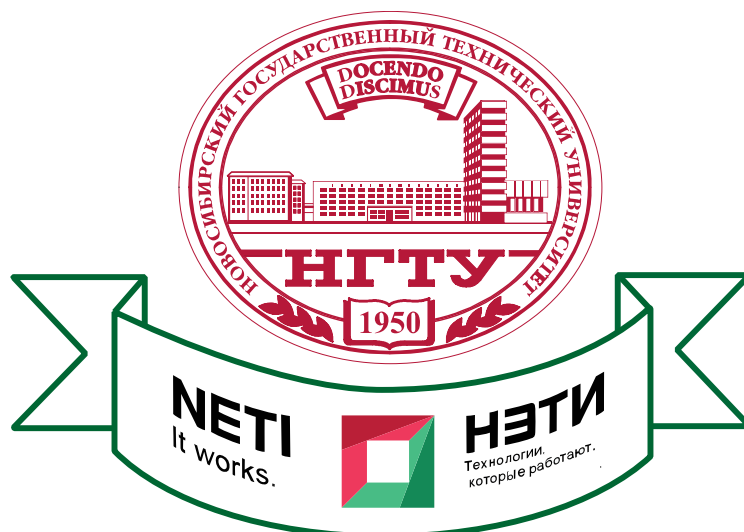


Министерство науки и высшего образования
Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

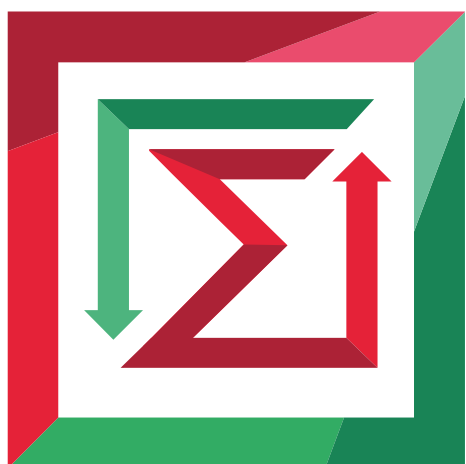


Кафедра теоретической и прикладной информатики

Лабораторная работа № 8

по дисциплине «Администрирование информационных систем»

Мониторинг PostgresPro



| | |
|----------------|---------------------------------|
| Факультет: | ПМИ |
| Группа: | ПМИ-02 |
| Бригада: | 8 |
| Студенты: | Сидоров Даниил, Дюков Богдан |
| Преподаватель: | Аврунев О.Е. |

Новосибирск

2026

Ход работы

1. Включили pg_stat_statements на нашем сервере, изменив строку в postgresql.conf:

```
# - Shared Library Preloading -  
  
#local_preload_libraries = ''  
#session_preload_libraries = ''  
shared_preload_libraries = 'pg_stat_statements' # (change requires restart)  
#jit_provider = 'llvmjit' # JIT library to use
```

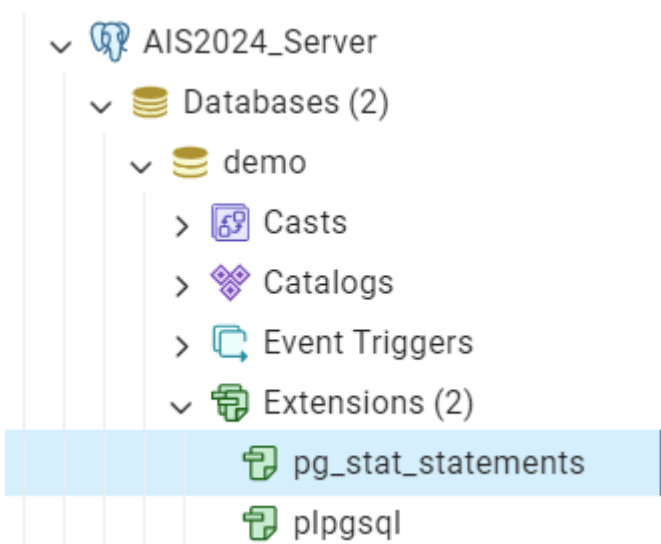
Перезапустили службу:

```
[root@centos-stream-8 ~]# systemctl restart postgrespro-std-15.service  
[root@centos-stream-8 ~]#
```

