PostgresPro

ОПЫТ ВНЕДРЕНИЯ SDL-ПРАКТИК в POSTGRES PROFESSIONAL

Панченко Иван Евгеньевич
заместитель директора
Postgres Professional
Попов Валерий Викторович
группа ИБ и сертификации

О КОМПАНИИ

Postgres Pro – российский разработчик систем управления базами данных на основе PostgreSQL.

5 лет
на рынке с 2015 г.

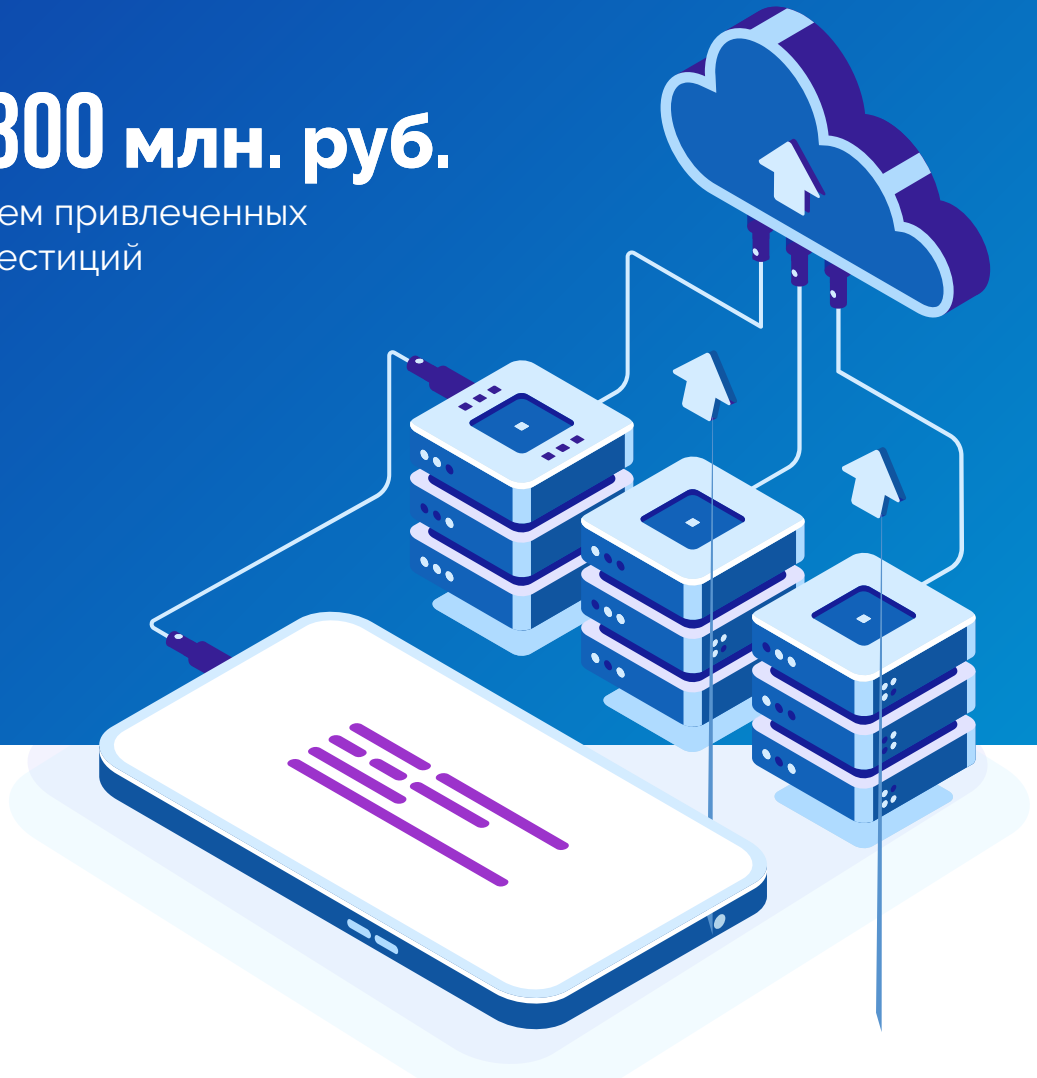
>20 лет
опыта в разработке PostgreSQL

>300 млн. руб.
объем привлеченных инвестиций

>70 чел.
штат компании

включая ведущих разработчиков (major contributor) PostgreSQL и **2х коммитеров** (committer), имеющих право вносить изменения в ядро PostgreSQL.

