Automated user and access management in DBaaS environments

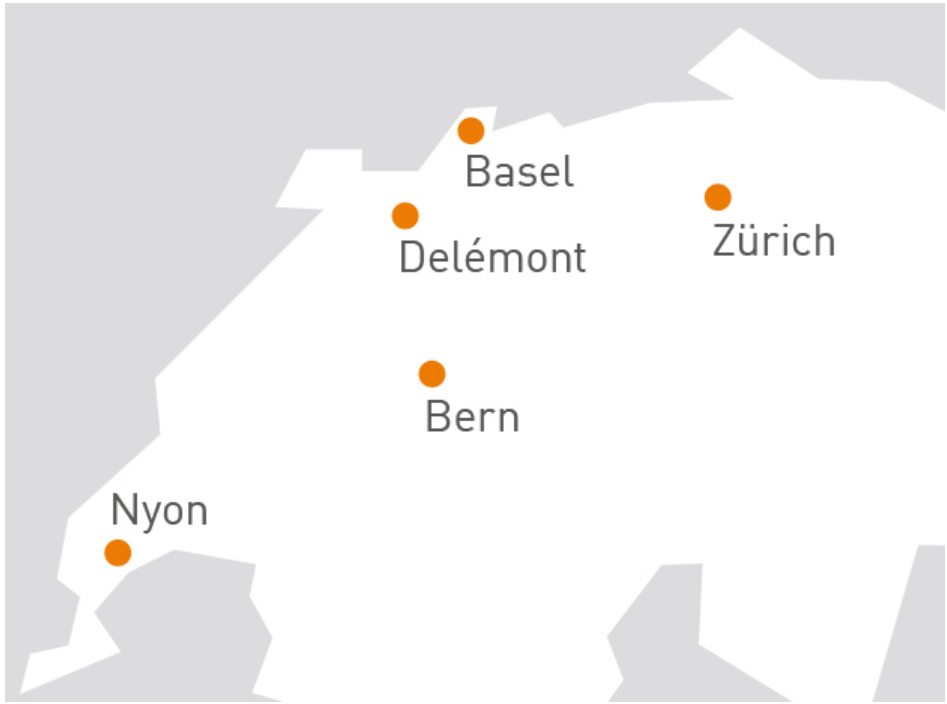
Who we are

The Company

- Founded in 2010
- More than 90 specialists
- Specialized in the Middleware Infrastructure
- The invisible part of IT
- Customers in Switzerland and all over Europe

