EXPLOITATION POSTGRESQL

USER GUIDE

Table des matières

[I. Introduction 4](#_Toc519179061)

[II. Prerequisites 4](#_Toc519179062)

[III. How to exploit a postgresql database : 4](#_Toc519179063)

[1. Stop/Start a postgresql database 4](#_Toc519179064)

[2. Connect to postgresql database 4](#_Toc519179065)

[a. As Applicatif USER : 4](#_Toc519179066)

[a. As Exploitation USER : 4](#_Toc519179067)

[3. Check status of a postgresql database (standalone or a master/slave): 5](#_Toc519179068)

[4. Do a failover of a postgresql database (slave to master): 5](#_Toc519179069)

[5. How to export a schema: 5](#_Toc519179070)

[6. How to check a SQL file: 5](#_Toc519179071)

[IV. How to exploit a Repmgr cluster : 6](#_Toc519179072)

[1. How to start/stop a Repmgr node : 6](#_Toc519179073)

[2. How to check a Repmgr cluster : 6](#_Toc519179074)

[3. How to start/stop/status a Repmgr witness : 6](#_Toc519179075)

[4. How to do a failover/switchover on a Repmgr cluster : 6](#_Toc519179076)

[V. How to exploit a Pgbouncer node : 7](#_Toc519179077)

[1. How to start/stop/status a Pgbouncer node : 7](#_Toc519179078)

MODIFICATION HISTORY & REVIEWS

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
| 02/05/2017 | V0.1 | EL AABOUDI Karim | Initial version |
| 28/06/2017 | V0.2 | EL AABOUDI Karim | Export schema from adm user |
| 05/10/2017 | V0.3 | EL AABOUDI Karim | Add Repmgr and Pgbouncer exploitation |
| 12/06/2018 | V1 | EL AABOUDI Karim | Add how to check a SQL file |
| 12/07/2018 | V2 | EL AABOUDI Karim | Add how to connect with psql |

# Introduction

This document describes how to exploit a postgresql database using the “pack\_dba” script with user <trig><env>adm.

# Prerequisites

If it’s not already the case (check with 🡺 sudo -l), you need the following sudo privilege:

***(root) NOPASSWD: /bin/su postgres -c /produits/admindb/postgres/public/DBA\_\*.ksh \****

# How to exploit a postgresql database :

## Stop/Start a postgresql database

*/produits/admindb/postgres/public/pack\_dba.ksh EXPLOIT <postgresql\_db\_name> START*

*/produits/admindb/postgres/public/pack\_dba.ksh EXPLOIT <postgresql\_db\_name> STOP*

## Connect to postgresql database

## As Applicatif USER :

. /produits/pgsql/env/env\_*<postgresql\_db\_name>*

*psql -U <APPLICATIF\_USER> -h $DBHOST -d $DBNAME*

*DBNAME#\i /path/to/sqlfile.sql*

## As Exploitation USER :

. /produits/pgsql/env/env\_*<postgresql\_db\_name>*

psql $DBNAME

## Check status of a postgresql database (standalone or a master/slave):

*/produits/admindb/postgres/public/pack\_dba.ksh EXPLOIT <postgresql\_db\_name> STATUS*

In case of a standalone or a master, you will have the below result:

***pg\_ctl: server is running (PID: 107481)***

***/opt/rh/rh-postgresql95/root/usr/bin/postgres***

***Database cluster state: in production***

***Time of latest checkpoint: Tue May 2 16:10:07 2017***

In case of a standby, you will have the below result:

***pg\_ctl: server is running (PID: xxxx)***

***/opt/rh/rh-postgresql95/root/usr/bin/postgres***

***Database cluster state: in archive recovery***

***Time of latest checkpoint: Tue May 2 16:05:07 2017***

You can check the gap between the master and the slave with the time of latest checkpoint.

## Do a failover of a postgresql database (slave to master):

**/!\ WARNING : This command must be used only if the master is broken or unreachable.**

The below command must be run on the slave server to promote it and to become read/write:

*/produits/admindb/postgres/public/pack\_dba.ksh EXPLOIT <postgresql\_db\_name> FAILOVER*

## How to export a schema:

To dump a schema to a SQL file, you need to connect to the machine as “<trig><env>adm” user and do:

*. /produits/pgsql/env/env\_<postgresql\_db\_name>*

*pg\_dump -U “schema\_name” -h $DBHOST -d “db\_name” -W > schema\_$(date +%d%m%y%H%M).sql*

## How to check a SQL file:

Before executing a SQL file, you need to check it with the following command:

*/produits/admindb/postgres/public/pack\_dba.ksh EXPLOIT\_VERIF\_TYPE\_SQL /Directory/of/sql/files/*

# How to exploit the Repmgrd daemon :

## How to start/stop a Repmgrd daemon :

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name> START*

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name> STOP*

## How to check a Repmgr cluster :

From any node :

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name> STATUS*

## How to start/stop/status a Repmgrd witness daemon :

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name\_witness> STARTWITNESS*

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name\_witness> STOPWITNESS*

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name\_witness> STATUSWITNESS*

## How to do a failover/switchover on a Repmgr cluster :

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name> FAILOVER*

*/produits/admindb/postgres/public/pack\_dba.ksh REPMGR <postgres\_node\_name> SWITCHOVER*

# How to exploit a Pgbouncer node :

## How to start/stop/status a Pgbouncer node :

From any node :

*/produits/admindb/postgres/public/pack\_dba.ksh PGBOUNCER APP START*

*/produits/admindb/postgres/public/pack\_dba.ksh PGBOUNCER APP STOP*

*/produits/admindb/postgres/public/pack\_dba.ksh PGBOUNCER APP STATUS*