>90 патчей
вносят сотрудники компании в каждый релиз PostgreSQL



СУБД POSTGRES PRO

СУБД Postgres Pro – первый в России коммерческий продукт на основе PostgreSQL.
Входит в Единый реестр отечественных программ и баз данных Минкомсвязи.

Standard
Современная СУБД, включает все новые функции PostgreSQL и полезные доработки от компании

Enterprise
Наиболее полнофункциональная СУБД с высокой производительностью и масштабируемостью

Certified
Сертифицированные ФСТЭК версии Standard и Enterprise

PostgresPro

POSTGRES в сравнении...

Top 50 Vendors By Total Number Of "Distinct" Vulnerabilities

Go to year: 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2

	Vendor Name	Number of Products	Number of Vulnerabilities	#Vulnerabilities/#Products
1	Microsoft	519	6417	12
2	Oracle	632	5888	9
3	Apple	118	4498	38
46	Mysql	8	243	30

Vendor	Product	BDU	CVE
Oracle Corp.	Database Server	51	442
Microsoft Corp.	Microsoft SQL Server	4	86
Oracle Corp.	MySQL	291	716
PGDG	PostgreSQL	35	112

Projects on Coverity Scan

Can't Find Your Project on the List?

Register a new project | Register my GitHub project

Postgres | Search | Show all

Did you know LibreOffice reduced its defect density from 1.1 to 0.08, fixing 6000 defects found by Coverity Scan? Read more.

Interested in a specific programming language?

- Java
- C/C++
- C#
- JavaScript
- PHP/Python/Ruby

Project	Lines of code analyzed	Language
PostgreSQL_REL_12_STABLE	1,621,427	C/C++
PostgreSQL_REL_13_STABLE	1,645,091	C/C++
Postgres	1,203,170	C/C++
Postgres Pro	1,323,387	C/C++

NIST

Information Technology Laboratory

NATIONAL VULNERABILITY DATABASE

PostgresPro

POSTGRES в сравнении...

LCOV - code coverage report

Current view: top level - src/backend/parser
Test: PostgreSQL 14devel
Date: 2020-09-18 10:06:36
Legend: Rating: low: < 75 % | medium: >= 75 % | high: >= 90 %

	Hit	Total	Coverage
Lines:	18606	21477	86.6 %
Functions:	418	427	97.9 %

Filename	Line Coverage		Functions	
analyze.c	90.2 %	814 / 902	96.8 %	30 / 31
gram.c	87.5 %	3782 / 4321	100.0 %	2 / 2
gram.y	87.3 %	5732 / 6564	97.5 %	39 / 40
parse_agg.c	75.9 %	558 / 735	100.0 %	22 / 22
parse_clause.c	92.9 %	931 / 1002	100.0 %	40 / 40
parse_coerce.c	91.0 %	774 / 851	100.0 %	26 / 26
parse_collate.c	94.5 %	256 / 271	100.0 %	10 / 10
parse_cte.c	88.2 %	284 / 322	100.0 %	9 / 9
parse_enr.c	100.0 %	4 / 4	100.0 %	2 / 2
parse_expr.c	72.6 %	963 / 1327	94.9 %	37 / 39
parse_func.c	85.7 %	670 / 782	100.0 %	14 / 14
parse_node.c	94.9 %	150 / 158	100.0 %	9 / 9
parse_oper.c	92.4 %	257 / 278	94.7 %	18 / 19
parse_param.c	88.7 %	94 / 106	100.0 %	9 / 9
parse_relation.c	86.4 %	969 / 1121	100.0 %	53 / 53
parse_target.c	85.0 %	482 / 567	100.0 %	20 / 20
parse_type.c	80.6 %	229 / 284	92.0 %	23 / 25
parse_utilcmd.c	88.9 %	1327 / 1493	100.0 %	28 / 28
parser.c	94.0 %	173 / 184	100.0 %	6 / 6
scan.l	86.4 %	114 / 132	88.9 %	16 / 18
scansup.c	58.9 %	43 / 73	100.0 %	5 / 5

