

Инструменты диагностики PostgreSQL/PostgresPro

Михаил Жилин, Postgres Professional

Михаил Жилин

- Выпускник Физтеха
- 17 лет занимаюсь
производительностью программ
- Любимые языки программирования:
Java и Си
- Обожаю Open Source
- Особенно FreeBSD и PostgreSQL



План доклада

- Классические инструменты и их недостатки

План доклада

- Классические инструменты и их недостатки
- Современные нужды диагностики

План доклада

- Классические инструменты и их недостатки
- Современные нужды диагностики
- Наши наработки и примеры использования

План доклада

- Классические инструменты и их недостатки
- Современные нужды диагностики
- Наши наработки и примеры использования
- Мемы, боли, страдания

pg_stat_activity

pg_stat_activity

```
select * from pg_stat_activity
```

datid	5
datname	postgres
pid	75865
leader_pid	
...	
backend_start	2025-11-30 19:42:25.38302+03
xact_start	2025-11-30 19:42:31.780465+03
query_start	2025-11-30 19:42:31.780475+03
state_change	2025-11-30 19:42:31.780475+03
wait_event_type	IO
wait_event	DataFileRead
state	active
backend_xid	
backend_xmin	1007322
query	SELECT abalance FROM pgbench_accounts WHERE aid = \$1;

pg_stat_activity

```
select * from pg_stat_activity
```

datid	5
datname	postgres
pid	75865
leader_pid	
...	
backend_start	2025-11-30 19:42:25.38302+03
xact_start	2025-11-30 19:42:31.780465+03
query_start	2025-11-30 19:42:31.780475+03
state_change	2025-11-30 19:42:31.780475+03
wait_event_type	IO
wait_event	DataFileRead
state	active
backend_xid	
backend_xmin	1007322
query	SELECT abalance FROM pgbench_accounts WHERE aid = \$1;

pg_stat_activity

```
select * from pg_stat_activity
```

```
+-----+
datid      | 5
datname    | postgres
pid        | 75865
leader_pid |
...
backend_start | 2025-11-30 19:42:25.38302+03
xact_start   | 2025-11-30 19:42:31.780465+03
query_start  | 2025-11-30 19:42:31.780475+03
state_change | 2025-11-30 19:42:31.780475+03
wait_event_type | IO
wait_event   | DataFileRead
state        | active
backend_xid  |
backend_xmin | 1007322
query        | SELECT abalance FROM pgbench_accounts WHERE aid = $1;
```

pg_stat_activity

```
select * from pg_stat_activity
```

```
+-----+
datid          | 5
datname        | postgres
pid            | 75865
leader_pid     |
...
backend_start  | 2025-11-30 19:42:25.38302+03
xact_start     | 2025-11-30 19:42:31.780465+03
query_start    | 2025-11-30 19:42:31.780475+03
state_change   | 2025-11-30 19:42:31.780475+03
wait_event_type | IO
wait_event     | DataFileRead
state          | active
backend_xid    |
backend_xmin   | 1007322
query          | SELECT abalance FROM pgbench_accounts WHERE aid = $1;
```

pg_stat_activity

```
select * from pg_stat_activity
```

```
+-----+
datid          | 5
datname        | postgres
pid            | 75865
leader_pid     |
...
backend_start  | 2025-11-30 19:42:25.38302+03
xact_start     | 2025-11-30 19:42:31.780465+03
query_start    | 2025-11-30 19:42:31.780475+03
state_change   | 2025-11-30 19:42:31.780475+03
wait_event_type | IO
wait_event     | DataFileRead
state          | active
backend_xid    |
backend_xmin   | 1007322
query          | SELECT abalance FROM pgbench_accounts WHERE aid = $1;
```

pg_stat_activity

```
select * from pg_stat_activity
```

datid	5
datname	postgres
pid	75865
leader_pid	
...	
backend_start	2025-11-30 19:42:25.38302+03
xact_start	2025-11-30 19:42:31.780465+03
query_start	2025-11-30 19:42:31.780475+03
state_change	2025-11-30 19:42:31.780475+03
wait_event_type	IO
wait_event	DataFileRead
state	active
backend_xid	
backend_xmin	1007322
query	SELECT abalance FROM pgbench_accounts WHERE aid = \$1;

pg_stat_activity

- 👍 Полный перечень процессов: подключения, служебные, параллельные воркеры

pg_stat_activity

- 👍 Полный перечень процессов: подключения, служебные, параллельные воркеры
- 👍 Информация о том какие запросы выполняются:

pg_stat_activity

- 👍 Полный перечень процессов: подключения, служебные, параллельные воркеры
- 👍 Информация о том какие запросы выполняются:

```
query          | SELECT abalance FROM pgbench_accounts WHERE aid = $1;
```

pg_stat_activity

- 👍 Найти запрос который долго выполняется

```
postgres=# select now()-query_start as duration, query
postgres=#   from pg_stat_activity
postgres=#   order by query_start nulls last limit 1 \gx
-[ RECORD 1 ]-----
duration | 00:03:59.375816
query    | select pg_sleep(1000);
```

pg_stat_activity

- 👍 Найти запрос который долго выполняется

```
postgres=# select now()-query_start as duration, query
postgres=#   from pg_stat_activity
postgres=#   order by query_start nulls last limit 1 \gx
-[ RECORD 1 ]-----
duration | 00:03:59.375816
query    | select pg_sleep(1000);
```

pg_stat_activity

- 👍 Найти запрос который долго выполняется

```
postgres=# select now()-query_start as duration, query
postgres=#   from pg_stat_activity
postgres=#   order by query_start nulls last limit 1 \gx
-[ RECORD 1 ]-----
duration | 00:03:59.375816
query    | select pg_sleep(1000);
```

pg_stat_activity

- 👍 Найти запрос который долго выполняется

```
postgres=# select now()-query_start as duration, query
postgres=#   from pg_stat_activity
postgres=#   order by query_start nulls last limit 1 \gx
-[ RECORD 1 ]-----
duration | 00:03:59.375816
query    | select pg_sleep(1000);
```

pg_stat_activity

- 👍 Найти запрос который долго выполняется

```
postgres=# select now()-query_start as duration, query
postgres=#   from pg_stat_activity
postgres=#   order by query_start nulls last limit 1 \gx
-[ RECORD 1 ]-----
duration | 00:03:59.375816
query    | select pg_sleep(1000);
```

- 👍 Ожидания и их количество

pg_stat_activity

- 👍 Найти запрос который долго выполняется

```
postgres=# select now()-query_start as duration, query
postgres=#   from pg_stat_activity
postgres=#   order by query_start nulls last limit 1 \gx
-[ RECORD 1 ]-----
duration | 00:03:59.375816
query    | select pg_sleep(1000);
```

- 👍 Ожидания и их количество

```
wait_event      | DataFileRead
```

pg_stat_activity

- 🙅 Состояние на данный момент

pg_stat_activity

- 🙅 Состояние на данный момент
- 🙅 Неатомарное состояние: значения собираются не моментально

pg_stat_activity

- 🙅 Состояние на данный момент
- 🙅 Неатомарное состояние: значения собираются не моментально
- 🙅 Отсутствует время выполнения и частота запросов

pg_stat_statements

pg_stat_statements

```
select * from pg_stat_statements
```

```
-[ RECORD ]-----+----->
```

