# Руководство по установке pgBadger под CentOS 8

# Предварительная настройка PostgreSQL

Перед установкой pgBadger необходимо предварительно настроить PostgreSQL, открываем файл конфигурации (/var/lib/pgsql/13/data/postgresql.conf) и устанавливаем следующие значения:

log\_min\_duration\_statement = 10

log\_checkpoints = on

log\_connections = on

log\_disconnections = on

log\_error\_verbosity = default

log\_line\_prefix = '%t [%p]: db=%d,user%u,app=%a,client=%h'

log\_lock\_waits = on

log\_statement = 'none'

log\_temp\_files = 0

log\_autovacuum\_min\_duration = 0

# Установка pgBadger

Пропускаем если уже установлено:

# sudo yum install -y perl perl-devel

# sudo yum install wget

# sudo yum install git

Устанавливаем pgBadger из GitHub либо из архива в поставке:

# sudo wget <https://github.com/darold/pgbadger/archive/v11.6.tar.gz>

# sudo tar xzvf v11.6.tar.gz

# cd pgbadger-11.6

# sudo perl Makefile.PL

# sudo make && sudo make install

Проверяем, что сборка завершена корректно:

# pgbadger -V

# systemctl restart postgresql-13

Проверяем применились ли настройки:

# su postgres

# psql

#postgres=# show log\_line\_prefix;

Должно отобразиться измененное значение! Пока данные настройки не применятся – не получится собрать необходимый отчет.

# Создание отчета pgBadger

# cd pgbadger-11.6

# pgbadger /var/lib/pgsql/13/data/log/postgresql-Fri.log

Если необходимо объединить в отчете все файлы лога:

# pgbadger /var/lib/pgsql/13/data/log/postgresql-\* l-o reportName.html

В папке будет создан файл out.html – это и есть нужный отчет!