Our Offer

- Consulting
- Service Level Agreements (SLA)
- Trainings
- License Management



swiss safety center ISO 20000 certified system

swiss safety center ISO 27001 certified system

About me

Karsten Lenz

Open Infrastructure Consultant

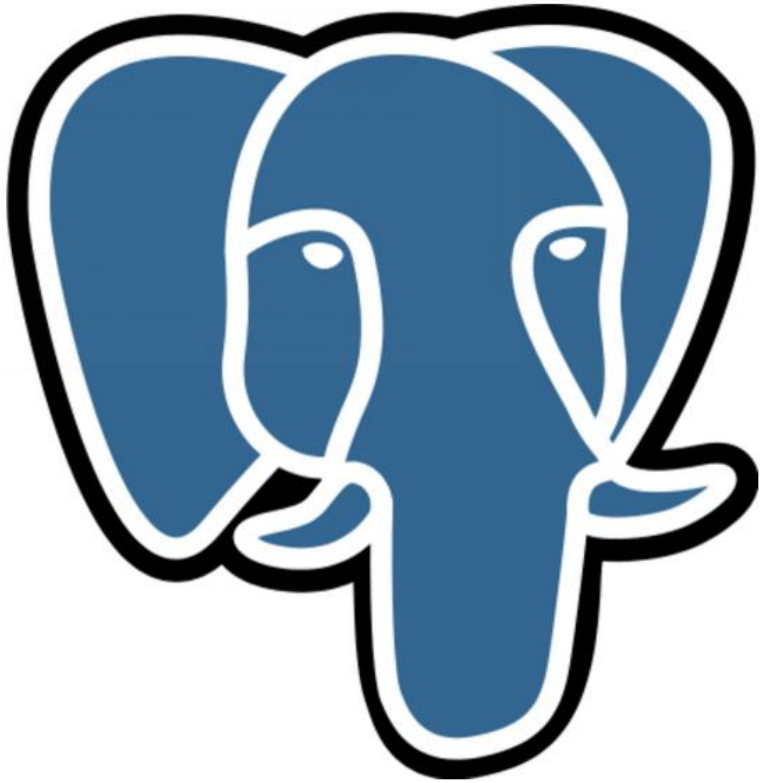
+41 78 799 0146

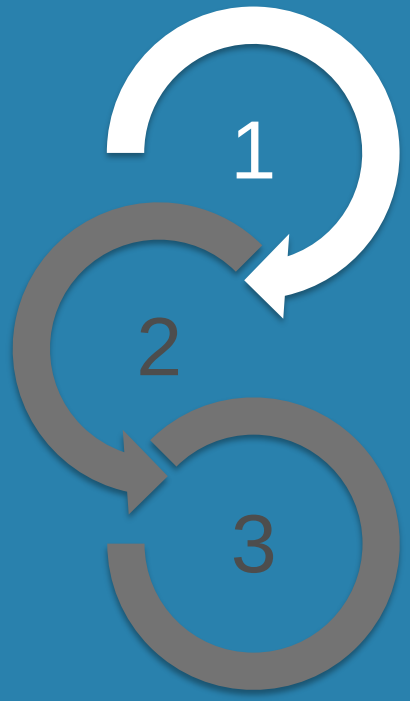
karsten.lenz[at]dbi-services.com

Agenda

- Automated user and access management in DBaaS environments
- What is needed
- How we implment

Automated user and access management in DBaaS environments





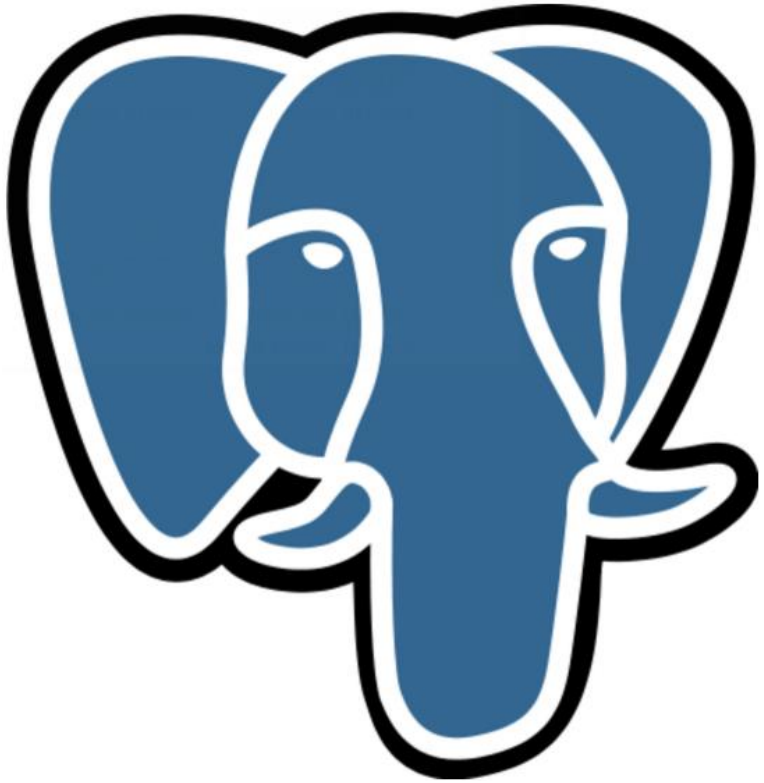
Automated user and access management in DBaaS environments

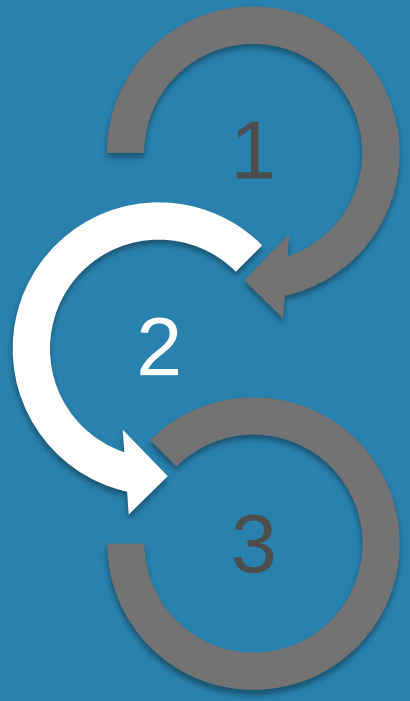
- The task was how to implement a automated user and access mamanment.
- Technical base are virtual machines using Linux (In this case SLES).
- Central usable by Swiss Re developed Web Frontend.
- Use of PotsgreSQL functionality as much as possible.