...

queryid	-278880456134375717
query	SELECT abalance FROM pgbench_accounts WHERE aid = \$1
plans	2560
total_plan_time	1109.1220739999965
mean_plan_time	0.43325081015624983
calls	13897659
total_exec_time	171703.87871069682
mean_exec_time	0.012354877804309718
rows	13897659
shared_blks_hit	55425830
shared_blks_read	323842
shared_blks_dirtied	1347
wal_records	1359
wal_fpi	1347
wal_bytes	10328391
stats_since	2025-11-26 20:02:51.687043+03
minmax_stats_since	2025-11-26 20:02:51.687043+03

pg_stat_statements

```
select * from pg_stat_statements
```

```
-[ RECORD ]-----+----->
```

...

queryid	-278880456134375717
query	SELECT abalance FROM pgbench_accounts WHERE aid = \$1
plans	2560
total_plan_time	1109.1220739999965
mean_plan_time	0.43325081015624983
calls	13897659
total_exec_time	171703.87871069682
mean_exec_time	0.012354877804309718
rows	13897659
shared_blks_hit	55425830
shared_blks_read	323842
shared_blks_dirtied	1347
wal_records	1359
wal_fpi	1347
wal_bytes	10328391
stats_since	2025-11-26 20:02:51.687043+03
minmax_stats_since	2025-11-26 20:02:51.687043+03

pg_stat_statements

```
select * from pg_stat_statements
```

```
-[ RECORD ]-----+----->
```

```
...
queryid          | -278880456134375717
query            | SELECT abalance FROM pgbench_accounts WHERE aid = $1
plans            | 2560
total_plan_time  | 1109.1220739999965
mean_plan_time   | 0.43325081015624983
calls            | 13897659
total_exec_time  | 171703.87871069682
mean_exec_time   | 0.012354877804309718
rows             | 13897659
shared_blks_hit  | 55425830
shared_blks_read | 323842
shared_blks_dirtied | 1347
wal_records      | 1359
wal_fpi          | 1347
wal_bytes        | 10328391
stats_since      | 2025-11-26 20:02:51.687043+03
minmax_stats_since | 2025-11-26 20:02:51.687043+03
```

pg_stat_statements

```
select * from pg_stat_statements
```

```
-[ RECORD ]-----+----->
```

```
...
```

queryid	-278880456134375717
query	SELECT abalance FROM pgbench_accounts WHERE aid = \$1
plans	2560
total_plan_time	1109.1220739999965
mean_plan_time	0.43325081015624983
calls	13897659
total_exec_time	171703.87871069682
mean_exec_time	0.012354877804309718
rows	13897659
shared_blks_hit	55425830
shared_blks_read	323842
shared_blks_dirtied	1347
wal_records	1359
wal_fpi	1347
wal_bytes	10328391
stats_since	2025-11-26 20:02:51.687043+03
minmax_stats_since	2025-11-26 20:02:51.687043+03

pg_stat_statements

```
select * from pg_stat_statements
```

```
-[ RECORD ]-----+----->
```

```
...
```

queryid	-278880456134375717
query	SELECT abalance FROM pgbench_accounts WHERE aid = \$1
plans	2560
total_plan_time	1109.1220739999965
mean_plan_time	0.43325081015624983
calls	13897659
total_exec_time	171703.87871069682
mean_exec_time	0.012354877804309718
rows	13897659
shared_blks_hit	55425830
shared_blks_read	323842
shared_blks_dirtied	1347
wal_records	1359
wal_fpi	1347
wal_bytes	10328391
stats_since	2025-11-26 20:02:51.687043+03
minmax_stats_since	2025-11-26 20:02:51.687043+03

pg_stat_statements

- 👍 Агрегация во времени в отличие от pg_stat_activity

```
postgres=# select shared_blks_hit, total_exec_time, total_plan_time, query from pg_stat_statements
postgres=# order by shared_blks_hit desc
postgres=# limit 1 \gx
-[ RECORD 1 ]---+-----
shared_blks_hit | 55425830
total_exec_time | 171703.87871069682
total_plan_time | 1109.1220739999965
query           | SELECT abalance FROM pgbench_accounts WHERE aid = $1
```

pg_stat_statements

- 👍 Агрегация во времени в отличие от pg_stat_activity
- 👍 Какие запросы больше всего кушают:

```
postgres=# select shared_blks_hit, total_exec_time, total_plan_time, query from pg_stat_statements
postgres=# order by shared_blks_hit desc
postgres=# limit 1 \gx
-[ RECORD 1 ]---+-----
shared_blks_hit | 55425830
total_exec_time | 171703.87871069682
total_plan_time | 1109.1220739999965
query           | SELECT abalance FROM pgbench_accounts WHERE aid = $1
```

pg_stat_statements

- 👍 Агрегация во времени в отличие от pg_stat_activity
- 👍 Какие запросы больше всего кушают:

```
postgres=# select shared_blks_hit, total_exec_time, total_plan_time, query from pg_stat_statements
postgres=# order by shared_blks_hit desc
postgres=# limit 1 \gx
-[ RECORD 1 ]---+-----
shared_blks_hit | 55425830
total_exec_time | 171703.87871069682
total_plan_time | 1109.1220739999965
query           | SELECT abalance FROM pgbench_accounts WHERE aid = $1
```

pg_stat_statements

- 👍 Агрегация во времени в отличие от pg_stat_activity
- 👍 Какие запросы больше всего кушают:

```
postgres=# select shared_blks_hit, total_exec_time, total_plan_time, query from pg_stat_statements
postgres=# order by shared_blks_hit desc
postgres=# limit 1 \gx
-[ RECORD 1 ]---+-----
shared_blks_hit | 55425830
total_exec_time | 171703.87871069682
total_plan_time | 1109.1220739999965
query           | SELECT abalance FROM pgbench_accounts WHERE aid = $1
```

pg_stat_statements

- 👍 Агрегация во времени в отличие от pg_stat_activity
- 👍 Какие запросы больше всего кушают:

```
postgres=# select shared_blks_hit, total_exec_time, total_plan_time, query from pg_stat_statements
postgres=# order by shared_blks_hit desc
postgres=# limit 1 \gx
-[ RECORD 1 ]---+-----
shared_blks_hit | 55425830
total_exec_time | 171703.87871069682
total_plan_time | 1109.1220739999965
query           | SELECT abalance FROM pgbench_accounts WHERE aid = $1
```

pg_stat_statements

- 💡⚠️ Замер метрик планирования стоит включать

```
postgres=# select name, setting, boot_val
postgres=#   from pg_settings
postgres=#   where name = 'pg_stat_statements.track_planning' \gx
-[ RECORD 1 ]-----
name       | pg_stat_statements.track_planning
setting    | on
boot_val   | off
```

pg_stat_statements

- 💡⚠️ Замер метрик планирования стоит включать

```
postgres=# select name, setting, boot_val
postgres=#   from pg_settings
postgres=#   where name = 'pg_stat_statements.track_planning' \gx
-[ RECORD 1 ]-----
name       | pg_stat_statements.track_planning
setting    | on
boot_val   | off
```

pg_stat_statements

- 💡⚠️ Замер метрик планирования стоит включать

```
postgres=# select name, setting, boot_val
postgres-#   from pg_settings
postgres-#   where name = 'pg_stat_statements.track_planning' \gx
-[ RECORD 1 ]-----
name       | pg_stat_statements.track_planning
setting    | on
boot_val   | off
```

pg_stat_statements

- 💡⚠️ Замер метрик планирования стоит включать

```
postgres=# select name, setting, boot_val
postgres-#   from pg_settings
postgres-#   where name = 'pg_stat_statements.track_planning' \gx
-[ RECORD 1 ]-----
name       | pg_stat_statements.track_planning
setting    | on
boot_val   | off
```

