd-coredump input     3 May  6 10:14 PG_VERSION
drwx-----. 3 systemd-coredump input    60 May  6 10:14 pg_wal
drwx-----. 2 systemd-coredump input    18 May  6 10:14 pg_xact
-rw-----. 1 systemd-coredump input    88 May  6 10:14 postgresql.auto.conf
-rw-----. 1 systemd-coredump input 28156 May  6 10:14 postgresql.conf
-rw-----. 1 systemd-coredump input    36 May  6 10:14 postmaster.opts
-rw-----. 1 systemd-coredump input    94 May  6 10:14 postmaster.pid
```

Finally, restart the synapse pod by doing:

```
kubectl delete pod -n element-onprem instance-synapse-main-0
```

Wait for that pod to restart and be completely running again. Verify with `kubectl get pods -n element-onprem` that you have a line similar to:

```
instance-synapse-main-0          1/1      Running    0          2m36s
```

Root Cause

In 0.6.1, we had a bug which caused the included postgresql database to not get written to disk and thus it did not survive restarts. The bug has been fixed in 1.0.0, however, prior versions of the installer did get as far as writing a `data` directory into the `postgresql` storage set up by microk8s. As such, postgres finds this directory on start up and fails to init a new database with the specific log mentioned in the Issue section.

If you do not have this specific error, please do not run the steps in the Resolution section of this knowledge base solution.

Revision #6

Created 2022-05-06 12:38:17 UTC by Karl Abbott

Updated 2024-11-06 12:49:20 UTC by Kieran Mitchell Lane