PostgreSQL default functionality

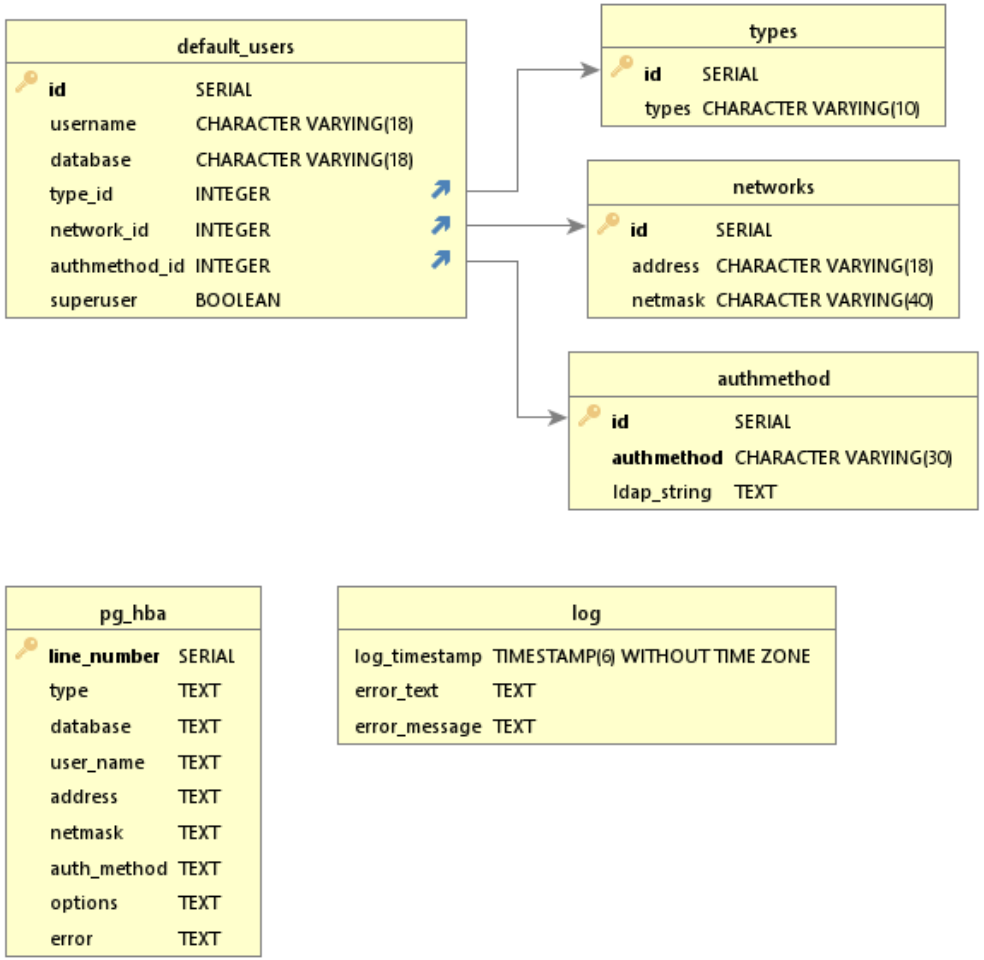
- Using pg_hba_file_rules to read out existing configuration
- Table pg_hba has the structure of pg_hba_file_rules.
- Data is written back using the structure of pg_hba_file_rules.
- Functionality written in plpgsql.

Description





Datamodel


default_users
id SERIAL
username CHARACTER VARYING(18)
database CHARACTER VARYING(18)
type_id INTEGER
network_id INTEGER
authmethod_id INTEGER
superuser BOOLEAN
types
id SERIAL
types CHARACTER VARYING(10)
networks
id SERIAL
address CHARACTER VARYING(18)
netmask CHARACTER VARYING(40)
authmethod
id SERIAL
authmethod CHARACTER VARYING(30)
ldap_string TEXT
pg_hba
line_number SERIAL
type TEXT
database TEXT
user_name TEXT
address TEXT
netmask TEXT
auth_method TEXT
options TEXT
error TEXT
log
log_timestamp TIMESTAMP(6) WITHOUT TIME ZONE
error_text TEXT
error_message TEXT

Description

- default_users are defined users for administration.
- types, allowed types are local, host, hostssl, hostnossl, checked by function.
- networks, definition of allowed networks for connection, checked by function..
- authmethod, scram-sha-256, ldap, peer etc. checked by function.
- log, error logging.
- pg_hba, table to copy pg_hba.conf using pg_hba_file_rules.