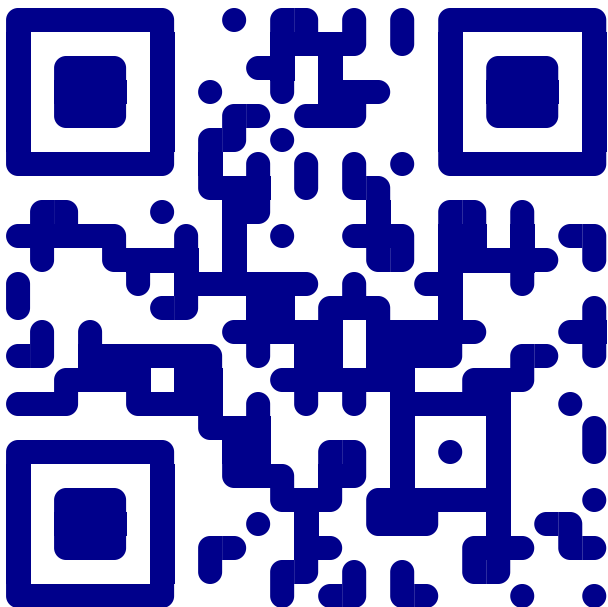
pg_stat_statements

- 🙅‍♂️ Агрегация за долгий период времени
- 🙅‍♂️ Агрегация за непонятный период времени (до PG 17)

Что была за проблема в 6 вечера вчера?



pg_profile



<> Code Issues 34 Pull requests 1 Discussions Actions Projects

pg_profile Public Watch 14 Fork 38 Starred 284

ma... Go to file + <> Code About

Postgres historic workload reports

Readme View license Activity 284 stars 14 watching 38 forks Report repository

Releases 23 Release 4.11 Latest 2 days ago + 22 releases

Contributors 5

Deployments 39 github-pages 3 weeks ago + 38 deployments

ma... Report example... 306f0fe · 3 weeks ago

data	add restartpoint...	3 weeks ago
doc	A little tweak in t...	3 weeks ago
expected	Issue #142 fixed.	3 weeks ago
grafana	grafana/profile_...	3 weeks ago
management	Issue #142 fixed.	3 weeks ago
migration	Issue #142 fixed.	3 weeks ago
privileges	Sizes collection ...	5 months ago
report	Issue #146 - thr...	3 weeks ago
report_examples	Report example...	3 weeks ago
sample	Fixed missed CT...	3 weeks ago

View all files

README License

pg_profile

This extension for PostgreSQL helps you to find out

pg_profile

```
tar xzf pg_profile--4.11.tar.gz ...
```

pg_profile

```
tar xzf pg_profile--4.11.tar.gz ...
```

```
CREATE EXTENSION dblink;  
CREATE EXTENSION pg_stat_statements;  
CREATE SCHEMA profile;  
CREATE EXTENSION pg_profile SCHEMA profile;
```

pg_profile

```
postgres=# select * from profile.take_sample();
```

server	result	elapsed
local	OK	00:00:01.09

(1 row)

```
postgres=# select * from profile.take_sample();
```

server	result	elapsed
local	OK	00:00:01.57

(1 row)

pg_profile

```
postgres=# select * from profile.take_sample();
server | result | elapsed
-----+-----+-----
local  | OK      | 00:00:01.09
(1 row)
```

```
postgres=# select * from profile.take_sample();
server | result | elapsed
-----+-----+-----
local  | OK      | 00:00:01.57
(1 row)
```

pg_profile

```
postgres=# select * from profile.take_sample();
```

server	result	elapsed
local	OK	00:00:01.09

(1 row)

```
postgres=# select * from profile.take_sample();
```

server	result	elapsed
local	OK	00:00:01.57

(1 row)

pg_profile

```
postgres=# select * from profile.take_sample();
 server | result |   elapsed
-----+-----+-----
  local | OK     | 00:00:01.09
(1 row)
```

```
postgres=# select * from profile.take_sample();
 server | result |   elapsed
-----+-----+-----
  local | OK     | 00:00:01.57
(1 row)
```

```
bash$ psql -Aqt "SELECT profile.get_report(11,12)" \
-o report_11_12.html
```

Report details

Server statistics

SQL query statistics

Top SQL by elapsed time

Top SQL by planning time

Top SQL by execution time

Top SQL by mean time

Top SQL by executions

Top SQL by shared blocks fetched

Top SQL by shared blocks read

Top SQL by shared blocks dirtied

Top SQL by shared blocks written

Top SQL by elapsed time

Query ID	Database	User	%Total	Time (s)			Plans	Executions	%Cvr
				Elapsed	Plan	Exec			
fc2137b3e54fc6db [caf79c9f7b]	postgres	mizhka	82.35	8.07		8.07	6	91928	11
SELECT abalance FROM pgbench_accounts WHERE aid = \$1									
d765a8f6c5d69d53 [9c71f9d8b1]	postgres	mizhka	16.03	1.57		1.57	1	1	99
c0e170b3804411d8 [871a150dcc]	testdb	mizhka	0.27	0.03	0.01	0.02	1	1	100
c0e170b3804411d8 [61a7c10aab]	postgres	mizhka	0.24	0.02	0.01	0.01	1	1	100
5ca1c924299891e [3bb0e9dadab]	postgres	mizhka	0.2	0.02		0.02	1	1	100
9efde8cd54d9d346 [711ac4ec16]	postgres	mizhka	0.18	0.02		0.02	1	1	
e78e3efd9edcc620 [d123620a7c]	postgres	mizhka	0.17	0.02		0.01	1	1	100
4dd87e7f9cd51b98 [fd1d75d066]	postgres	mizhka	0.16	0.02		0.02	1	1	
e78e3efd9edcc620 [f1013d123a]	testdb	mizhka	0.1	0.01		0.01	1	1	100

pg_profile

- 👍 Исторические отчёты

pg_profile

- 👍 Исторические отчёты
- 👍 Большое количество отчётов (по запросам, таблицам, индексам и прочему)

pg_profile

- 👍 Исторические отчёты
- 👍 Большое количество отчётов (по запросам, таблицам, индексам и прочему)
- 👍 Активно развивается!

pg_profile

- 👍 Исторические отчёты
- 👍 Большое количество отчётов (по запросам, таблицам, индексам и прочему)
- 👍 Активно развивается!
- 💡⚠️ Можно собирать из дополнительных расширений `pg_wait_sampling` и `pg_stat_kcache`

 Достаточно ли это для любых проблем? 

