PostgreSQL HA using repmgr and pgbouncer

Slavcho Trnkovski

Slavcho.Trnkovski@netcetera.com

2018-06-29, PGDay.IT 2018

netcetera

About me

▶ DBA@Netcetera

▶ Oracle, PostgreSQL, MS SQL

▶ PCI DSS

－ Patch management

PostgreSQL at Netcetera

▶ 2.5y ago
  − 9.4, 9.5, 9.6

▶ 30 Postgres servers

▶ SLA

▶ Regular update to latest minor version

netcetera

High Availability at Netcetera

▶ Oracle (Data Guard) and MS SQL (availability groups)

▶ 2 database servers (primary and standby)

▶ Application should always connect to primary only

▶ Easy switchover between primary and standby

▶ Primary dies -> Automatic failover

▶ Independent of postgres version

▶ Easy setup

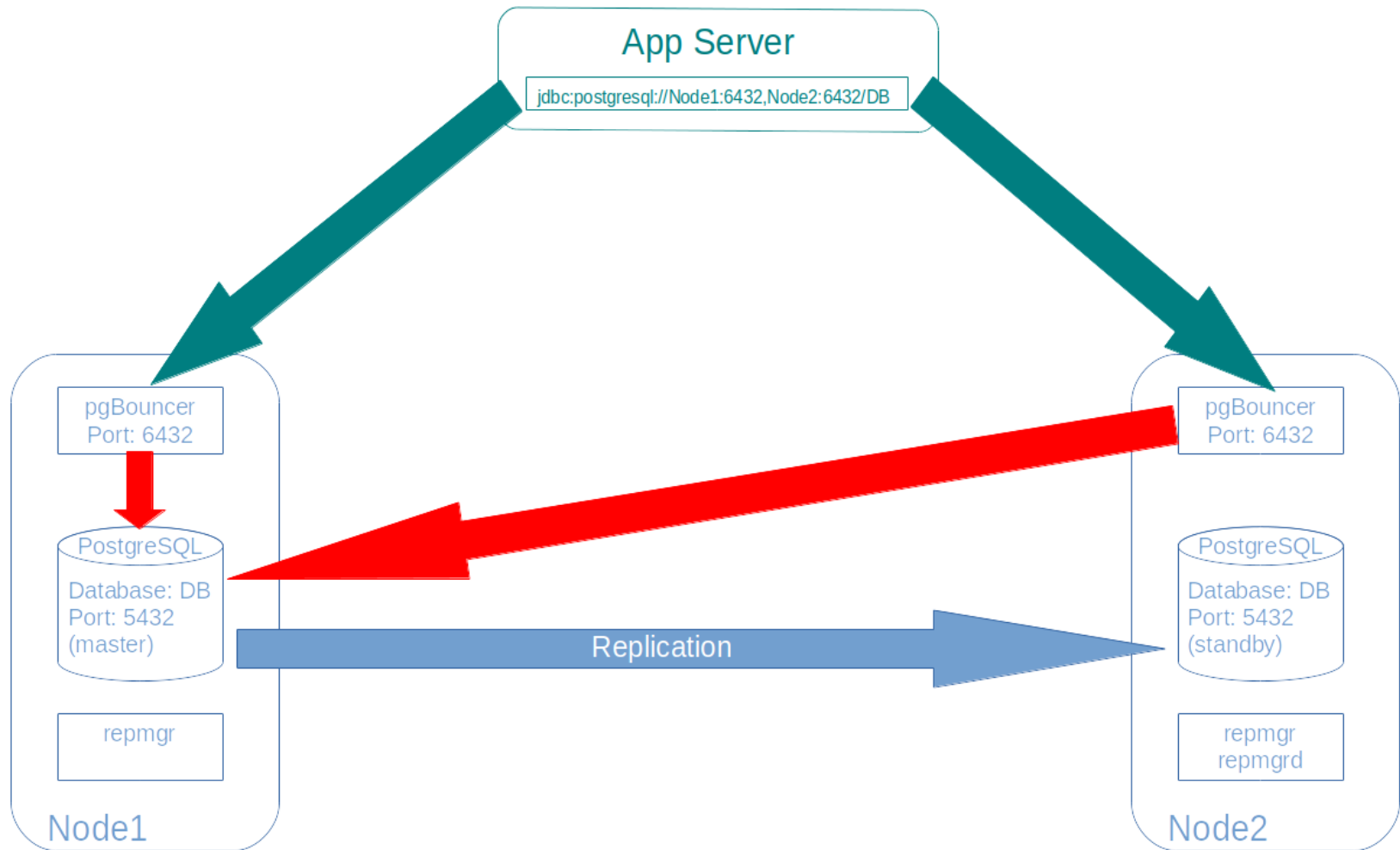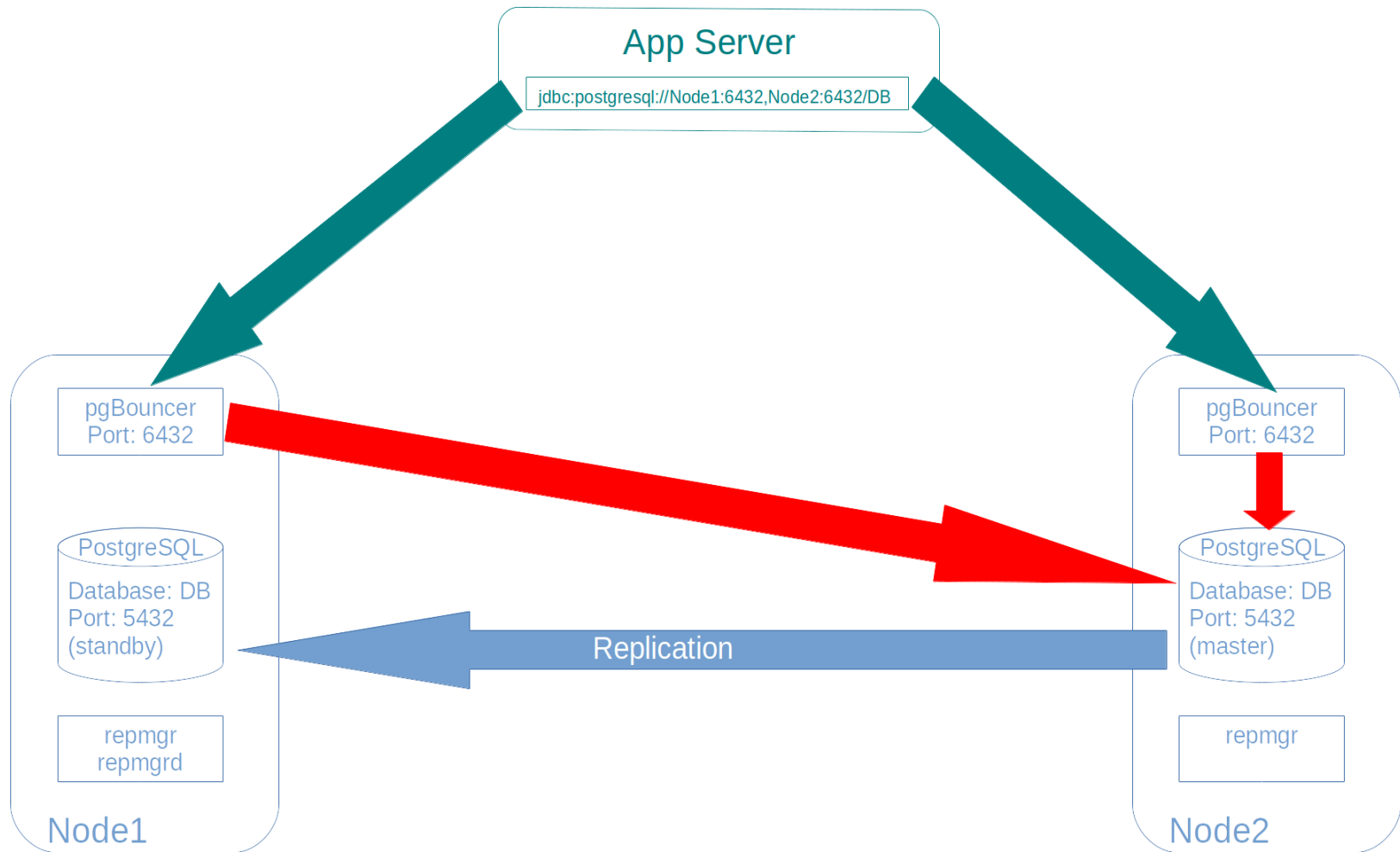netcetera

repmgr

- Open source tool (https://repmgr.org/)
  - 2nd Quadrant
- Replication and failover
- repmgr - command line tool
- repmgrd
- Version 3.* -> 4.*