Functions

- init_authent, initialization of default administration connectivity .
- set_authent, setting authentication for user and database.
- revoke_authent, revoking a defined access rule for a user and database.
- clean_authent_database, revoking all access rules and users for one database.
- clean_authent_user, revoking all access rules to all databases for one user.

All functions check if the authmethod, the type and the network are configured or not, if not user change or add will not be done!

pg_hba.conf and Table pg_hba

```
maintenance=# select * from procedures.pg_hba order by line_number;
 line_number |  type   |  database   | user_name |   address   |                 netmask                  | auth_method |     options      | error
-------------+---------+-------------+-----------+-------------+------------------------------------------+-------------+------------------+-------
           0 | # TYPE  | DATABASE    | USER      | IP-ADDRESS  | IP-MASK                                  | METHOD      | OPTIONS          |
           1 | local   | all         | all       |             |                                          | peer        |                  |
           2 | host    | all         | all       | 127.0.0.1   | 255.255.255.255                          | scram-sha-256 |                |
           3 | host    | all         | all       | 0.0.0.0     | 0.0.0.0                                  | trust       |                  |
           4 | host    | all         | all       | ::1         | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff  | scram-sha-256 |                |
           5 | local   | replication | all       |             |                                          | peer        |                  |
           6 | host    | replication | all       | 127.0.0.1   | 255.255.255.255                          | scram-sha-256 |                |
           7 | host    | replication | all       | ::1         | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff  | scram-sha-256 |                |
           8 | local   | replication | repmgr    |             |                                          | scram-sha-256 |                |
           9 | hostssl | replication | repmgr    | 127.0.0.1   | 255.255.255.255                          | scram-sha-256 |                |
          10 | hostssl | replication | repmgr    | 192.168.0.0 | 255.255.0.0                              | scram-sha-256 |                |
          11 | local   | repmgrdb    | repmgr    |             |                                          | scram-sha-256 |                |
          12 | hostssl | repmgrdb    | repmgr    | 127.0.0.1   | 255.255.255.255                          | scram-sha-256 |                |
          13 | hostssl | repmgrdb    | repmgr    | 192.168.0.0 | 255.255.0.0                              | scram-sha-256 |                |
          14 | local   | all         | postgres  |             |                                          | peer        | map=postgres-map |
          17 | host    | all         | all       | 0.0.0.0     | 0.0.0.0                                  | reject      |                  |
(16 rows)

maintenance=# \q
[postgres@pgconf ~]$ cat /pgdata/14/data/pg_hba.conf
# TYPE          DATABASE        USER            IP-ADDRESS              IP-MASK                                 METHOD          OPTIONS
local           all             all
host            all             all             127.0.0.1               255.255.255.255                         scram-sha-256
host            all             all             0.0.0.0                 0.0.0.0                                 trust
host            all             all             ::1                     ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff  scram-sha-256
local           replication     all
host            replication     all             127.0.0.1               255.255.255.255                         scram-sha-256
host            replication     all             ::1                     ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff  scram-sha-256
local           replication     repmgr                                                                          scram-sha-256
hostssl         replication     repmgr          127.0.0.1               255.255.255.255                         scram-sha-256
hostssl         replication     repmgr          192.168.0.0             255.255.0.0                             scram-sha-256
local           repmgrdb        repmgr                                                                          scram-sha-256
hostssl         repmgrdb        repmgr          127.0.0.1               255.255.255.255                         scram-sha-256
hostssl         repmgrdb        repmgr          192.168.0.0             255.255.0.0                             scram-sha-256
local           all             postgres                                                                        peer            map=postgres-map
host            all             all             0.0.0.0                 0.0.0.0                                 reject
[postgres@pgconf ~]$
```