Особенность национальной техподдержки

collect.sh собрали?

с реплики или с мастера?

houston, we've got a problem

-  Ошибки, тормоза, повреждения данных

houston, we've got a problem

- 🔍 Ошибки, тормоза, повреждения данных
- 🛠️ Информацию о конфигурации базы, железе, операционки

houston, we've got a problem

- 🔍 Ошибки, тормоза, повреждения данных
- 🛠️ Информацию о конфигурации базы, железе, операционки
- 📝 Логи

houston, we've got a problem

- 🔍 Ошибки, тормоза, повреждения данных
- 🛠️ Информацию о конфигурации базы, железе, операционки
- 📝 Логи
- 🛠️ pg_profile

houston, we've got a problem

- 🔍 Ошибки, тормоза, повреждения данных
- 🛠️ Информацию о конфигурации базы, железе, операционки
- 📝 Логи
- 🛠️ pg_profile
- Помогает только в 90% случаях

houston, we've got a problem

- Не хватает истории инстанса (обновления, переключения, восстановления)

houston, we've got a problem

- Не хватает истории инстанса (обновления, переключения, восстановления)
- Следы багов из предыдущих версий, но использовалась ли старая версия?

houston, we've got a problem

- Не хватает истории инстанса (обновления, переключения, восстановления)
- Следы багов из предыдущих версий, но использовалась ли старая версия?
- `pg_resetwal` может приводить к повреждениям данных

houston, we've got a problem

- Не хватает истории инстанса (обновления, переключения, восстановления)
- Следы багов из предыдущих версий, но использовалась ли старая версия?
- `pg_resetwal` может приводить к повреждениям данных
- `pg_resetwal` не логируется никак

pg_operation_log

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

```
select * from pg_operation_log()
```

event	edition	version	lsn	last	count
bootstrap	ent	15.4.3	0/1000028	2023-10-25 16:51:00+03	1
pg_resetwal	ent	15.4.3	8/9B000028	2023-11-01 20:54:18+03	1
startup	ent	15.4.3	26F/97D8D078	2023-11-23 15:04:06+03	18
pg_resetwal	ent	15.5.1	53B/E0000028	2023-12-13 23:14:47+03	1
startup	ent	15.5.1	1011/96C4E188	2024-03-05 15:26:48+03	27
startup	ent	15.6.2	29DF/581FBE90	2024-07-10 13:52:45+03	74
startup	ent	15.6.3	2A9C/22B47800	2024-07-14 22:00:20+03	11
pg_upgrade	ent	16.3.1	2A9E/10000028	2024-07-14 22:40:41+03	1
startup	ent	16.3.1	2A9E/9424AF88	2024-07-14 23:33:20+03	3

pg_operation_log

- 👍 Историческая информация об использованных версиях, датах, LSN-ах

pg_operation_log

- 👍 Историческая информация об использованных версиях, датах, LSN-ах
- 👍 Хранятся запуски pg_resetwal-a / pg_upgrade:

pg_operation_log

- 👍 Историческая информация об использованных версиях, датах, LSN-ах
- 👍 Хранятся запуски pg_resetwal-a / pg_upgrade:

pg_resetwal		ent		15.4.3		8/9B000028		2023-11-01 20:54:18+03		1
-------------	--	-----	--	--------	--	------------	--	------------------------	--	---

pg_operation_log

- 👍 Историческая информация об использованных версиях, датах, LSN-ах
- 👍 Хранятся запуски pg_resetwal-a / pg_upgrade:

```
pg_resetwal | ent      | 15.4.3 | 8/9B000028 | 2023-11-01 20:54:18+03 | 1
```

- 👍 Хранятся о переключениях (promote):

pg_operation_log

- 👍 Историческая информация об использованных версиях, датах, LSN-ах
- 👍 Хранятся запуски pg_resetwal-a / pg_upgrade:

pg_resetwal		ent		15.4.3		8/9B000028		2023-11-01 20:54:18+03		1
-------------	--	-----	--	--------	--	------------	--	------------------------	--	---

- 👍 Хранятся о переключениях (promote):

promoted		ent		15.8.2		889B/FE000028		2024-12-25 23:17:36+03		1
----------	--	-----	--	--------	--	---------------	--	------------------------	--	---

pg_operation_log

- 👍 Историческая информация об использованных версиях, датах, LSN-ах
- 👍 Хранятся запуски pg_resetwal-a / pg_upgrade:

pg_resetwal		ent		15.4.3		8/9B000028		2023-11-01 20:54:18+03		1
-------------	--	-----	--	--------	--	------------	--	------------------------	--	---

- 👍 Хранятся о переключениях (promote):

promoted		ent		15.8.2		889B/FE000028		2024-12-25 23:17:36+03		1
----------	--	-----	--	--------	--	---------------	--	------------------------	--	---

- ⚡ Доступно в PostgresPro Standard и Enterprise

Папа, у вас ~~мальчик~~ segfault

segfault

- 🌟 segfault в логах

segfault

- 🌟 segfault в логах
- Попытка доступа недоступной памяти

segfault

- 🌟 segfault в логах
- Попытка доступа недоступной памяти
- Требуется дампа памяти (место в коде, значения переменных)

segfault

- 🌟 segfault в логах
- Попытка доступа недоступной памяти
- Требуется дампа памяти (место в коде, значения переменных)
- "Мы что-то должны ещё вам прислать? у нас больше ничего нет"

segfault

- Логи ядра Linux

segfault

- Логи ядра Linux

```
[Fri Nov 21 00:30:13 2025] postgres[951277]: segfault at 6488daffb594 ip 00006488daffb594 sp 00007ffe0cb9e270
error 15 in postgres[6488daefe000+29d000] likely on CPU 15 (core 7, socket 1)
[Fri Nov 21 00:30:13 2025] Code:
    b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 80 1f
    b9 ff 40 1f b9 ff 70 1f b9 ff <e6> 23 b9 ff d0 23 b9 ff ba 23 b9 ff a4 23 b9 ff 8e 23 b9 ff 78 23
```

segfault

- Логи ядра Linux

```
[Fri Nov 21 00:30:13 2025] postgres[951277]: segfault at 6488daffb594 ip 00006488daffb594 sp 00007ffe0cb9e270
error 15 in postgres[6488daefe000+29d000] likely on CPU 15 (core 7, socket 1)
[Fri Nov 21 00:30:13 2025] Code:
    b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 80 1f
    b9 ff 40 1f b9 ff 70 1f b9 ff <e6> 23 b9 ff d0 23 b9 ff ba 23 b9 ff a4 23 b9 ff 8e 23 b9 ff 78 23
```

- Логи PostgreSQL

segfault

■ Логи ядра Linux

```
[Fri Nov 21 00:30:13 2025] postgres[951277]: segfault at 6488daffb594 ip 00006488daffb594 sp 00007ffe0cb9e270
error 15 in postgres[6488daefe000+29d000] likely on CPU 15 (core 7, socket 1)
[Fri Nov 21 00:30:13 2025] Code:
    b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 80 1f
    b9 ff 40 1f b9 ff 70 1f b9 ff <e6> 23 b9 ff d0 23 b9 ff ba 23 b9 ff a4 23 b9 ff 8e 23 b9 ff 78 23
```