PgBouncer

▶ https://pgbouncer.github.io

▶ Connection pooler

▶ **Isolating master after automatic failover**

▶ Separate database (pgbouncer) for maintenance

▶ Port: 6432

▶ jdbc:postgresql://*node1*:**6432**,*node2*:**6432**/*database*

App Server

jdbc:postgresql://Node1:6432,Node2:6432/DB

Node1

pgBouncer
Port: 6432

PostgreSQL
Database: DB
Port: 5432
(master)

repmgr

Node2

pgBouncer
Port: 6432

PostgreSQL
Database: DB
Port: 5432
(standby)

repmgr
repmgrd

Replication

App Server

jdbc:postgresql://Node1:6432,Node2:6432/DB

Node1

pgBouncer
Port: 6432

PostgreSQL
Database: DB
Port: 5432
(standby)

repmgr
repmgrd

Node2

pgBouncer
Port: 6432

PostgreSQL
Database: DB
Port: 5432
(master)

repmgr

Replication

Puppet support

▶ VMs – setup and maintenance

▶ Postgres – setup and maintenance

▶ HA
- LOG_INFO table
- pgbouncer.database.ini
- create_db_ini_file.sh
- switchover.sh
- promote.sh
- startup_pg_ha.sh

netcetera

pgbouncer_settings	ⓘ	`["##"]`	↗ ✎
pg_cluster_databases	ⓘ	pgday	↗ ✕ ☐
pg_cluster_hosts	ⓘ	skp-pg1-poc.netcetera.mk skp-pg2-poc.netcetera.mk	↗ ✕ ☐
pg_cluster_name	ⓘ	pgday_cluster	↗ ✕ ☐
pg_complete_version	ⓘ	9.5.13-1PGDG.rhel7	↗ ✕ ☐
pg_db_settings	ⓘ	`["wal_log_hints=on","max_wal_senders = 3","wal_level = 'hot_standby'","hot_standby = on","archive_mode = on","archive_command='cp %p /pgdata/archives/%f'","shared_preload_libraries = 'repmgr'"]`	↗ ✕ ☐
pg_ha_setup	ⓘ	true	↗ ✕ ☐
pg_hba_entries	ⓘ	`["local replication repmgr trust","host replication repmgr 10.1.24.62/16 trust","host replication repmgr 10.1.24.63/16 trust","local repmgr repmgr trust","host repmgr repmgr 10.1.24.62/16 trust","host repmgr repmgr 10.1.24.63/16 trust","host all all 10.1.0.0/16 md5"]`	↗ ✕ ☐
pg_major_version	ⓘ	9.5	↗ ✕ ☐
pg_node_id	ⓘ	1	↗ ✕ ☐
postgis_package_name	ⓘ	postgis24	↗ ✕ ☐
postgis_version	ⓘ	2.4.2-1.rhel7	↗ ✕ ☐
repmgr_settings	ⓘ	`["##"]`	↗ ✎
repmgr_version	ⓘ	4.0.5-1.rhel7	↗ ✕ ☐

pgdg

pgdg_db — definitions
```
orca:
  template: template1
  enable_postgis: 'true'
pgday:
  template: template1
  enable_postgis: 'true'
```

pgdg_schema — definitions
```
orca:
  database: orca
  dbuser: orca
pgday:
  database: pgday
  dbuser: pgday
```

