

Roteiro de Instalação do PostgreSQL 14 no Linux CentOS 7

MODIFICAÇÕES NESTE DOCUMENTO

Item: Criação do Documento

Alteração/Inclusão: todo o documento

Responsável: Gilberto A. Oliveira

Data: 20/04/2022

Item: Alterar o CHARSET

Alteração/Inclusão: inclusão do item alteração do CHARSET

Responsável: Gilberto A. Oliveira

Data: 10/05/2022

Índice

Objetivo.....	4
Instalação do PostgreSQL 14.....	4
Formato de DATA HORA.....	8
Alteração do CHARSET.....	9
Pacote de desenvolvimento PGSQLDEV-14.....	10
Instalação do PGSQLDEV-14.....	10
Instalação do LOGIN_HOOK.....	12
Instalação do PG_VARIABLES.....	15
Validação da Instalação.....	17
Criação de usuários.....	17

Objetivo

O objetivo desse roteiro é servir como guia (howto) para:

- Instalação do sistema gerenciador de banco de dados PostgreSQL 14 em um sistema operacional Linux CentOS 7.
- Instalação da extensão LOGIN_HOOK (trigger de logon)
- Instalação da extensão PG_VARIABLES (variável global)

O leitor desse documento deve saber que, neste guia não temos a preocupação de explicar os fundamentos dos comandos apresentados e nem as razões que orientaram as escolhas aqui destacadas.

Alerta: Este documento não tem a pretensão de substituir as documentações oficiais dos sistemas abordados. Recomenda-se as documentações oficiais do PostgreSQL 14 (<https://www.postgresql.org/docs/14/index.html>) e Linux CentOS 7 (<https://docs.centos.org/en-US/docs/>) para exaurir dúvidas e/ou fundamentos que não estejam cobertos por esse roteiro.

Instalação do PostgreSQL 14

O mantenedor do PostgreSQL oferece em seu site <https://www.postgresql.org/download/linux/redhat/> a opção de indicar quais serão os comandos necessários para instalação a partir da escolha da versão e do sistema operacional.

To use the PostgreSQL Yum Repository, follow these steps:

1. Select version:

2. Select platform:

3. Select architecture:

4. Copy, paste and run the relevant parts of the setup script:

Neste documento, escolhemos (setas vermelhas) a versão 14, plataforma (S.O.) CentOS 7 em uma arquitetura de software de 64 bits.

1. Select version:
→ 14

2. Select platform:
→ Red Hat Enterprise, CentOS, Scientific or Oracle version 7

3. Select architecture:
→ x86_64

4. Copy, paste and run the relevant parts of the setup script:

```
# Install the repository RPM:
sudo yum install -y https://download.postgresql.org/pub/repos/yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm

# Install PostgreSQL:
sudo yum install -y postgresql14-server

# Optionally initialize the database and enable automatic start:
sudo /usr/pgsql-14/bin/postgresql-14-setup initdb
sudo systemctl enable postgresql-14
sudo systemctl start postgresql-14
```

Copy Script

Observe que a partir dessas escolhas, o site indica quais comandos deverão ser executados para instalação do banco de dados (seta azul). Bastando que o usuário execute-os:

```
sudo yum install -y https://download.postgresql.org/pub/repos/yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
```

```
# sudo yum install -y https://download.postgresql.org/pub/repos/yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm

Plugins carregados: fastestmirror
pgdg-redhat-repo-latest.noarch.rpm | 8.1 kB 00:00:00
...
...
...
Instalados:
  pgdg-redhat-repo.noarch 0:42.0-24

Concluído!
#
```

```
sudo yum install -y postgresql14-server
```

```
# sudo yum install -y postgresql14-server
Plugins carregados: fastestmirror
Determining fastest mirrors
...
...
...
Instalados:
  postgresql14-server.x86_64 0:14.2-1PGDG.rhel7

Dependência(s) instalada(s):
  libicu.x86_64 0:50.2-4.el7_7 postgresql14.x86_64 0:14.2-1PGDG.rhel7
  postgresql14-libs.x86_64 0:14.2-1PGDG.rhel7

Concluído!
#
```