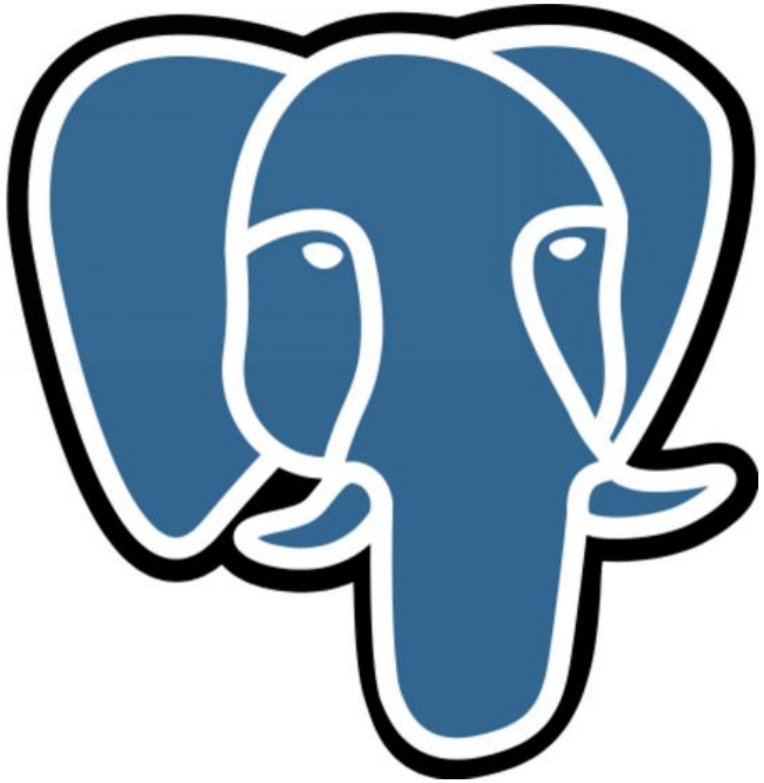
Coding in plpsql

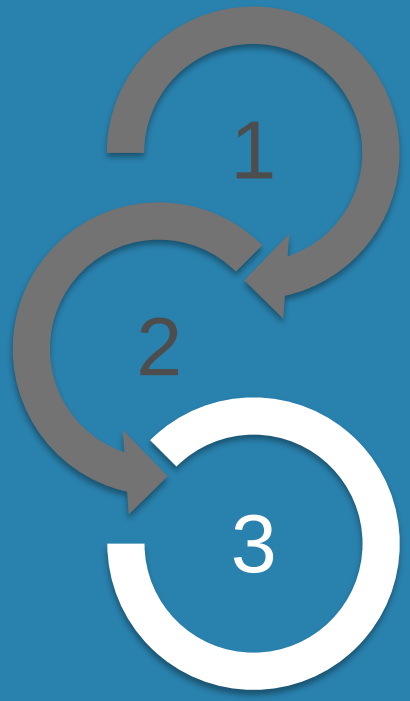
```
CREATE OR REPLACE FUNCTION "procedures"."init_authent" ()  RETURNS integer
  VOLATILE
  SECURITY DEFINER
AS $body$
--by Karsten Lenz dbi services SA 2021.03.04
declare
        ------------------------
        --  define variables   --
        ------------------------
        _query text;
        _type varchar(18);
        _type_id int;
        _address_id int;
        _authmethod_id int;
        _database varchar(50);
        _user_name varchar(18);
        _auth_method varchar(25);
        _address varchar(18);
        _netmask varchar(18);
        _ldap_string text;
        _log_timestamp timestamp := now();
        _error_text text;
        _log_table varchar(25) = 'procedures.log';
        _error_message text;
        _user_exists text;
        _pg_hba varchar(50) := setting from pg_settings where name like '%hba%';
        _check int = 0;
        _return int = 0;
        _count int = 0;
        _id int = 0;
        _position int;
        _reject_id int;
        _max_line int;
        _exist text;
        _superuser boolean;

BEGIN
        --lock table in exclusive lock
        _query = 'lock table procedures.pg_hba IN ACCESS EXCLUSIVE MODE';
        raise notice 'lock query : %',_query;
        begin
                execute _query;
                        exception when others then
                        begin
                                --log timestamp on error
                                _log_timestamp := now();
                                --error message
                                _error_text = ''||_query||' failed. Errorcode = ' || SQLSTATE || ' Message = ' || SQLERRM || '';
                                --raise warning
                                raise warning 'ERROR: %', _error_text;
                                --error message
                                _error_message = '' || SQLSTATE || ' Message = ' || SQLERRM || '';
                                --build log query
                                _query = 'insert into '||_log_table||' (log_timestamp, error_text, error_message) values ('''||_log_timestamp||''','''||_error_text||''','''||_error_message||''');';
                                --raise query
                                raise notice 'insert log query : %',_query;
                                --execute query
                                execute _query;
                                return 1; --lock table faild
                        end;
        end;

        --truncate table for reload
        _query = 'truncate table procedures.pg_hba;';
        raise notice 'truncate table pg_hba : %',_query;
        begin
                execute _query;
                        exception when others then
                        begin
                                --log timestamp on error
                                _log_timestamp := now();
                                --error message
                                _error_text = ''||_query||' failed. Errorcode = ' || SQLSTATE || ' Message = ' || SQLERRM || '';
                                --raise warning
```

Error handling

- Each error is logged in log table.
- Return codes are given back to web frontend for error handling.
- Table pg_hba is exclusive locked that only one change can be done at time.