pgdg_user — definitions
```
orca:
  osuser: nca
pgday:
  osuser: nca
```

create_db_ini_file.sh

```bash
set -x

# Configurable items
hosts="skp-pg1-poc.netcetera.mk skp-pg2-poc.netcetera.mk"


REPMGR_DB="repmgr"
REPMGR_USER="repmgr"

PGBOUNCER_DATABASE_INI="/etc/pgbouncer/pgbouncer.database.ini"
PGBOUNCER_DATABASE_INI_NEW="/tmp/pgbouncer.database.ini"
PGBOUNCER_DATABASE="pgday"
PGBOUNCER_PORT=6432

for HOST in $hosts
do
    # Recreate the pgbouncer config file
    echo -e "[databases]\n" > $PGBOUNCER_DATABASE_INI_NEW

    for pg_base in $PGBOUNCER_DATABASE
    do
        psql -d $REPMGR_DB -U $REPMGR_USER -t -A -c \
                    "SELECT '${pg_base}= ' || 'application_name=pgbouncer_${HOST} ' || conninfo FROM nodes WHERE active = TRUE AND type='primary'" >> $PGBOUNCER_DATABASE_INI_NEW
            done

    # Remove the user= and dbname= strings from the new file because we don't need those.
    sed -i 's/\<user.*\>//g' $PGBOUNCER_DATABASE_INI_NEW

    rsync $PGBOUNCER_DATABASE_INI_NEW $HOST:$PGBOUNCER_DATABASE_INI

    psql -tc "reload" -h $HOST -p $PGBOUNCER_PORT -U manager pgbouncer

done

# Clean up generated file
rm $PGBOUNCER_DATABASE_INI_NEW
```

pgbouncer.ini / pgbouncer.database.ini

```
][pgbouncer]
logfile = /var/log/pgbouncer/pgbouncer.log
pidfile = /var/run/pgbouncer/pgbouncer.pid
listen_addr = *
listen_port = 6432
auth_type = md5
auth_file = /etc/pgbouncer/userlist.txt
ignore_startup_parameters = extra_float_digits
admin_users = manager
stats_users = manager
pool_mode = transaction
max_client_conn = 100
default_pool_size = 20
min_pool_size = 5
reserve_pool_size = 5
reserve_pool_timeout = 3
log_connections = 1
log_disconnections = 1
log_pooler_errors = 1

##

%include /etc/pgbouncer/pgbouncer.database.ini
```

```
[databases]

pgday= application_name=pgbouncer_skp-pg1-poc.netcetera.mk host=10.1.24.62
```

repmgr.conf / repmgr.conf.erb

```
node_id=1
log_file='/var/log/repmgr/repmgr.log'
data_directory=/pgdata/9.5/
node_name=skp-pg1-poc
conninfo='host=10.1.24.62 user=repmgr dbname=repmgr'
pg_bindir='/usr/pgsql-9.5/bin'
failover=automatic
promote_command=/var/lib/pgsql/promote.sh
follow_command='repmgr standby follow -f /etc/repmgr/9.5/repmgr.conf --log-to-file'
service_start_command=sudo systemctl start postgresql-9.5.service
service_stop_command=sudo systemctl stop postgresql-9.5.service
service_reload_command=sudo systemctl reload postgresql-9.5.service
service_restart_command=sudo systemctl restart postgresql-9.5.service

##
```

```
<% if @repmgr_version.start_with?("4") -%>
node_id=<%=@pg_node_id+"\n"-%>
log_file='/var/log/repmgr/repmgr.log'
data_directory=/pgdata/<%=@pg_major_version-%>/
<% else %>
cluster=<%=@pg_cluster_name+"\n"-%>
node=<%=@pg_node_id+"\n"-%>
logfile='/var/log/repmgr/repmgr.log'
<% end -%>
node_name=<%=@hostname+"\n"-%>
conninfo='host=<%=@ipaddress-%> user=repmgr dbname=repmgr'
pg_bindir='/usr/pgsql-<%=@pg_major_version-%>/bin'
failover=automatic
promote_command=/var/lib/pgsql/promote.sh
follow_command='repmgr standby follow -f /etc/repmgr/<%=@pg_major_version-%>/repmgr.conf --log-to-file'
service_start_command=sudo systemctl start postgresql-<%=@pg_major_version-%>.service
service_stop_command=sudo systemctl stop postgresql-<%=@pg_major_version-%>.service
service_reload_command=sudo systemctl reload postgresql-<%=@pg_major_version-%>.service
service_restart_command=sudo systemctl restart postgresql-<%=@pg_major_version-%>.service

<% @repmgr_settings.each do |setting| -%>
<%=setting %>
<% end -%>
```