Caso seja necessário mudar o local de armazenagem dos dados.
(<https://desenvolvedorinteroperavel.wordpress.com/2017/10/29/centos-7-postgresql-instale-o-banco-de-dados-e-use-uma-localizacao-diferente-da-padrao/>)

```
[gilberto@testeostgresql ~]$ systemctl status postgresql-14
● postgresql-14.service - PostgreSQL 14 database server
   Loaded: loaded (/usr/lib/systemd/system/postgresql-14.service; enabled; vendor preset: disabled)
   Active: active (running) since Sex 2022-05-20 09:18:50 -03; 2s ago
     Docs: https://www.postgresql.org/docs/14/static/
   Process: 9094 ExecStartPre=/usr/pgsql-14/bin/postgresql-14-check-db-dir ${PGDATA} (code=exited, status=0/SUCCESS)
   Main PID: 9099 (postmaster)
      CGroup: /system.slice/postgresql-14.service
              └─9099 /usr/pgsql-14/bin/postmaster -D /postgresql/data/data/
                  └─9101 postgres: logger
                      └─9103 postgres: checkpointer
                          └─9104 postgres: background writer
                              └─9105 postgres: walwriter
                                  └─9106 postgres: autovacuum launcher
                                      └─9107 postgres: stats collector
                                          └─9108 postgres: logical replication launcher

Mai 20 09:18:50 testeostgresql systemd[1]: Starting PostgreSQL 14 database server...
Mai 20 09:18:50 testeostgresql postmaster[9099]: 2022-05-20 09:18:50.440 -03 [9099] LOG:  redirecting log...cess
Mai 20 09:18:50 testeostgresql postmaster[9099]: 2022-05-20 09:18:50.440 -03 [9099] HINT:  Future log out...og".
Mai 20 09:18:50 testeostgresql systemd[1]: Started PostgreSQL 14 database server.
Hint: Some lines were ellipsized, use -l to show in full.
[gilberto@testeostgresql ~]$ sudo su - postgres
Último login:Sex Mai 20 09:12:40 -03 2022em pts/0
-bash-4.2$ psql
psql (14.3)
Type "help" for help.

postgres=# SHOW data_directory;
 data_directory
-----
 /postgresql/data/data  (1 row)

postgres=#
```

```
sudo /usr/pgsql-14/bin/postgresql-14-setup initdb
```

```
# sudo /usr/pgsql-14/bin/postgresql-14-setup initdb
Initializing database ... OK
#
```

```
sudo systemctl enable postgresql-14
```

```
# sudo systemctl enable postgresql-14
Created symlink from /etc/systemd/system/multi-user.target.wants/postgresql-14.service to /usr/lib/systemd/system/postgresql-14.service.
#
```

```
sudo systemctl start postgresql-14
```

```
# sudo systemctl start postgresql-14
#
```

Para verificar se o status do banco de dados

```
sudo systemctl status postgresql-14
```

```
# sudo systemctl status postgresql-14
● postgresql-14.service - PostgreSQL 14 database server
   Loaded: loaded (/usr/lib/systemd/system/postgresql-14.service; enabled; vendor preset: disabled)
   Active: active (running) since Sex 2022-04-22 10:50:12 -03; 1min 25s ago
     Docs: https://www.postgresql.org/docs/14/static/
   Process: 8542 ExecStartPre=/usr/pgsql-14/bin/postgresql-14-check-db-dir ${PGDATA} (code=exited, status=0/SUCCESS)
  Main PID: 8548 (postmaster)
    CGroup: /system.slice/postgresql-14.service
            └─8548 /usr/pgsql-14/bin/postmaster -D /var/lib/pgsql/14/data/
               └─8551 postgres: logger
                  └─8553 postgres: checkpointer
                     └─8554 postgres: background writer
                        └─8555 postgres: walwriter
                           └─8556 postgres: autovacuum launcher
                              └─8557 postgres: stats collector
                                 └─8558 postgres: logical replication launcher

Abr 22 10:50:12 centos73 systemd[1]: Starting PostgreSQL 14 database server...
Abr 22 10:50:12 centos73 postmaster[8548]: 2022-04-22 10:50:12.403 -03 [8548] LOG:  redirecting log output to stderr
Abr 22 10:50:12 centos73 postmaster[8548]: 2022-04-22 10:50:12.403 -03 [8548] HINT:  Future log output will be to stderr.
Abr 22 10:50:12 centos73 systemd[1]: Started PostgreSQL 14 database server.
Hint: Some lines were ellipsized, use -l to show in full.
```