Выполнили команду для создания представления, необходимого для доступа к данным:

```
demo=# create extension pg_stat_statements;  
CREATE EXTENSION  
demo=#
```

Проверили наличие расширения:



2. Создали скрипты с sql-запросами:

| Вариант | Скрипты |
|---------|--|
| 8 | Удаление строки из таблицы, интервал между итерациями 0,5 сек; Вставка строк в таблицу, интервал между итерациями 0,5 сек; 500 итераций в цикле. |

Файл insert.py:

```
import psycopg2
from time import sleep

conn = psycopg2.connect("dbname='demo' user='dba' host='127.0.0.1' port='5432' password='disidorov2020'")
cur = conn.cursor()

i = 1
while i <= 500:
    cur.execute("INSERT INTO myschema.table1 (id) VALUES (%s)", (i,))
    conn.commit()
    sleep(0.5)
    i += 1

conn.close()
```

Файл delete.py:

```
import psycopg2
from time import sleep

conn = psycopg2.connect("dbname='demo' user='dba' host='127.0.0.1' port='5432' password='disidorov2020'")
cur = conn.cursor()

i = 1
while i <= 500:
    cur.execute("DELETE FROM myschema.table1 WHERE id = %s", (i,))
    conn.commit()
    sleep(0.5)
    i += 1

conn.close()
```

3. Для соответствующих выполняемым скриптам сеансов привели информацию из представлений: pg_stat_activity и pg_stat_statements.

Запустили скрипты:

```
[dba@centos-stream-8 ~]$ python3 delete.py &
[1] 110982

[dba@centos-stream-8 ~]$ python3 insert.py &
[1] 111356

[dba@centos-stream-8 ~]$
```

Соответствующие PID в PgAdmin:

| | | | | | | | | | | | |
|--|--|--|--------|------|-----|--|-----------|------------------------|--|------|--------------------|
| | | | 110984 | demo | dba | | 127.0.0.1 | 2024-03-26 22:09:35... | | idle | Client: ClientRead |
| | | | 111359 | demo | dba | | 127.0.0.1 | 2024-03-26 22:15:22... | | idle | Client: ClientRead |

Информация из pg_stat_activity:

```
demo=# select datid, datname, pid, usesysid, username, application_name, client_addr, client_hostname, client_port from pg_stat_activity where pid=110984;
 datid | datname | pid  | usesysid | username | application_name | client_addr | client_hostname | client_port
-----+-----+-----+-----+-----+-----+-----+-----+-----
 16389 | demo   | 110984 |      16388 | dba      |                  | 127.0.0.1   |                  |         43706
(1 row)

demo=# select backend_start, xact_start, query_start, state_change, state from pg_stat_activity where pid=110984;
 backend_start | xact_start | query_start | state_change | state
-----+-----+-----+-----+-----
 2024-03-26 22:09:35.406982+07 |           | 2024-03-26 22:10:45.911275+07 | 2024-03-26 22:10:45.912236+07 | idle
(1 row)

demo=# select backend_xid, backend_xmin, query from pg_stat_activity where pid=110984;
 backend_xid | backend_xmin | query
-----+-----+-----
           |           | COMMIT
(1 row)

demo=# select datid, datname, pid, usesysid, username, application_name, client_addr, client_hostname, client_port from pg_stat_activity where pid=111359;
 datid | datname | pid  | usesysid | username | application_name | client_addr | client_hostname | client_port
-----+-----+-----+-----+-----+-----+-----+-----+-----
 16389 | demo   | 111359 |      16388 | dba      |                  | 127.0.0.1   |                  |         44388
(1 row)

demo=# select backend_start, xact_start, query_start, state_change, state from pg_stat_activity where pid=111359;
 backend_start | xact_start | query_start | state_change | state
-----+-----+-----+-----+-----
 2024-03-26 22:15:22.482795+07 |           | 2024-03-26 22:16:49.075643+07 | 2024-03-26 22:16:49.076926+07 | idle
(1 row)

demo=# select backend_xid, backend_xmin, query from pg_stat_activity where pid=111359;
 backend_xid | backend_xmin | query
-----+-----+-----
           |           | COMMIT
(1 row)

demo=#
```