switchover.sh

```bash
hosts="skp-pg1-poc.netcetera.mk skp-pg2-poc.netcetera.mk"
host_dir="/etc/repmgr/9.5/repmgr.conf"
remote_dir="/etc/repmgr/9.5/repmgr.conf"
log_dir="/var/log/repmgr/switchover.log"
log_dir_old="/var/log/repmgr/switchover.old"
switchover_done=false
REPMGR_DB="repmgr"
REPMGR_USER="repmgr"
REPMGR_SCHEMA="repmgr_pgday_cluster"
PGBOUNCER_DATABASE_INI="/etc/pgbouncer/pgbouncer.database.ini"
PGBOUNCER_DATABASE_INI_NEW="/tmp/pgbouncer.database.ini"
PGBOUNCER_DATABASE="pgday"
PGBOUNCER_PORT=6432
# 1. Clear the log file from before, then execute switchover command
# Backup old log file
if [ -e $log_dir ] ; then
    cat $log_dir > $log_dir_old
fi
truncate -s 0 $log_dir
repmgr -f $host_dir -C $remote_dir --log-level DEBUG --verbose standby switchover &> $log_dir
# 2. Check if switchover completed succesfully
if grep -Fxq "NOTICE: switchover was successful" $log_dir
then
        switchover_done=true
else
        echo "Switchover was not completed succesfully check your config file.";
        exit 1;
fi
# 3. If the switchover was completed succesfully we need to start changing pgbouncers config file
if [ "$switchover_done" = "true" ] ; then
    for HOST in $hosts
    do
        # Recreate the pgbouncer config file
        echo -e "[databases]\n" > $PGBOUNCER_DATABASE_INI_NEW

        for pg_base in $PGBOUNCER_DATABASE
        do
        psql -d $REPMGR_DB -U $REPMGR_USER -t -A -c \
                "SELECT '${pg_base}= ' || 'application_name=pgbouncer_${HOST} ' || conninfo FROM nodes WHERE active = TRUE AND type='primary'" >> $PGBOUNCER_DATABASE_INI_NEW
        done
        # Remove the user= and dbname= strings from the new file because we don't need those.
        sed -i 's/\<user.*\>//g' $PGBOUNCER_DATABASE_INI_NEW
        rsync $PGBOUNCER_DATABASE_INI_NEW $HOST:$PGBOUNCER_DATABASE_INI
        psql -tc "reload" -h $HOST -p $PGBOUNCER_PORT -U manager pgbouncer
    done
    # Clean up generated file
    rm $PGBOUNCER_DATABASE_INI_NEW
fi
# 4. Add what happend to the log table
psql -d repmgr -c "INSERT INTO log_info(time, info, current_master) VALUES ((SELECT now()), 'Switchover initialized by ${HOSTNAME} and was completed succesfully.', '${HOSTNAME}')"
echo "Switchover and reconfiguration of pgbouncer completed."
```

promote.sh

```bash
PGBOUNCER_HOSTS="skp-pg1-poc.netcetera.mk skp-pg2-poc.netcetera.mk"
PGBOUNCER_DATABASE_INI="/etc/pgbouncer/pgbouncer.database.ini"
PGBOUNCER_DATABASE_INI_NEW="/tmp/pgbouncer.database.ini"
PGBOUNCER_DATABASE="pgday"
PGBOUNCER_PORT=6432
REPMGR_DB="repmgr"
REPMGR_USER="repmgr"
REPMGR_SCHEMA="repmgr_pgday_cluster"

# 1. Promote this node from standby to master
/usr/pgsql-9.5/bin/repmgr standby promote -f /etc/repmgr/9.5/repmgr.conf

# 2. Reconfigure pgbouncer instances
for HOST in $PGBOUNCER_HOSTS
do
    # Recreate the pgbouncer config file
    echo -e "[databases]\n" > $PGBOUNCER_DATABASE_INI_NEW

    for pg_base in $PGBOUNCER_DATABASE
    do
        psql -d $REPMGR_DB -U $REPMGR_USER -t -A -c \
            "SELECT '${pg_base}= ' || 'application_name=pgbouncer_${HOST} ' || conninfo FROM nodes WHERE active = TRUE AND type='primary'" >> $PGBOUNCER_DATABASE_INI_NEW

    done    # Remove the user= and dbname= strings from the new file because we don't need those.
    sed -i 's/\<user.*\>//g' $PGBOUNCER_DATABASE_INI_NEW
    rsync $PGBOUNCER_DATABASE_INI_NEW $HOST:$PGBOUNCER_DATABASE_INI
    psql -tc "reload" -h $HOST -p $PGBOUNCER_PORT -U manager pgbouncer
done

# Clean up generated file
rm $PGBOUNCER_DATABASE_INI_NEW

psql -d repmgr -c "INSERT INTO log_info(time, info, current_master) VALUES ((SELECT now()), 'Automatic failover initialized by ${HOSTNAME} and was completed succesfully.', '${HOSTNAME}')"
echo "Automatic failover and reconfiguration of pgbouncer completed."
```