Formato de DATA HORA

Por padrão o postgresql usa o fomato padrão de data como YYYY-MM-DD HH24:MI:SS

```
# su - postgres
-bash-4.2$ psql
psql (14.2)
Type "help" for help.

postgres=# select now();
           now
-----
2022-04-22 10:56:33.216068-03
(1 row)

postgres=#
```

Para tornar o banco de dados compatível com os sistemas que trabalham com o formato de data brasileiro, deve-se alterá-lo para o formato DD/MM/YYYY HH24:MI:SS com o comando:

```
ALTER DATABASE nomedabase SET datestyle TO SQL, DMY;
```

Alterando o formato de data para o banco **postgres**

```
ALTER DATABASE postgres SET datestyle TO SQL, DMY;
```

```
postgres=# ALTER DATABASE postgres SET datestyle TO SQL, DMY;
ALTER DATABASE
postgres=#
```

no **próximo** login o padrão de data já estará no formato correto.

```
# su - postgres
-bash-4.2$ psql
psql (14.2)
Type "help" for help.

postgres=# select now();
           now
-----
22/04/2022 11:01:21.336934 -03
(1 row)

postgres=#
```


Alteração do CHARSET

O postgresql usa o formato UTF-8 como conjunto de caracteres padrão, e isso gera uma incompatibilidade de migração de textos que contém caracteres fora desse escopo. Por exemplo, uma palavra acentuada importada do oracle gera uma palavra com "sujeira" no postgres.

Por exemplo: a palavra **acentuação** originada do oracle (WE8MSWIN1252) escreve **acentuaÃ\$Ã£o** no postgres (UTF-8).

Para verificar qual o CHARSET configurado no postgres use o comando

```
#\l
```

```
postgres=# \l
               Lista dos bancos de dados
  Nome      |  Dono   | Codificação | Collate   | Ctype     | Privilégios de acesso |
-----+-----+-----+-----+-----+-----+
postgres   | postgres | UTF8        | pt_BR.UTF-8 | pt_BR.UTF-8 | =Tc/postgres          +
...        |          |             |             |             | postgres=Ctc/postgres +
```

Para que isso não aconteça é necessário alterar o CHARSET do postgres de modo a deixá-lo compatível com o oracle, ou seja, configurá-lo para LATIN-1 com o comando

ATENÇÃO: O comando abaixo deve ser executado para cada novo banco!!

```
update pg_database set collate = pg_char_to_collate('LATIN1') where datname =
'nomedabase';
```

Alterando o formato de data para o banco **postgres**

```
update pg_database set encoding = pg_char_to_encoding('LATIN1') where datname =
'postgres';
```

```
postgres=# update pg_database set encoding = pg_char_to_encoding('LATIN1') where datname =
'postgres';

UPDATE 1
postgres=#
```

Verificando se o banco foi alterado:

```
postgres=# \l
```

Lista dos bancos de dados					
Nome	Dono	Codificação	Collate	Ctype	Privilégios de acesso
postgres	postgres	LATIN1	pt_BR.UTF-8	pt_BR.UTF-8	=Tc/postgres +
....					postgres=CtC/postgres +

Recomenda-se “reinstalar” o banco de dados para aplicação das alterações.

Pacote de desenvolvimento PGSQLDEV-14

É importante saber que o PostgreSQL é um banco de dados que permite a customização de suas funcionalidades de acordo com a necessidade de seu usuário, desse modo é possível adicionar novas características e/ou funções a ele na forma de extensões que são desenvolvidas em linguagens C, sendo necessário a compilação desses programas antes de atribuí-los ao SGBD. O PostgreSQL fornece um pacote de compilação próprio para cada uma de suas versões PGSQLDEV que deve ser previamente instalado para realização da compilação das extensões.