Demo





Creating user kle for all databases using ident = superuser

```
maintenance=# select procedures.set_authent ('hostssl', 'all', 'kle', '192.168.198.0', 'ident', true);
NOTICE:  type      : hostssl
NOTICE:  database  : all
NOTICE:  user      : kle
NOTICE:  network   : 192.168.198.0
NOTICE:  method    : ident
NOTICE:  address   : 192.168.198.0
NOTICE:  netmask   : 255.255.255.255
NOTICE:  path pg_hba.conf : /pgdata/14/data/pg_hba.conf
NOTICE:  lock query : lock table procedures.pg_hba IN ACCESS EXCLUSIVE MODE
NOTICE:  truncate query : truncate table procedures.pg_hba;
NOTICE:  reload query : insert into  procedures.pg_hba (TYPE, DATABASE, user_name, address, netmask, auth_method, OPTIONS, error) (SELECT type, database, user_name, coalesce(address,''), coa
lesce(netmask,''), auth_method, coalesce(options,''), error FROM procedures.pg_hba_file_read) order by line_number;
NOTICE:  position of reject : 15
NOTICE:  add data line query : insert into procedures.pg_hba (line_number, type, database, user_name, address, netmask, auth_method, options) VALUES (0, '# TYPE', 'DATABASE', 'USER', 'IP-ADD
RESS', 'IP-MASK', 'METHOD','OPTIONS');
NOTICE:  add data line query : insert into procedures.pg_hba (type, database, user_name, address, netmask, auth_method, options) values ('hostssl','all','kle','192.168.198.0','255.255.255.25
5','ident','');
NOTICE:  add data line query : insert into procedures.pg_hba (type, database, user_name, address, netmask, auth_method) VALUES ('host', 'all', 'all', '0.0.0.0', '0.0.0.0', 'reject');
NOTICE:  write pg_hba.conf query : copy (SELECT FORMAT('%-8s',type), FORMAT('%-15s',database), FORMAT('%-15s',user_name), FORMAT('%-20s',address), FORMAT('%-40s',netmask), FORMAT('%-15s',aut
h_method), options from procedures.pg_hba order by line_number) to '/pgdata/14/data/pg_hba.conf' WITH (NULL'');
NOTICE:  user exits : 0
NOTICE:  create user query : create user kle;
NOTICE:  alter user query : alter user kle with superuser;
 set_authent
-------------
           0
(1 row)

maintenance=#
```

Again pg_hba.conf and table pg_hba with user kle

```
maintenance=# select * from procedures.pg_hba order by line_number;
 line_number |  type   |  database   | user_name |    address    |                 netmask                 |  auth_method  |     options      | error
-------------+---------+-------------+-----------+---------------+-----------------------------------------+---------------+------------------+-------
           0 | # TYPE  | DATABASE    | USER      | IP-ADDRESS    | IP-MASK                                 | METHOD        | OPTIONS          |
           1 | local   | all         | all       |               |                                         | peer          |                  |
           2 | host    | all         | all       | 127.0.0.1     | 255.255.255.255                         | scram-sha-256 |                  |
           3 | host    | all         | all       | 0.0.0.0       | 0.0.0.0                                 | trust         |                  |
           4 | host    | all         | all       | ::1           | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff | scram-sha-256 |                  |
           5 | local   | replication | all       |               |                                         | peer          |                  |
           6 | host    | replication | all       | 127.0.0.1     | 255.255.255.255                         | scram-sha-256 |                  |
           7 | host    | replication | all       | ::1           | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff | scram-sha-256 |                  |
           8 | local   | replication | repmgr    |               |                                         | scram-sha-256 |                  |
           9 | hostssl | replication | repmgr    | 127.0.0.1     | 255.255.255.255                         | scram-sha-256 |                  |
          10 | hostssl | replication | repmgr    | 192.168.0.0   | 255.255.0.0                             | scram-sha-256 |                  |
          11 | local   | repmgrdb    | repmgr    |               |                                         | scram-sha-256 |                  |
          12 | hostssl | repmgrdb    | repmgr    | 127.0.0.1     | 255.255.255.255                         | scram-sha-256 |                  |
          13 | hostssl | repmgrdb    | repmgr    | 192.168.0.0   | 255.255.0.0                             | scram-sha-256 |                  |
          14 | local   | all         | postgres  |               |                                         | peer          | map=postgres-map |
          16 | hostssl | all         | kle       | 192.168.198.0 | 255.255.255.255                         | ident         |                  |
          17 | host    | all         | all       | 0.0.0.0       | 0.0.0.0                                 | reject        |                  |
(17 rows)

maintenance=# \q
[postgres@pgconf ~]$ cat /pgdata/14/data/pg_hba.conf
# TYPE          DATABASE        USER            IP-ADDRESS              IP-MASK                                 METHOD          OPTIONS
local           all             all                                                                             peer
host            all             all             127.0.0.1               255.255.255.255                         scram-sha-256
host            all             all             0.0.0.0                 0.0.0.0                                 trust
host            all             all             ::1                     ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff scram-sha-256
local           replication     all                                                                             peer
host            replication     all             127.0.0.1               255.255.255.255                         scram-sha-256
host            replication     all             ::1                     ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff scram-sha-256
local           replication     repmgr                                                                          scram-sha-256
hostssl         replication     repmgr          127.0.0.1               255.255.255.255                         scram-sha-256
hostssl         replication     repmgr          192.168.0.0             255.255.0.0                             scram-sha-256
local           repmgrdb        repmgr                                                                          scram-sha-256
hostssl         repmgrdb        repmgr          127.0.0.1               255.255.255.255                         scram-sha-256
hostssl         repmgrdb        repmgr          192.168.0.0             255.255.0.0                             scram-sha-256
local           all             postgres                                                                        peer            map=postgres-map
hostssl         all             kle             192.168.198.0           255.255.255.255                         ident
host            all             all             0.0.0.0                 0.0.0.0                                 reject
[postgres@pgconf ~]$
```