startup_pg_ha.sh

```bash
REPMGR_DB="repmgr"
REPMGR_USER="repmgr"

HOST_1=skp-pg1-poc.netcetera.mk
HOST_2=skp-pg2-poc.netcetera.mk
host=$(hostname -I)

PGBOUNCER_DATABASE_INI="/etc/pgbouncer/pgbouncer.database.ini"
PGBOUNCER_DATABASE="pgday"

# 1. Get the id from the log table on both servers
id_server1=$(psql -h $HOST_1 -d repmgr -U repmgr -t -A -c "SELECT max(id) FROM log_info")
id_server2=$(psql -h $HOST_2 -d repmgr -U repmgr -t -A -c "SELECT max(id) FROM log_info")

# 2. Compare the two ids
if [ "$id_server1" -eq "$id_server2" ] ; then
    echo "id is equal you should check if the right master is in the config file"
elif [ "$id_server1" -gt "$id_server2" ] ; then
    echo -e "[databases]\n" > $PGBOUNCER_DATABASE_INI

    for pg_base in $PGBOUNCER_DATABASE
    do
    psql -h $HOST_1 -d $REPMGR_DB -U $REPMGR_USER -t -A -c \
                    "SELECT '${pg_base}= ' || 'application_name=pgbouncer_${HOST_1} ' || conninfo FROM nodes WHERE active = TRUE AND type='primary'" >> $PGBOUNCER_DATABASE_INI

    done
    sed -i 's/\<user.*\>//g' $PGBOUNCER_DATABASE_INI
elif [ "$id_server1" -lt "$id_server2" ] ; then
    echo -e "[databases]\n" > $PGBOUNCER_DATABASE_INI

    for pg_base in $PGBOUNCER_DATABASE
    do
        psql -h $HOST_2 -d $REPMGR_DB -U $REPMGR_USER -t -A -c
                    "SELECT '${pg_base}= ' || 'application_name=pgbouncer_${HOST_2} ' || conninfo FROM nodes WHERE active = TRUE AND type='primary'" >> $PGBOUNCER_DATABASE_INI

    done
    sed -i 's/\<user.*\>//g' $PGBOUNCER_DATABASE_INI
else
    echo "Error check the id in the tables"
fi
```

netcetera

pgbouncer.conf / pgbouncerModifier.service

```
[Unit]
 After=pgbouncerModifier.service
```

```
[Unit]
Description=Check who is current master and modify pgbouncer file if needed
After=postgresql-9.5.service

[Service]
ExecStart=/usr/local/bin/startup_pg_ha.sh
Type=oneshot
RemainAfterExit=yes
PIDFile=/var/run/pgbouncer/startup_pg_ha.pid
User=postgres

[Install]
WantedBy=default.target
```

netcetera

Use cases (switchover)

▶ Postgres minor version update
   — stop repmgrd on standby
   — update standby
   — switchover.sh
   — update postgres on new standby
   — switchover.sh
   — start repmgrd on standby

▶ Host patching (reboot)
   — stop repmgrd on standby
   — stop pgbouncer, patch and reboot standby
   — switchover.sh
   — stop pgbouncer, patch and reboot new standby
   — switchover.sh
   — start repmgrd on standby

Witness?

▶ VMware VMs

▶ Additional maintenance during switchover

▶ Isolate master -> repmgr

netcetera

Q & A

KEEP CALM AND USE POSTGRES