■ Логи PostgreSQL

```
2025-11-21 03:50:55 MSK LOG:  server process (PID 1591592) was terminated by signal 11: Segmentation fault
```

segfault

- Логи ядра Linux

```
[Fri Nov 21 00:30:13 2025] postgres[951277]: segfault at 6488daffb594 ip 00006488daffb594 sp 00007ffe0cb9e270
error 15 in postgres[6488daefe000+29d000] likely on CPU 15 (core 7, socket 1)
[Fri Nov 21 00:30:13 2025] Code:
    b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 10 1f b9 ff 80 1f
    b9 ff 40 1f b9 ff 70 1f b9 ff <e6> 23 b9 ff d0 23 b9 ff ba 23 b9 ff a4 23 b9 ff 8e 23 b9 ff 78 23
```

- Логи PostgreSQL

```
2025-11-21 03:50:55 MSK LOG:  server process (PID 1591592) was terminated by signal 11: Segmentation fault
```

- PostgreSQL полностью перезагружается при SEGFAULT-ax

segfault

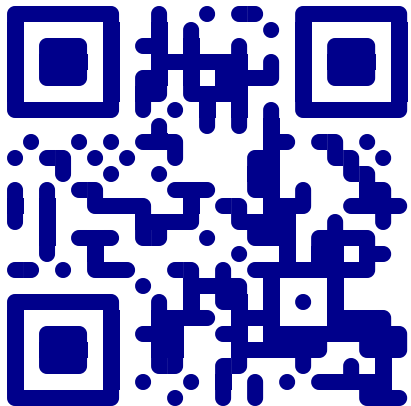
- Настроить coredump

segfault

- Настроить coredump
- <https://support.postgrespro.ru/note/36>

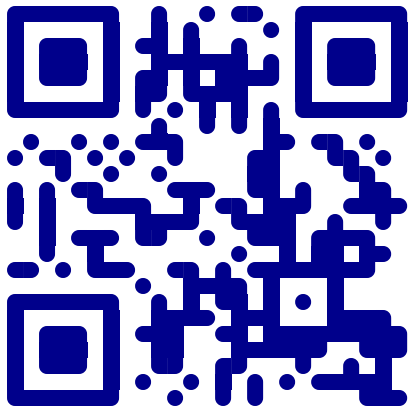
segfault

- Настроить coredump
- <https://support.postgrespro.ru/note/36>



segfault

- Настроить coredump
- <https://support.postgrespro.ru/note/36>



- 🙄 Часто либо не настроен, либо не сработал, либо проблема с правами

segfault

- Java HotSpot Errors: hs_err_pid<PID>



segfault

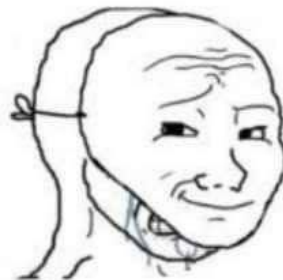
- Java HotSpot Errors: hs_err_pid<PID>

```
#
# A fatal error has been detected by the Java Runtime Environment:
#
# Internal Error (relocInfo.cpp:530), pid=5888, tid=0x000045a2e0a7ffe0
# guarantee(sect  $\neq$  CodeBuffer::SECT_NONE) failed: lost track of this address
#
# JRE version: OpenJDK Runtime Environment (8.0_152-b16) (build 1.8.0_152-b16)
# Java VM: OpenJDK 64-Bit Server VM (RVM 3.4.0) (25.152-b16 mixed mode linux- )
# mercurial id c4c786060952 2017-12-25 08:58 +0700
# Failed to write core dump. Core dumps have been disabled.
#
# If you would like to submit a bug report, please visit:
#   http://bugreport.java.com/bugreport/crash.jsp
#
```

T H R E A D

```
Current thread (0x00008b465e090000):  JavaThread "O3 CompilerThread0" daemon [_thread_in_vm, id=5901, stack( ... )]
```

Дизайнеры



Смотри, у нас
одинаковые идеи



Нееет! Ты украл
мою работу

💡 А может сделаем что-то похожее?

Программисты



Я украл
твой код



Это не мой

crash_info aka pg_diag_dump

crash_info

- crash_info: аналог Java HotSpot Errors

crash_info

- crash_info: аналог Java HotSpot Errors
- `$PGDATA/crash_info/crash_<TIMESTAMP>_pid<PID>.log`

crash_info

- crash_info: аналог Java HotSpot Errors
- `$PGDATA/crash_info/crash_<TIMESTAMP>_pid<PID>.log`
- Ловим сигнал `SEGFAULT` и вызываем свой обработчик

crash_info

■ Стек вызовов

```
# Backtrace
1  postgres + 0x719940          0x560529b7cd37  0x00007ffd01c8cc30  WriteBacktraceSection + 0xac7
   bt_crash_handler crash_info/crash_handler.c:524
2  libc.so.6 + 0x3cbdf         0x7ff263e63be0  0x00007ffd01c8cd00  __sigaction + 0x50
3  oracle_fdw.so + 0xb960      0x7ff26095d905  0x00007ffd01c8d9f0  _PG_init + 0xffffffff
   getColumnData /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2936
   getFdwState /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2842
4  oracle_fdw.so + 0x126d0     0x7ff260963741  0x00007ffd01c8dae0  _PG_init + 0x1691
   oracleGetForeignRelSize /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:824
5  postgres + 0x447a40        0x5605298abdd2  0x00007ffd01c8db80  set_function_size_estimates + 0x1522
   set_foreign_size optimizer/path/allpaths.c:926
   set_rel_size optimizer/path/allpaths.c:398
6  postgres + 0x448fd0        0x5605298ac098  0x00007ffd01c8dc80  make_one_rel + 0xc8
   set_base_rel_sizes optimizer/path/allpaths.c:300
   make_one_rel optimizer/path/allpaths.c:189
...
```

crash_info

■ Стек вызовов

```
# Backtrace
1  postgres + 0x719940          0x560529b7cd37  0x00007ffd01c8cc30  WriteBacktraceSection + 0xac7
   bt_crash_handler crash_info/crash_handler.c:524
2  libc.so.6 + 0x3cbdf          0x7ff263e63be0  0x00007ffd01c8cd00  __sigaction + 0x50
3  oracle_fdw.so + 0xb960        0x7ff26095d905  0x00007ffd01c8d9f0  _PG_init + 0xffffffff
   getColumnData /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2936
   getFdwState /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2842
4  oracle_fdw.so + 0x126d0       0x7ff260963741  0x00007ffd01c8dae0  _PG_init + 0x1691
   oracleGetForeignRelSize /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:824
5  postgres + 0x447a40          0x5605298abdd2  0x00007ffd01c8db80  set_function_size_estimates + 0x1522
   set_foreign_size optimizer/path/allpaths.c:926
   set_rel_size optimizer/path/allpaths.c:398
6  postgres + 0x448fd0          0x5605298ac098  0x00007ffd01c8dc80  make_one_rel + 0xc8
   set_base_rel_sizes optimizer/path/allpaths.c:300
   make_one_rel optimizer/path/allpaths.c:189
...
```