Запустили скрипты ещё раз:

```
[dba@centos-stream-8 ~]$ python3 delete.py &
[1] 111756
[dba@centos-stream-8 ~]$ python3 insert.py &
[2] 111761
[dba@centos-stream-8 ~]$
```

| | | | | | | | | | | | |
|--|--|--|--------|------|-----|--|-----------|------------------------|--|------|--------------------|
| | | | 111757 | demo | dba | | 127.0.0.1 | 2024-03-26 22:27:04... | | idle | Client: ClientRead |
| | | | 111762 | demo | dba | | 127.0.0.1 | 2024-03-26 22:27:11... | | idle | Client: ClientRead |

Информация из pg_stat_statements:

```

demo=# select userid, dbid, queryid, calls, total_plan_time, min_plan_time from pg_stat_statements;
userid | dbid |      queryid      | calls | total_plan_time | min_plan_time
-----+-----+-----+-----+-----+-----
16388 | 16389 | 7547780484281152998 | 20 | 0 | 0
16388 | 16389 | -7583660836945374690 | 2 | 0 | 0
16388 | 5 | 7923662078611323493 | 3 | 0 | 0
16388 | 16389 | -4704991346601685052 | 997 | 0 | 0
16388 | 16389 | -3755512860370358302 | 1 | 0 | 0
16388 | 16389 | -450830737520461923 | 16 | 0 | 0
16388 | 16389 | 50231264506710404 | 20 | 0 | 0
16388 | 16389 | -2913445568734097908 | 10 | 0 | 0
16388 | 16389 | 2397681704071010949 | 2497 | 0 | 0
16388 | 16389 | -7353591178260607276 | 2 | 0 | 0
16388 | 16389 | -2356550944868553968 | 4 | 0 | 0
16388 | 16389 | 2261200698118043237 | 1 | 0 | 0
16388 | 16389 | 5235866162762847415 | 14 | 0 | 0
16388 | 16389 | -2987976877938653778 | 2 | 0 | 0
16388 | 5 | 7515305722061141122 | 12 | 0 | 0
16388 | 16389 | 1429998583312352561 | 20 | 0 | 0
16388 | 16389 | 5805784599309057006 | 1 | 0 | 0
16388 | 5 | 50231264506710404 | 2 | 0 | 0
16388 | 16389 | 3694949039461716331 | 2479 | 0 | 0
16388 | 16389 | -9102621307683535516 | 20 | 0 | 0
16388 | 16389 | -387259410797068724 | 26 | 0 | 0
16388 | 5 | 7912147036522798865 | 42 | 0 | 0
16388 | 16389 | 4645464847702058055 | 16 | 0 | 0
16388 | 5 | 905659920195589556 | 6 | 0 | 0
16388 | 16389 | -6008779406163511298 | 2 | 0 | 0
16388 | 16389 | 267935211159193185 | 2 | 0 | 0
16388 | 5 | 6384784041518614479 | 2 | 0 | 0
16388 | 5 | 1429998583312352561 | 2 | 0 | 0
16388 | 5 | 7553164001195393538 | 2 | 0 | 0
16388 | 16389 | 2680898860122635153 | 21 | 0 | 0
16388 | 16389 | 7565987115177184934 | 148 | 0 | 0
16388 | 16389 | 905976906289400631 | 20 | 0 | 0
16388 | 16389 | 7878559025315074592 | 59 | 0 | 0
16388 | 16389 | -3995683845443549838 | 2 | 0 | 0
16388 | 5 | 3322742666069815207 | 3 | 0 | 0
16388 | 5 | 2395810797885923577 | 21 | 0 | 0
16388 | 5 | 7547780484281152998 | 2 | 0 | 0
16388 | 16389 | 7553164001195393538 | 20 | 0 | 0
16388 | 5 | -4611011376356164422 | 17 | 0 | 0
16388 | 16389 | -6623627158212871993 | 32 | 0 | 0
16388 | 16389 | 5814878174718500201 | 19 | 0 | 0
16388 | 5 | -9102621307683535516 | 2 | 0 | 0
16388 | 16389 | 2147166802619366732 | 16 | 0 | 0
16388 | 16389 | 3737582156084185939 | 20 | 0 | 0
16388 | 5 | 905976906289400631 | 2 | 0 | 0
16388 | 16389 | -2312193446587064208 | 1 | 0 | 0
16388 | 5 | 3737582156084185939 | 2 | 0 | 0
16388 | 5 | 653864183907931385 | 13 | 0 | 0
16388 | 5 | 4644049984140703386 | 13 | 0 | 0