Cleaning user kle for all databases using ident = superuser

```
maintenance=# select procedures.clean_authent('kle', true);
NOTICE:  user      : kle
NOTICE:  complete    : t
NOTICE:  path pg_hba.conf : /pgdata/14/data/pg_hba.conf
NOTICE:  lock query : lock table procedures.pg_hba IN ACCESS EXCLUSIVE MODE
NOTICE:  reload query : insert into  procedures.pg_hba (TYPE, DATABASE, user_name, address, netmask, auth_method, OPTIONS, error) (SELECT type, database, user_name, coalesce(address,''), coa
lesce(netmask,''), auth_method, coalesce(options,''), error FROM procedures.pg_hba_file_read) order by line_number;
NOTICE:  reject_id : 16
NOTICE:  delete reject line query : delete from procedures.pg_hba where line_number = 16;
NOTICE:  id of user record : 15
NOTICE:  add data line query : insert into procedures.pg_hba (type, database, user_name, address, netmask, auth_method, options) VALUES ('host', 'all', 'all', '0.0.0.0', '0.0.0.0', 'reject',
 ' ');
NOTICE:  write pg_hba.conf query : copy (SELECT FORMAT('%-8s',type), FORMAT('%-15s',database), FORMAT('%-15s',user_name), FORMAT('%-20s',address), FORMAT('%-40s',netmask), FORMAT('%-15s',aut
h_method), options from procedures.pg_hba order by line_number) to '/pgdata/14/data/pg_hba.conf' WITH (NULL'');
NOTICE:  drop user query : drop user kle;
 clean_authent
---------------
             0
(1 row)

maintenance=#
```