crash_info

■ Стек вызовов

```
# Backtrace
1  postgres + 0x719940          0x560529b7cd37  0x00007ffd01c8cc30  WriteBacktraceSection + 0xac7
   bt_crash_handler crash_info/crash_handler.c:524
2  libc.so.6 + 0x3cbdf          0x7ff263e63be0  0x00007ffd01c8cd00  __sigaction + 0x50
3  oracle_fdw.so + 0xb960        0x7ff26095d905  0x00007ffd01c8d9f0  _PG_init + 0xffffffff
   getColumnData /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2936
   getFdwState /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2842
4  oracle_fdw.so + 0x126d0       0x7ff260963741  0x00007ffd01c8dae0  _PG_init + 0x1691
   oracleGetForeignRelSize /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:824
5  postgres + 0x447a40          0x5605298abdd2  0x00007ffd01c8db80  set_function_size_estimates + 0x1522
   set_foreign_size optimizer/path/allpaths.c:926
   set_rel_size optimizer/path/allpaths.c:398
6  postgres + 0x448fd0          0x5605298ac098  0x00007ffd01c8dc80  make_one_rel + 0xc8
   set_base_rel_sizes optimizer/path/allpaths.c:300
   make_one_rel optimizer/path/allpaths.c:189
...
```

crash_info

■ Стек вызовов

```
# Backtrace
1  postgres + 0x719940          0x560529b7cd37  0x00007ffd01c8cc30  WriteBacktraceSection + 0xac7
   bt_crash_handler crash_info/crash_handler.c:524
2  libc.so.6 + 0x3cbdf         0x7ff263e63be0  0x00007ffd01c8cd00  __sigaction + 0x50
3  oracle_fdw.so + 0xb960       0x7ff26095d905  0x00007ffd01c8d9f0  _PG_init + 0xffffffff
   getColumnData /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2936
   getFdwState /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2842
4  oracle_fdw.so + 0x126d0      0x7ff260963741  0x00007ffd01c8dae0  _PG_init + 0x1691
   oracleGetForeignRelSize /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:824
5  postgres + 0x447a40         0x5605298abdd2  0x00007ffd01c8db80  set_function_size_estimates + 0x1522
   set_foreign_size optimizer/path/allpaths.c:926
   set_rel_size optimizer/path/allpaths.c:398
6  postgres + 0x448fd0         0x5605298ac098  0x00007ffd01c8dc80  make_one_rel + 0xc8
   set_base_rel_sizes optimizer/path/allpaths.c:300
   make_one_rel optimizer/path/allpaths.c:189
...
```

crash_info

■ Стек вызовов

```
# Backtrace
1  postgres + 0x719940          0x560529b7cd37  0x00007ffd01c8cc30  WriteBacktraceSection + 0xac7
   bt_crash_handler crash_info/crash_handler.c:524
2  libc.so.6 + 0x3cbdf         0x7ff263e63be0  0x00007ffd01c8cd00  __sigaction + 0x50
3  oracle_fdw.so + 0xb960       0x7ff26095d905  0x00007ffd01c8d9f0  _PG_init + 0xffffffff
   getColumnData /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2936
   getFdwState /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2842
4  oracle_fdw.so + 0x126d0      0x7ff260963741  0x00007ffd01c8dae0  _PG_init + 0x1691
   oracleGetForeignRelSize /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:824
5  postgres + 0x447a40         0x5605298abdd2  0x00007ffd01c8db80  set_function_size_estimates + 0x1522
   set_foreign_size optimizer/path/allpaths.c:926
   set_rel_size optimizer/path/allpaths.c:398
6  postgres + 0x448fd0         0x5605298ac098  0x00007ffd01c8dc80  make_one_rel + 0xc8
   set_base_rel_sizes optimizer/path/allpaths.c:300
   make_one_rel optimizer/path/allpaths.c:189
...
```

crash_info

- Регистры

crash_info

■ Регистры

```
# Context
RAX = 0000000000000000
RBX = 0000000000000000
RCX = 00000000000000c8
RDX = 0000000000000079
RBP = 000056379bbe5bc8
RSP = 00007ffe980dc920
RSI = 00007f30dce2c55c
RDI = 000056379bbe5bd4

...

RIP = 00007f30dce1a905
```

crash_info

- Версии

crash_info

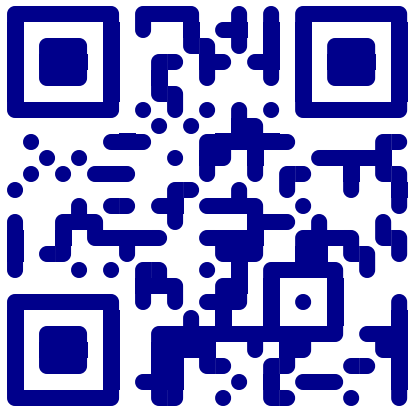
■ Версии

```
# Content of /etc/os-release
NAME="RED OS"
VERSION="8.0"
PLATFORM_ID="platform:red80"
ID="redos"
ID_LIKE="rhel centos fedora"
VERSION_ID="8.0"
PRETTY_NAME="RED OS 8.0"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:redos:redos:8"
HOME_URL="https://redos.red-soft.ru"
BUG_REPORT_URL="https://support.red-soft.ru"
EDITION="Standard"

Postgres Pro (enterprise) 16.6.1 on x86_64-pc-linux-gnu,
compiled by gcc (GCC) 12.4.1 20240730 (RED SOFT 12.4.0-1), 64-bit
```

crash_info

- **crash_info** - ВКЛ / ВЫКЛ
- **crash_info_dump** - что писать
- **crash_info_location** - куда писать
- Документация



crash_info: пример #1

пример #1: oracle_fdw.so

```
# Backtrace
1  postgres + 0x719940          0x560529b7cd37  0x00007ffd01c8cc30  WriteBacktraceSection + 0xac7
   bt_crash_handler  crash_info/crash_handler.c:524
2  libc.so.6 + 0x3cbdf         0x7ff263e63be0  0x00007ffd01c8cd00  __sigaction + 0x50
3  oracle_fdw.so + 0xb960       0x7ff26095d905  0x00007ffd01c8d9f0  _PG_init + 0xffffffff
   getColumnData    /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2936
   getFdwState      /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:2842
4  oracle_fdw.so + 0x126d0      0x7ff260963741  0x00007ffd01c8dae0  _PG_init + 0x1691
   oracleGetForeignRelSize /usr/src/debug/oracle-fdw-ent-16-2.7.0-1.red80.x86_64/oracle_fdw.c:824
5  postgres + 0x447a40         0x5605298abdd2  0x00007ffd01c8db80  set_function_size_estimates + 0x1522
   set_foreign_size  optimizer/path/allpaths.c:926
   set_rel_size     optimizer/path/allpaths.c:398
6  postgres + 0x448fd0         0x5605298ac098  0x00007ffd01c8dc80  make_one_rel + 0xc8
   set_base_rel_sizes optimizer/path/allpaths.c:300
   make_one_rel      optimizer/path/allpaths.c:189
...
```

