Scripting handwork rg. PostgreSQL bug CVE-2017-7547

Motivation

PostgreSQL has just detected to a really bad information disclosure bug, CVE-2017-7547.

Unfortunately upgrading to a fixed version (for Debian see their security-tracker on CVE-2017-7547) is not enough, existing installations need manual work, as described in PostgreSQL's own news article 1772 describes. That howto is not only less then optimal (first half of step 4 should happen before step 3 for easier scripting) there does not seem to be a script yet.

Therefore I decided to create the following scripts ...

Scripted Solution (for 'main' cluster)

pg_fix_usermappings.sql code

For manual execution and the interested here is what our full script (see below) puts in `/tmp/pg_fix_user_mappings.sql` and "executes" on all databases after making additional config changes to and restarting postgres:

pg_fix_user_mappings.sql

```
SET search_path = pg_catalog;

CREATE OR REPLACE VIEW pg_user_mappings AS
  SELECT
    U.oid       AS umid,
    S.oid       AS srvid,
    S.srvname   AS srvname,
    U.umuser    AS umuser,
    CASE WHEN U.umuser = 0 THEN
      'public'
    ELSE
      A.rolname
    END AS usename,
    CASE WHEN
      (U.umuser <> 0 AND A.rolname = current_user AND (pg_has_role(S.srvowner, 'USAGE')
        OR has_server_privilege(S.oid, 'USAGE')))
      OR (U.umuser = 0 AND pg_has_role(S.srvowner, 'USAGE'))
      OR (SELECT rolsuper FROM pg_authid WHERE rolname = current_user)
    THEN U.umoptions
    ELSE NULL END AS umoptions
  FROM pg_user_mapping U
  LEFT JOIN pg_authid A ON (A.oid = U.umuser)
  JOIN pg_foreign_server S ON (U.umserver = S.oid);
```

What happens in `pg_fix_usermappings.sh`

The script `pg_fix_usermappings.sh` performs the following operations:

- Some santiy checks. It's safe to call the script without any parameters
- Create `/tmp/pg_fix_user_mappings.sql` (see above)
- Copy `postgres.conf` to `postgres.conf.bak`
- Patch `postgresql.conf` with `allow_system_table_mods=true`
- Restart postgres
- Sleep 60 seconds, because postgres start asynchronously
- Enable changes to `template0`
- Apply`/tmp/pg_fix_user_mappings.sql` to ALL databases
- Disable changes to `template0`
- Restore `postgresql.conf.bak` to `postgres.conf`
- Restart postgres again

pg_fix_usermappings.sh download & execution

To try to fix your PostgreSQL installation in a debian or similar environment:

- Review`pg_fix_usermappings.sh`
- Review it!

- Download and execute it as user postgres

Overall (several variants, read before execution)

```
# download
#sudo apt-get install ca-certificates
wget https://download.clazzes.org/pg_fix_usermappings/pg_fix_usermappings.sh \
  -O /tmp/pg_fix_usermappings.sh


# make it executable, for user postgres
chmod ugo+rx /tmp/pg_fix_usermappings.sh

# it's safe to call the script without any parameters ...
/tmp/pg_fix_usermappings.sh

# think about version
ls -ld /etc/postgresql/*
dpkg -l |egrep " postgresql-9.[0-9] "

# execute for 9.6 as non root logging the output
export MYPGVER=9.6
( sudo sudo -u postgres /tmp/pg_fix_usermappings.sh ${MYPGVER} ) \
  2>&1 |tee /var/tmp/pg_fix_usermappings_${MYPGVER}.log

# after success maybe document
sudo mv -v /tmp/pg_fix_usermappings.sh /var/tmp/pg_fix_usermappings*.log /var/log/postgresql/
```