```


4. Привели значения конфигурационных параметров, отвечающих за запись журнала сообщений сервера:

```
#-----
# REPORTING AND LOGGING
#-----

# - Where to Log -

#log_destination = 'stderr'          # Valid values are combinations of
                                     # stderr, csvlog, jsonlog, syslog, and
                                     # eventlog, depending on platform.
                                     # csvlog and jsonlog require
                                     # logging_collector to be on.

# This is used when logging to stderr:
logging_collector = on # Enable capturing of stderr, jsonlog,
                       # and csvlog into log files. Required
                       # to be on for csvlogs and jsonlogs.
                       # (change requires restart)

# These are only used if logging_collector is on:
#log_directory = 'log'               # directory where log files are written,
                                     # can be absolute or relative to PGDATA
#log_filename = 'postgresql-%Y-%m-%d_%H%M%S.log' # log file name pattern,
                                     # can include strftime() escapes
#log_file_mode = 0600                # creation mode for log files,
                                     # begin with 0 to use octal notation
#log_rotation_age = 1d                # Automatic rotation of logfiles will
                                     # happen after that time. 0 disables.
#log_rotation_size = 10MB            # Automatic rotation of logfiles will
                                     # happen after that much log output.
                                     # 0 disables.
#log_truncate_on_rotation = off       # If on, an existing log file with the
                                     # same name as the new log file will be
                                     # truncated rather than appended to.
                                     # But such truncation only occurs on
                                     # time-driven rotation, not on restarts
                                     # or size-driven rotation. Default is
                                     # off, meaning append to existing files
                                     # in all cases.

# These are relevant when logging to syslog:
#syslog_facility = 'LOCAL0'
#syslog_ident = 'postgres'
#syslog_sequence_numbers = on
#syslog_split_messages = on

# This is only relevant when logging to eventlog (Windows):
# (change requires restart)
#event_source = 'PostgreSQL'
```