пример #1: oracle_fdw.so

```
/* loop through foreign table columns */
index = 0;
for (i=0; i<tupdesc->natts; ++i)
{
    ...
    ++index;
    ...

    foreach(option, options)
    {
        ...
        if (strcmp(def->defname, OPT_KEY) == 0 && getBoolVal(def))
        {
            /* mark the column as primary key column */
            oraTable->cols[index-1]->pkey = 1;
        }
        else if (strcmp(def->defname, OPT_STRIP_ZEROS) == 0 && getBoolVal(def))
            oraTable->cols[index-1]->strip_zeros = 1;
    }
}
```

пример #1: oracle_fdw.so

```
/* loop through foreign table columns */
index = 0;
for (i=0; i<tupdesc→natts; ++i)
{
    ...
    ++index;
    ...

    foreach(option, options)
    {
        ...
        if (strcmp(def→defname, OPT_KEY) == 0 && getBoolVal(def))
        {
            /* mark the column as primary key column */
            oraTable→cols[index-1]→pkey = 1;
        }
        else if (strcmp(def→defname, OPT_STRIP_ZEROS) == 0 && getBoolVal(def))
            oraTable→cols[index-1]→strip_zeros = 1;
    }
}
```

пример #1: oracle_fdw.so

```
/* loop through foreign table columns */
index = 0;
for (i=0; i<tupdesc->natts; ++i)
{
    ...
    ++index;
    ...

    foreach(option, options)
    {
        ...
        if (strcmp(def->defname, OPT_KEY) == 0 && getBoolVal(def))
        {
            /* mark the column as primary key column */
            oraTable->cols[index-1]->pkey = 1;
        }
        else if (strcmp(def->defname, OPT_STRIP_ZEROS) == 0 && getBoolVal(def))
            oraTable->cols[index-1]->strip_zeros = 1;
    }
}
```

пример #1: oracle_fdw.so

```
/* loop through foreign table columns */
index = 0;
for (i=0; i<tupdesc->natts; ++i)
{
    ...
    ++index;
    ...

    foreach(option, options)
    {
        ...
        if (strcmp(def->defname, OPT_KEY) == 0 && getBoolVal(def))
        {
            /* mark the column as primary key column */
            oraTable->cols[index-1]->pkey = 1;
        }
        else if (strcmp(def->defname, OPT_STRIP_ZEROS) == 0 && getBoolVal(def))
            oraTable->cols[index-1]->strip_zeros = 1;
    }
}
```

пример #1: oracle_fdw.so

- Создать таблицу, настроить FDW

пример #1: oracle_fdw.so

- Создать таблицу, настроить FDW
- Oracle: `alter table dbo.oratab drop column id;`

пример #1: oracle_fdw.so

- Создать таблицу, настроить FDW
- Oracle: alter table dbo.oratab drop column id;
- PostgreSQL: select * from oratab;

пример #1: oracle_fdw.so

- Создать таблицу, настроить FDW
- Oracle: alter table dbo.oratab drop column id;
- PostgreSQL: select * from oratab;
- База данных падает

Commit 4b44505



laurenz committed on Mar 28

Verified

Fix crash with column options on non-existing columns

If the Oracle table has fewer columns than the PostgreSQL foreign table and one of these extra columns has a column option set, `oracle_fdw` would write past the end of an array, leading to memory corruption and crashes.

Fix by ignoring options for such columns. That may cause other errors further down the line, but it seems premature to error out right away, because some operations on such a table could still succeed.

This closes [#744](#), reported by "ganshinm". Analysis by "ziva777".



master



ORACLE_FDW_2_8_0

1 parent [2b1af12](#) commit 4b44505

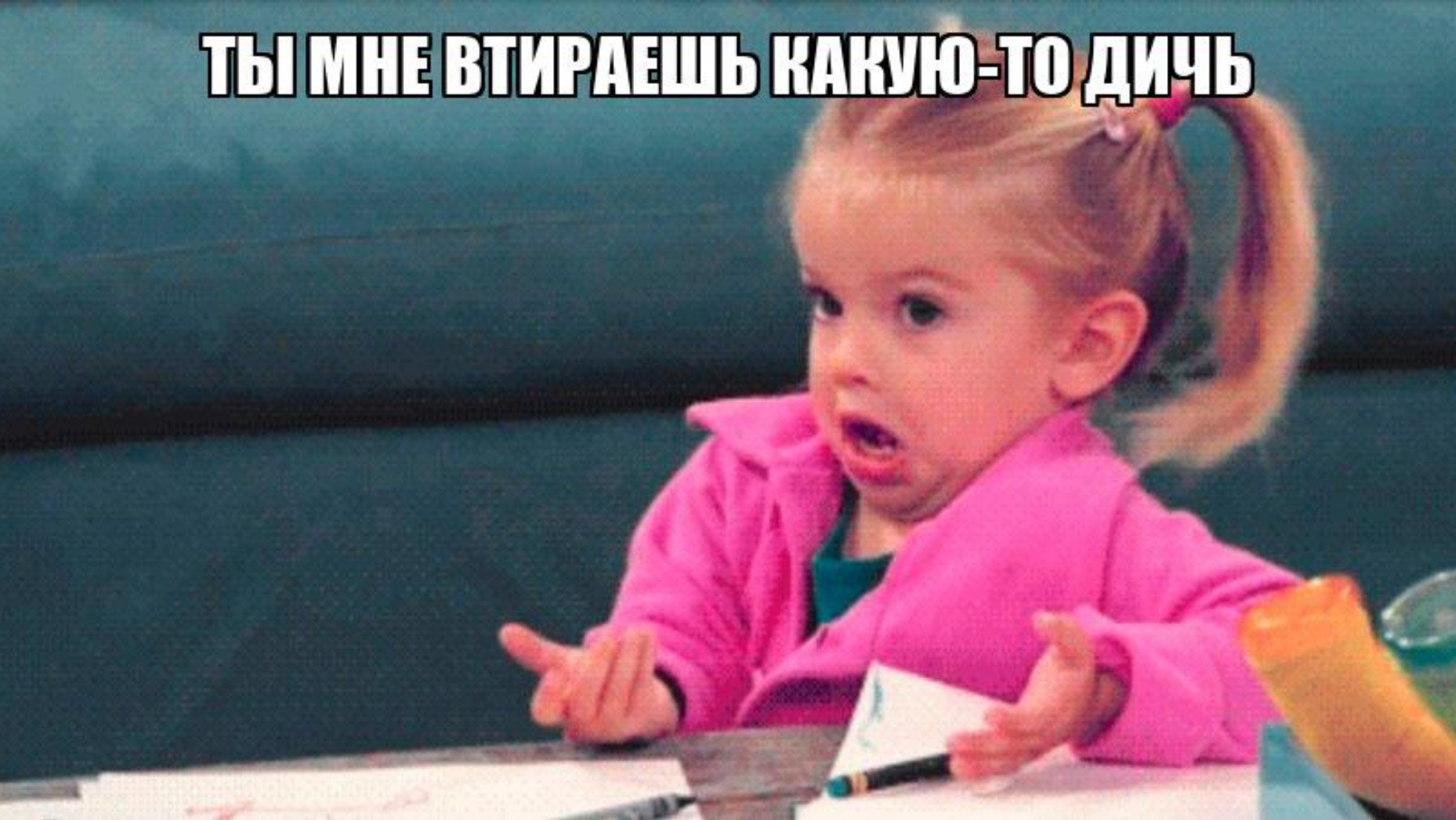


crash_info: пример #2

 **crash_info ИНОГДА спасает Вашу базу данных от SEGFAULT-ов** 🤯

- * незапланированная особенность реализации crash_info
- * и только для SEGFAULT-ов из-за дефекта Linux версий 6.1 - 6.8