Neste documento trabalharemos com o PGSQLDEV-14 que é a versão compatível com o PostgreSQL 14.

Instalação do PGSQLDEV-14

O processo está documentado no site https://centos.pkgs.org/7/postgresql-14-x86_64/postgresql14-devel-14.1-1PGDG.rhel7.x86_64.rpm.html

1 - Pré-instalação:

Antes de iniciar de fato a instalação do PGSQLDEV-14 é necessário garantir que o sistema operacional esteja preparado para execução dos comandos e contenha todos os pacotes que serão utilizados.

1.1- Adicionar a pasta bin do postgresql14 no path do sistema.

```
export PATH=/usr/pgsql-14/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
```

para verificar o path

```
echo $PATH
```

1.2 – EPEL-7

```
yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

```
# yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
Plugins carregados: fastestmirror
epel-release-latest-7.noarch.rpm | 15 kB 00:00:00
Examinando /var/tmp/yum-root-snCBBF/epel-release-latest-7.noarch.rpm: epel-release-7-14.noarch
...
...
...
Instalados:
  epel-release.noarch 0:7-14

Concluído!
```

1.3 – llvm5.0-devel

```
yum install llvm5.0-devel
```

```
# yum install llvm5.0-devel
Plugins carregados: fastestmirror
Loading mirror speeds from cached hostfile
epel/x86_64/metalink | 48 kB 00:00:00
...
...
...
Instalados:
  llvm5.0-devel.x86_64 0:5.0.1-7.el7

Dependência(s) instalada(s):
  libedit-devel.x86_64 0:3.0-12.20121213cvs.el7
  llvm5.0.x86_64 0:5.0.1-7.el7
  llvm5.0-libs.x86_64 0:5.0.1-7.el7
  ncurses-devel.x86_64 0:5.9-14.20130511.el7_4

Concluído!
```

1.4 – centos-release-scl-rh

```
yum install centos-release-scl-rh
```

```
# yum install centos-release-scl-rh
Plugins carregados: fastestmirror
Loading mirror speeds from cached hostfile
...
...
...
Instalados:
  centos-release-scl-rh.noarch 0:2-3.el7.centos

Concluído!
```

1.5 -llvm-toolset-7-clang

```
yum install llvm-toolset-7-clang
```

```
# yum install llvm-toolset-7-clang
Plugins carregados: fastestmirror
Loading mirror speeds from cached hostfile
...
...
...
Instalados:
  llvm-toolset-7-clang.x86_64 0:5.0.1-4.el7

Dependência(s) instalada(s):
  audit-libs-python.x86_64 0:2.8.5-4.el7                checkpolicy.x86_64 0:2.5-8.el7
...
...
...

Dependência(s) atualizada(s):
  glibc.x86_64 0:2.17-325.el7_9                glibc-common.x86_64 0:2.17-325.el7_9

Concluído!
```

2- Instalação do postgresql14-devel

```
yum install postgresql14-devel
```

```
# yum install postgresql14-devel
Plugins carregados: fastestmirror
Loading mirror speeds from cached hostfile
...
...
...
Instalados:
  postgresql14-devel.x86_64 0:14.2-1PGDG.rhel7

Dependência(s) instalada(s):
  libicu-devel.x86_64 0:50.2-4.el7_7
```

Instalação do LOGIN_HOOK

A instalação dessa extensão permite ao PostgreSQL a funcionalidade do trigger de logon, ou seja, o disparo de um conjunto de ações no momento em que o usuário se conecta no banco de dados. A documentação e download dos arquivos necessários estão disponíveis em https://github.com/splendiddata/login_hook.

1 – baixar e descompactar o pacote do login_hook em um diretório temporário

2 – Editar o arquivo login_hook.c e excluir o bloco de comandos entre as linhas 161 e 183.