- When to Log -

```
#log_min_messages = warning      # values in order of decreasing detail:
                                  #   debug5
                                  #   debug4
                                  #   debug3
                                  #   debug2
                                  #   debug1
                                  #   info
                                  #   notice
                                  #   warning
                                  #   error
                                  #   log
                                  #   fatal
                                  #   panic

#log_min_error_statement = error  # values in order of decreasing detail:
                                  #   debug5
                                  #   debug4
                                  #   debug3
                                  #   debug2
                                  #   debug1
                                  #   info
                                  #   notice
                                  #   warning
                                  #   error
                                  #   log
                                  #   fatal
                                  #   panic (effectively off)

#log_min_duration_statement = -1 # -1 is disabled, 0 logs all statements
                                  # and their durations, > 0 logs only
                                  # statements running at least this number
                                  # of milliseconds

#log_min_duration_sample = -1    # -1 is disabled, 0 logs a sample of statements
                                  # and their durations, > 0 logs only a sample of
                                  # statements running at least this number
                                  # of milliseconds;
                                  # sample fraction is determined by log_statement_sample_rate

#log_statement_sample_rate = 1.0 # fraction of logged statements exceeding
                                  # log_min_duration_sample to be logged;
                                  # 1.0 logs all such statements, 0.0 never logs

#log_transaction_sample_rate = 0.0 # fraction of transactions whose statements
                                  # are logged regardless of their duration; 1.0 logs all
                                  # statements from all transactions, 0.0 never logs
```

```

#log_transaction_sample_rate = 0.0      # fraction of transactions whose statements
                                         # are logged regardless of their duration; 1.0 logs all
                                         # statements from all transactions, 0.0 never logs

#log_startup_progress_interval = 10s    # Time between progress updates for
                                         # long-running startup operations.
                                         # 0 disables the feature, > 0 indicates
                                         # the interval in milliseconds.

# - What to Log -

#debug_print_parse = off
#debug_print_rewritten = off
#debug_print_plan = off
#debug_pretty_print = on
#log_autovacuum_min_duration = 10min    # log autovacuum activity;
                                         # -1 disables, 0 logs all actions and
                                         # their durations, > 0 logs only
                                         # actions running at least this number
                                         # of milliseconds.

#log_checkpoints = on
#log_connections = off
#log_disconnections = off
#log_duration = off
#log_error_verbosity = default          # terse, default, or verbose messages
#log_hostname = off
#log_line_prefix = '%m [%p] '          # special values:
                                         # %a = application name
                                         # %u = user name
                                         # %d = database name
                                         # %r = remote host and port
                                         # %h = remote host
                                         # %b = backend type
                                         # %p = process ID
                                         # %P = process ID of parallel group leader
                                         # %t = timestamp without milliseconds
                                         # %m = timestamp with milliseconds
                                         # %n = timestamp with milliseconds (as a Unix epoch)
                                         # %Q = query ID (0 if none or not computed)
                                         # %i = command tag
                                         # %e = SQL state
                                         # %c = session ID
                                         # %l = session line number
                                         # %s = session start timestamp
                                         # %v = virtual transaction ID
                                         # %x = transaction ID (0 if none)
                                         # %q = stop here in non-session
                                         #       processes
                                         # %% = '%'
                                         # e.g. '%u%%%d>'

```

```

#log_lock_waits = off                  # e.g. '<%u%%%d>'
#log_recovery_conflict_waits = off    # log lock waits >= deadlock_timeout
#log_parameter_max_length = -1        # log standby recovery conflict waits
                                         # >= deadlock_timeout
                                         # when logging statements, limit logged
                                         # bind-parameter values to N bytes;
                                         # -1 means print in full, 0 disables
#log_parameter_max_length_on_error = 0 # when logging an error, limit logged
                                         # bind-parameter values to N bytes;
                                         # -1 means print in full, 0 disables
#log_statement = 'none'               # none, ddl, mod, all
#log_replication_commands = off
#log_temp_files = -1                  # log temporary files equal or larger
                                         # than the specified size in kilobytes;
                                         # -1 disables, 0 logs all temp files

log_timezone = 'Asia/Novosibirsk'

```

5. Включили запись выполняемых операторов. Значение all для параметра log_statement означает, что все SQL-операторы будут записаны в журнал:

```
log_statement = 'all'           # -1 means print in full, 0 disables
                                # none, ddl, mod, all
#log_replication_commands = off
#log_temp_files = -1           # log temporary files equal or larger
                                # than the specified size in kilobytes;
                                # -1 disables, 0 logs all temp files

log_timezone = 'Asia/Novosibirsk'
```

Перезапустили службу и убедились, что параметр установлен:

```
[dba@centos-stream-8 ~]$ systemctl restart postgrespro-std-15
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'postgrespro-std-15.service'.
Authenticating as: dba
Password:
==== AUTHENTICATION COMPLETE ====
```

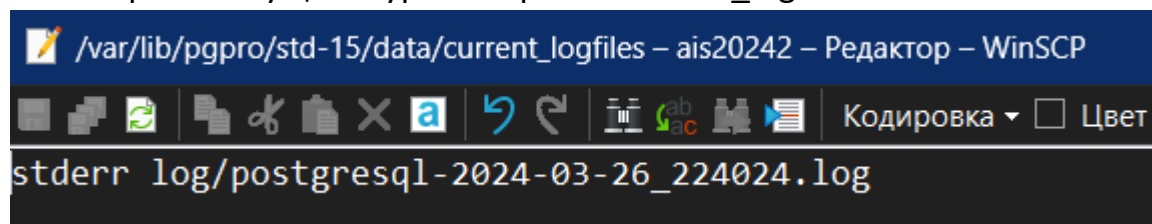
```
demo=# show log_statement;
 log_statement
-----
      all
(1 row)

demo=#
```