Again pg_hba.conf and table pg_hba without user kle

```
maintenance=# select * from procedures.pg_hba order by line_number;
 line_number |  type   |  database   | user_name |   address   |                 netmask                 |  auth_method  |     options      | error
-------------+---------+-------------+-----------+-------------+-----------------------------------------+---------------+------------------+-------
           0 | # TYPE  | DATABASE    | USER      | IP-ADDRESS  | IP-MASK                                 | METHOD        | OPTIONS          |
           1 | local   | all         | all       |             |                                         | peer          |                  |
           2 | host    | all         | all       | 127.0.0.1   | 255.255.255.255                         | scram-sha-256 |                  |
           3 | host    | all         | all       | 0.0.0.0     | 0.0.0.0                                 | trust         |                  |
           4 | host    | all         | all       | ::1         | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff | scram-sha-256 |                  |
           5 | local   | replication | all       |             |                                         | peer          |                  |
           6 | host    | replication | all       | 127.0.0.1   | 255.255.255.255                         | scram-sha-256 |                  |
           7 | host    | replication | all       | ::1         | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff | scram-sha-256 |                  |
           8 | local   | replication | repmgr    |             |                                         | scram-sha-256 |                  |
           9 | hostssl | replication | repmgr    | 127.0.0.1   | 255.255.255.255                         | scram-sha-256 |                  |
          10 | hostssl | replication | repmgr    | 192.168.0.0 | 255.255.0.0                             | scram-sha-256 |                  |
          11 | local   | repmgrdb    | repmgr    |             |                                         | scram-sha-256 |                  |
          12 | hostssl | repmgrdb    | repmgr    | 127.0.0.1   | 255.255.255.255                         | scram-sha-256 |                  |
          13 | hostssl | repmgrdb    | repmgr    | 192.168.0.0 | 255.255.0.0                             | scram-sha-256 |                  |
          14 | local   | all         | postgres  |             |                                         | peer          | map=postgres-map |
          17 | host    | all         | all       | 0.0.0.0     | 0.0.0.0                                 | reject        |                  |
(16 rows)

maintenance=# \q
[postgres@pgconf ~]$ cat /pgdata/14/data/pg_hba.conf
# TYPE          DATABASE        USER            IP-ADDRESS              IP-MASK                                 METHOD          OPTIONS
local           all             all
host            all             all             127.0.0.1               255.255.255.255                         scram-sha-256
host            all             all             0.0.0.0                 0.0.0.0                                 trust
host            all             all             ::1                     ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff scram-sha-256
local           replication     all
host            replication     all             127.0.0.1               255.255.255.255                         scram-sha-256
host            replication     all             ::1                     ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff scram-sha-256
local           replication     repmgr                                                                          scram-sha-256
hostssl         replication     repmgr          127.0.0.1               255.255.255.255                         scram-sha-256
hostssl         replication     repmgr          192.168.0.0             255.255.0.0                             scram-sha-256
local           repmgrdb        repmgr                                                                          scram-sha-256
hostssl         repmgrdb        repmgr          127.0.0.1               255.255.255.255                         scram-sha-256
hostssl         repmgrdb        repmgr          192.168.0.0             255.255.0.0                             scram-sha-256
local           all             postgres                                                                        peer            map=postgres-map
host            all             all             0.0.0.0                 0.0.0.0                                 reject
[postgres@pgconf ~]$
```

Some publications from my side

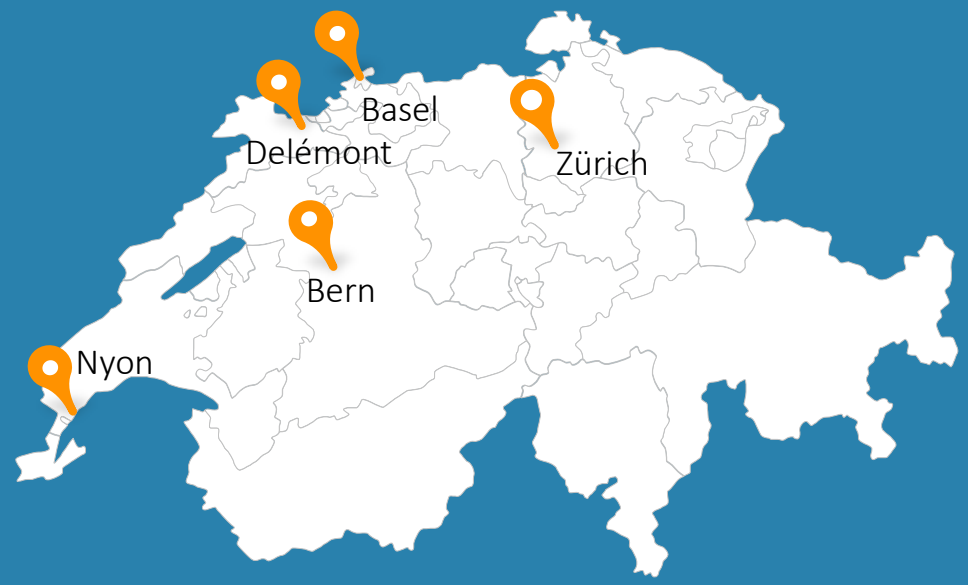
- https://blog.dbi-services.com/recurring-postgresql-installations-using-rhel-8-and-clones/
- https://www.heise.de/ratgeber/PostgreSQL-installieren-mit-den-Community-Paketen-4877556.html

Question from my side

We would like to give the coding for the base installation (shell scripts) and the functionaily we have presented here back to the community if there is any interest.

Any questions?

Please do ask!



We would love to boost your IT-Infrastructure

How about you?