```

PG_CATCH();
{
    // Make sure function login_hook.is_executing_login_hook() will return false ever after
    isExecutingLogin = false;

    AbortCurrentTransaction();
    startedATransaction = false;
    if (superuser())
    {
        ErrorData *edata = CopyErrorData();
        ereport(WARNING,
                ( errcode(edata->sqlerrcode),
                  errmsg("Function login_hook.login() returned with error in database
                        %s.\nPlease resolve the error as only superusers can login now.",
                        dbName),
                  errhint("original message = %s", edata->message)));
    }
    else
    {
        elog(ERROR,
              "Function login_hook.login() returned with error in database %s, only a
              superuser can login",
              dbName);
    }
}
}

```

Isso se faz necessário, para contornar o erro de acesso exclusivo do usuário de sistema.

3 – Certificar que os pacotes gcc e make estão instalados

Antes de executar o próximo passo certifique-se que os aplicativos MAKE e GCC estejam instalados. Caso não estejam instale-os.

```

yum install make
yum install gcc

```

```

# yum install make
Plugins carregados: fastestmirror
Loading mirror speeds from cached hostfile
...
O pacote 1:make-3.82-24.el7.x86_64 já está instalado em sua última versão
Nada a ser feito

# yum install gcc
Plugins carregados: fastestmirror
Loading mirror speeds from cached hostfile
...
...
...
Instalados:
  gcc.x86_64 0:4.8.5-44.el7

Dependência(s) instalada(s):
  cpp.x86_64 0:4.8.5-44.el7

```

4 - acessar o diretório temporário onde a extensão foi salva

```

# cd /tmp/login_hook-master/
#

```

5 – executar os comandos de compilação

make

```
# make
gcc -std=gnu99 -Wall -Wmissing-prototypes -Wpointer-arith -Wdeclaration-after-statement -Werror=vla -
Wendif-labels -Wmissing-format-attribute -Wformat-security -fno-strict-aliasing -fwrapv -fexcess-
precision=standard -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-strong
--param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -fPIC -I. -I./ -I/usr/pgsql-
14/include/server -I/usr/pgsql-14/include/internal -D_GNU_SOURCE -I/usr/include/libxml2
-I/usr/include -c -o login_hook.o login_hook.c
gcc -std=gnu99 -Wall -Wmissing-prototypes -Wpointer-arith -Wdeclaration-after-statement -Werror=vla -
Wendif-labels -Wmissing-format-attribute -Wformat-security -fno-strict-aliasing -fwrapv -fexcess-
precision=standard -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-strong
--param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -fPIC -shared -o login_hook.so
login_hook.o -L/usr/pgsql-14/lib -L/usr/lib64/llvm5.0/lib -L/usr/lib64 -Wl,--as-needed -Wl,-
rpath,'/usr/pgsql-14/lib',--enable-new-dtags
/opt/rh/llvm-toolset-7/root/usr/bin/clang -Wno-ignored-attributes -fno-strict-aliasing -fwrapv -Wno-
unused-command-line-argument -O2 -I. -I./ -I/usr/pgsql-14/include/server
-I/usr/pgsql-14/include/internal -D_GNU_SOURCE -I/usr/include/libxml2 -I/usr/include -flto=thin -
emit-llvm -c -o login_hook.bc login_hook.c
#
```

make install

```
# make install
/bin/mkdir -p '/usr/pgsql-14/lib'
...
...
...
cd '/usr/pgsql-14/lib/bitcode' && /usr/lib64/llvm5.0/bin/llvm-lto -thinlto -thinlto-action=thinlink -
o login_hook.index.bc login_hook/login_hook.bc
#
```

6 – Acrescentar a extensão no arquivo de configuração POSTGRESQL.CONF

6.1 - Para ver o local do arquivo você deve conectar no banco de dados e executar o comando

SHOW config_file;

```
# su - postgres
Último login:Sex Abr 22 10:56:21 -03 2022em pts/0
-bash-4.2$ psql
psql (14.2)
Type "help" for help.

postgres=# SHOW config_file;

               config_file
-----
/var/lib/pgsql/14/data/postgresql.conf
(1 row)

postgres=#
```

6.2 – Editar o arquivo postgresql.conf e acrescentar a linha

session_preload_libraries = 'login_hook'

```
#shared_preload_libraries = ''          # (change requires restart)
# (change requires restart)
session_preload_libraries = 'login_hook' ←
# - Cost-Based Vacuum Delay -
```

6.3 – restart o serviço do PostgreSQL

```
systemctl restart postgresql-14
```

```
# systemctl restart postgresql-14
#
```

6.4 – conectado no banco de dados criar a extensão login_hook

ATENÇÃO: O comando abaixo deve ser executado para cada novo banco!!