Повторно выполнили сценарии:

```
[dba@centos-stream-8 ~]$ python3 delete.py &
[1] 112326
[dba@centos-stream-8 ~]$ python3 insert.py &
[2] 112328
[dba@centos-stream-8 ~]$
```

Посмотрели текущий журнал в файле current_logfiles:



```
/var/lib/pgpro/std-15/data/current_logfiles - ais20242 - Редактор - WinSCP
stderr log/postgresql-2024-03-26_224024.log
```

Содержимое этого журнала:

```
2024-03-26 22:41:58.266 +07 [112327] LOG: statement: BEGIN
2024-03-26 22:41:58.267 +07 [112327] LOG: statement: DELETE FROM myschema.table1 WHERE id = 11
2024-03-26 22:41:58.267 +07 [112327] LOG: statement: COMMIT
2024-03-26 22:41:58.769 +07 [112327] LOG: statement: BEGIN
2024-03-26 22:41:58.770 +07 [112327] LOG: statement: DELETE FROM myschema.table1 WHERE id = 12
2024-03-26 22:41:58.770 +07 [112327] LOG: statement: COMMIT
2024-03-26 22:41:59.273 +07 [112327] LOG: statement: BEGIN
2024-03-26 22:41:59.273 +07 [112327] LOG: statement: DELETE FROM myschema.table1 WHERE id = 13
2024-03-26 22:41:59.274 +07 [112327] LOG: statement: COMMIT
2024-03-26 22:41:59.691 +07 [112329] LOG: statement: BEGIN
2024-03-26 22:41:59.692 +07 [112329] LOG: statement: INSERT INTO myschema.table1 (id) VALUES (1)
2024-03-26 22:41:59.694 +07 [112329] LOG: statement: COMMIT
2024-03-26 22:41:59.776 +07 [112327] LOG: statement: BEGIN
2024-03-26 22:41:59.776 +07 [112327] LOG: statement: DELETE FROM myschema.table1 WHERE id = 14
2024-03-26 22:41:59.777 +07 [112327] LOG: statement: COMMIT
2024-03-26 22:42:00.197 +07 [112329] LOG: statement: BEGIN
2024-03-26 22:42:00.198 +07 [112329] LOG: statement: INSERT INTO myschema.table1 (id) VALUES (2)
2024-03-26 22:42:00.198 +07 [112329] LOG: statement: COMMIT
2024-03-26 22:42:00.279 +07 [112327] LOG: statement: BEGIN
2024-03-26 22:42:00.279 +07 [112327] LOG: statement: DELETE FROM myschema.table1 WHERE id = 15
2024-03-26 22:42:00.280 +07 [112327] LOG: statement: COMMIT
2024-03-26 22:42:00.700 +07 [112329] LOG: statement: BEGIN
2024-03-26 22:42:00.701 +07 [112329] LOG: statement: INSERT INTO myschema.table1 (id) VALUES (3)
2024-03-26 22:42:00.701 +07 [112329] LOG: statement: COMMIT
2024-03-26 22:42:00.781 +07 [112327] LOG: statement: BEGIN
2024-03-26 22:42:00.782 +07 [112327] LOG: statement: DELETE FROM myschema.table1 WHERE id = 16
2024-03-26 22:42:00.782 +07 [112327] LOG: statement: COMMIT
2024-03-26 22:42:01.203 +07 [112329] LOG: statement: BEGIN
2024-03-26 22:42:01.204 +07 [112329] LOG: statement: INSERT INTO myschema.table1 (id) VALUES (4)
2024-03-26 22:42:01.204 +07 [112329] LOG: statement: COMMIT
```