ТЫ МНЕ ВТИРАЕШЬ КАКУЮ-ТО ДИЧЬ



Linux bug

- DBA находит 3 файла crash_info
 - `crash_1764605313879686_pid2130208.log`
 - `crash_1764605308648131_pid1867399.log`
 - `crash_1764605308641242_pid2130207.log`

Linux bug

- DBA находит 3 файла crash_info
 - crash_1764605313879686_pid2130208.log
 - crash_1764605308648131_pid1867399.log
 - crash_1764605308641242_pid2130207.log
- Упало создание индексов
 - program_invocation_name: postgres: mydb: postgres myuser local CREATE INDEX

Linux bug

- DBA находит 3 файла crash_info
 - crash_1764605313879686_pid2130208.log
 - crash_1764605308648131_pid1867399.log
 - crash_1764605308641242_pid2130207.log
- Упало создание индексов
 - program_invocation_name: postgres: mydb: postgres myuser local CREATE INDEX
- ВНИМАНИЕ!!! Никто не пострадал

Linux bug

- DBA находит 3 файла crash_info
 - crash_1764605313879686_pid2130208.log
 - crash_1764605308648131_pid1867399.log
 - crash_1764605308641242_pid2130207.log
- Упало создание индексов
 - program_invocation_name: postgres: mydb: postgres myuser local CREATE INDEX
- ВНИМАНИЕ!!! Никто не пострадал
- Индекс создался, в логах пусто и база не перезагружалась!

**ГОВОРЯТ SEGFAULT
НЕ НАСТОЯЩИЙ!**



Linux bug

- Без crash_info

Linux bug

- Без crash_info
- Случается SEGFAULT

Linux bug

- Без crash_info
- Случается SEGFAULT
- Инструкция процессора по адресу записанному в регистр RIP

Linux bug

- Без crash_info
- Происходит SEGFAULT
- Инструкция процессора по адресу записанному в регистр RIP
- Ядерный обработчик логирует, делает coredump и убивается процесс

Linux bug

- В случаи crash_info

Linux bug

- В случаи `crash_info`
- Устанавливаем свой обработчик

Linux bug

- В случаи `crash_info`
- Устанавливаем свой обработчик
- Делаем полезное действие

Linux bug

- В случаи `crash_info`
- Устанавливаем свой обработчик
- Делаем полезное действие
- Потом возвращаем старый обработчик и заканчиваем обработку

Linux bug

- Повторяется инструкция!
- И повторяется SEGFAULT
- А иногда и не повторяется: время прошло
- `crash_info` делает вторую попытку выполнить инструкцию

Linux bug

- Дефект Linux 6.1 - 6.8

Linux bug

- Дефект Linux 6.1 - 6.8
- Давление памяти (swap закончился)

Linux bug

- Дефект Linux 6.1 - 6.8
- Давление памяти (swar закончился)
- Страница памяти может становиться обнуленной **на некоторое время**

Linux bug

- Дефект Linux 6.1 - 6.8
- Давление памяти (swap закончился)
- Страница памяти может становиться обнуленной **на некоторое время**
- Подмена через маппинг виртуальной памяти

Linux bug

- Дефект Linux 6.1 - 6.8
- Давление памяти (swap закончился)
- Страница памяти может становиться обнуленной **на некоторое время**
- Подмена через маппинг виртуальной памяти
- На 6.9 дефект не воспроизводится

Linux bug

- Дефект Linux 6.1 - 6.8
- Давление памяти (swar закончился)
- Страница памяти может становиться обнуленной **на некоторое время**
- Подмена через маппинг виртуальной памяти
- На 6.9 дефект не воспроизводится
- Больше никакой определенности и деталей

Бонус

Бонус

-  Генерировать файл для живых процессов

Бонус

-  Генерировать файл для живых процессов
- 40ой сигнал

Бонус

-  Генерировать файл для живых процессов
- 40ой сигнал
- Различать файлы от SEGFAULT
 - расширение .state вместо .log

Бонус

- 💡 Генерировать файл для живых процессов
- 40ой сигнал
- Различать файлы от SEGFAULT
 - расширение .state вместо .log
- SQL функция: `select pg_diagdump(PID)`

PGMeetup.VLD 2025, 20:30

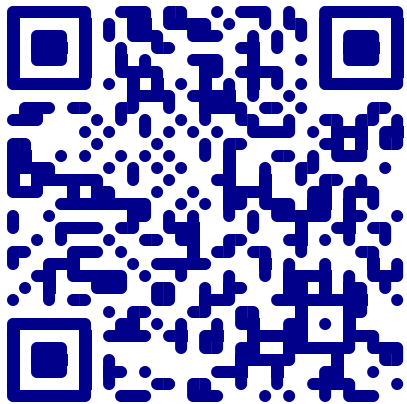
Андрей Билле

**Если ваш админ сказочный самурай или
история о восстановлении очень нужных
данных**



pg_uprobe

- pg_uprobe - расширение для трассирования запросов
- https://github.com/postgrespro/pg_uprobe
- Версия 0.3 - для тестирования, но не для продакшена



- Готовы ответить на любые вопросы и рады обратной связи

Итоги этой болтовни

- На живой проблеме сначала смотрим `pg_stat_activity`

Итоги этой болтовни

- На живой проблеме сначала смотрим `pg_stat_activity`
- `pg_profile` должен быть установлен (ну или другой аналог если они есть)

Итоги этой болтовни

- На живой проблеме сначала смотрим `pg_stat_activity`
- `pg_profile` должен быть установлен (ну или другой аналог если они есть)
- `pg_operation_log` и `crash_info` - наш Swiss Knife

Итоги этой болтовни

- На живой проблеме сначала смотрим `pg_stat_activity`
- `pg_profile` должен быть установлен (ну или другой аналог если они есть)
- `pg_operation_log` и `crash_info` - наш Swiss Knife
- `crash_info` может "лечить" `segfault`-ы

Итоги этой болтовни

- На живой проблеме сначала смотрим `pg_stat_activity`
- `pg_profile` должен быть установлен (ну или другой аналог если они есть)
- `pg_operation_log` и `crash_info` - наш Swiss Knife
- `crash_info` может "лечить" `segfault`-ы
- Stay tuned! Мы только начинаем сегодняшний митап

Итоги этой болтовни

- На живой проблеме сначала смотрим `pg_stat_activity`
- `pg_profile` должен быть установлен (ну или другой аналог если они есть)
- `pg_operation_log` и `crash_info` - наш Swiss Knife
- `crash_info` может "лечить" `segfault`-ы
- Stay tuned! Мы только начинаем сегодняшний митап
- Давайте общаться!

Stay tuned! Мы только начинаем
сегодняшний митап

А данный доклад закончился

Вопросы?