```
create extension login_hook;
```

```
# su - postgres
Último login:Sex Abr 22 12:01:19 -03 2022em pts/0
-bash-4.2$ psql
psql (14.2)
Type "help" for help.

postgres=# create extension login_hook;
CREATE EXTENSION
postgres=#
```

Instalação do PG_VARIABLES

A extensão PG_VARIABLES permite ao PostgreSQL definir variáveis globais, ou seja, variáveis que ficam disponíveis em durante a sessão do usuário e desse modo possam ser acessadas sempre que necessário. Da mesma forma, será necessário a compilação e criação da extensão. A documentação e download dos arquivos necessários estão disponíveis em https://github.com/postgrespro/pg_variables.

ATENÇÃO: Antes de continuar se certifique que os cuidados de pré-instalação pgsqlddev-14, make, gcc e path relatados anteriormente tenham sido tomados.

1 - baixar e descompactar o pacote do pg_variables em um diretório temporário

2 – acessar o diretório temporário onde a extensão foi salva

```
# cd /tmp/pg_variables/
#
```

3 - executar os comandos de compilação:

```
make USE_PGXS=1
```

```
# make USE_PGXS=1
gcc -std=gnu99 -Wall -Wmissing-prototypes -Wpointer-arith -Wdeclaration-after-statement -Werror=vla -Wendif-labels -Wmissing-format-attribute -Wformat-security -fno-strict-aliasing -fwrapv -fexcess-
...
...
emit-llvm -c -o pg_variables_record.bc pg_variables_record.c
#
```

```
make USE_PGXS=1 install
```

```
# make USE_PGXS=1 install
/bin/mkdir -p '/usr/pgsql-14/lib'
/bin/mkdir -p '/usr/pgsql-14/share/extension'
...
...
/bin/mkdir -p '/usr/pgsql-14/share/extension'
cd '/usr/pgsql-14/lib/bitcode' && /usr/lib64/llvm5.0/bin/llvm-lto -thinlto -thinlto-action=thinlink -
o pg_variables.index.bc pg_variables/pg_variables.bc pg_variables/pg_variables_record.bc
#
```

4 – conectado no banco de dados criar a extensão pg_variables

ATENÇÃO: O comando abaixo deve ser executado para cada novo banco!!

```
CREATE EXTENSION pg_variables;
```

```
# su - postgres
Último login:Sex Abr 22 12:08:20 -03 2022em pts/0
-bash-4.2$ psql
WARNING: Function login_hook.login() is not invoked because it does not exist in database postgres
psql (14.2)
Type "help" for help.

postgres=# CREATE EXTENSION pg_variables;
CREATE EXTENSION
postgres=#
```

5 - Criação da função login_hook.login

```
CREATE OR REPLACE FUNCTION login_hook.login() RETURNS VOID LANGUAGE PLPGSQL AS $$
DECLARE
    VLR_NRORG INT;
BEGIN
    IF NOT login_hook.is_executing_login_hook()
    THEN
        RAISE EXCEPTION 'The login_hook.login() function should only be
invoked by the login_hook code';
    END IF;

    BEGIN
        -- select que recupera o número da organização
        select case when (substr(current_user, 9) between '0' and '9999') then
cast (substr(current_user,9) as int) else 1 end into VLR_NRORG;
        -- comando que cria a variável global para a sessão;
        SELECT public.pgv_set('nrorg', 'int1', VLR_NRORG);
    END ;
END
$$;
```

6 – Concessão de privilégio de execução para public

```
GRANT EXECUTE ON FUNCTION login_hook.login() TO PUBLIC;
```

```
postgres=# GRANT EXECUTE ON FUNCTION login_hook.login() TO PUBLIC;
GRANT
postgres=#
```


Validação da Instalação

1 – Alterne para o usuário postgres

```
su - postgres
```

```
# su - postgres
Último login: Sex Abr 22 12:14:39 -03 2022 em pts/0
-bash-4.2$
```

2 – Faça uma conexão com o banco de dados

```
psql
```

```
-bash-4.2$ psql
WARNING: transaction left non-empty SPI stack
HINT: Check for missing "SPI_finish" calls.
psql (14.2)
Type "help" for help.

postgres=#
```

3 – Verifique a existência da variável global

```
SELECT pgv_get('nrorg', 'int1', NULL::int);
```

```
postgres=# SELECT pgv_get('nrorg', 'int1', NULL::int);
 pgv_get
-----
      1
(1 row)
```

O resultado esperado é o valor **1**, isto porque a lógica da trigger de logon é retornar o número da organização ou o valor 1 quando não o tiver. Nesse momento conectamos com usuário padrão do banco de dados que é postgres (item 2), ou seja, não tem número de organização.

Criação de usuários

1 – Por padrão, o PostgreSQL vem com o nível de autenticação de usuário por peer

Isso significa que não é necessário o uso de senhas para conectar no banco de dados, faremos a alteração do banco para que ele passe a exigir a senha.

1.1 – adicionar senha para o super usuário postgres

```
ALTER USER postgres ENCRYPTED PASSWORD 'senha';
```

```
postgres=# ALTER USER postgres ENCRYPTED PASSWORD 'senha';
ALTER ROLE
```

1.2 – alterar o arquivo de configuração de segurança do banco de dados pg_hba.conf

```
vi /var/lib/pgsql/14/data/pg_hba.conf
```

Alterar a linha de configuração local de **peer** para o método **md5**

#	TYPE	DATABASE	USER	ADDRESS	METHOD
# "local" is for Unix domain socket connections only					
local	all		all		md5
# IPv4 local connections:					
host	all		all	127.0.0.1/32	ident
#host	all		all	0.0.0.0/0	md5
# IPv6 local connections:					
host	all		all	:::1/128	ident
# Allow replication connections from localhost, by a user with the					
# replication privilege.					
local	replication		all		peer
host	replication		all	127.0.0.1/32	ident
host	replication		all	:::1/128	ident

1.3 – Reiniciar o banco de dados

```
systemctl restart postgresql-14
```

```
# systemctl restart postgresql-14  
#
```

1.4 – para validar a alteração de segurança basta realizar uma conexão com usuário postgres e verificar a solicitação da senha (definida no item 1.1 desta seção)

```
psql
```

```
[root@centos73 pg_variables]# su - postgres  
Último login: Sex Abr 22 12:19:32 -03 2022 em pts/0  
-bash-4.2$ psql  
Password for user postgres: █
```

2 – criação de novos usuários com permissão de criar novos bancos (os parâmetros devem estar em maiúsculo)

```
create user nome_usuario WITH PASSWORD 'senha' CREATEDB;
```

Abaixo estamos criando o usuário **usr_org_1050** com a senha **usr_org_1050**.

```
create user usr_org_1050 WITH PASSWORD 'usr_org_1050' CREATEDB;
```

```
postgres=# create user usr_org_1050 WITH PASSWORD 'usr_org_1050' CREATEDB;  
CREATE ROLE  
postgres=#
```

2.1 – validação do usuário

```
psql -U usr_org_1050 postgres
```

```
-bash-4.2$ psql -U usr_org_1050 postgres
Password for user usr_org_1050:
WARNING:  transaction left non-empty SPI stack
HINT:  Check for missing "SPI_finish" calls.
psql (14.2)
Type "help" for help.

postgres=>
```

Observe que na segunda linha está a solicitação da senha.

2.2 – validação do trigger de logon

```
SELECT pgv_get('nrorg', 'intl', NULL::int);
```

```
postgres=> SELECT pgv_get('nrorg', 'intl', NULL::int);
 pgv_get
-----
      1050
(1 row)

postgres=>
```

O resultado esperado é 1050 que corresponde ao número da organização.