Generated by: LCOV version 1.13

Regression tests + VALGRIND

#define USE_VALGRIND in src/include/pg_config_manual.h

+ src/tools/valgrind.supp – подавление ненужных ошибок

make installcheck - ловит неинициализированную память,

PostgreSQL Build Farm Log

Details for system "valgrind", status 'OK', snapshot taken 2020-09-16 18:35:56

System Information	Farm member	Branch	OS	Compiler	Architecture	Owner
	valgrind	REL_11_1C_DEV	Debian 10	gcc 8.3.0	x86_64	v.wagner [a t] postgrespro.ru

Other branches	REL9_6_1C_DEV	REL_12_1C_DEV	REL_10_1C_DEV	PGPROEE9_6_DEV	PGPROEE12_DEV	PGPROEE12_3518_pg_fdw	PGPROEE11_DEV	PGPROEE11_4062	PGPROEE10_DEV

Stage Logs Total run time 03:33:51				
	SCM-checkout (00:00:22)	configure (00:00:21)	make (00:03:51)	check (00:00:42)
	make-contrib (00:00:30)	contrib-check (00:03:27)	pl-check (00:00:21)	make-testmodules (00:00:02)
	make-install (00:00:03)	install-contrib (00:00:02)	install-testmodules (00:00:00)	check-pg_upgrade (00:02:11)
	test-decoding-check (00:00:25)	initdb-check (00:00:14)	pg_archivecleanup-check (00:00:01)	pg_basebackup-check (00:01:13)
	pg_config-check (00:00:01)	pg_controldata-check (00:00:02)	pg_ctl-check (00:00:11)	pg_dump-check (00:00:47)
	pg_resetwal-check (00:00:05)	pg_rewind-check (00:01:27)	pgbench-check (00:00:09)	scripts-check (00:00:48)
	recovery-check (00:02:01)	subscription-check (00:01:24)	authentication-check (00:00:06)	initdb-C (00:00:05)
	startdb-C-1 (00:00:25)	install-check-C (01:28:12)	stopdb-C-1 (00:00:02)	startdb-C-2 (00:00:39)
	isolation-check (00:43:18)	stopdb-C-2 (00:00:01)	startdb-C-3 (00:00:34)	pl-install-check-C (00:06:56)
	stopdb-C-3 (00:00:05)	startdb-C-4 (00:00:32)	contrib-install-check-C (00:44:24)	stopdb-C-4 (00:00:02)
	startdb-C-5 (00:00:27)	testmodules-install-check-C (00:06:16)	stopdb-C-5 (00:00:02)	ecpg-check (00:01:05)

PostgresPro

Pg_regress_fuzzing, SQLSmith, …

- Основные регрессионые тесты – около 270
- TAP (Test Anything Protocol) тесты – около 80
- + тесты расширений (contrib) – несколько десятков

Идея pg_regress_fuzzing – перемешивание тестов.

Результаты:

CVE-2019-10164

Bug # 15668 15684 15694 15828 15899 15910
15943 16037 16050 16134 16137 16139 16266
16276 16325 16329 16378 16466 16527 16577

Генераторы запросов

- SQL Smith - генератор случайных правильных синтаксических и семантических запросов
- Генератор запросов на основе грамматик PostgreSQL

Как ловить баги:

- Core dump, assertions, PANIC
- Анализ логов
- Анализ зависаний или использования CPU

PostgresPro

Использование санитайзеров libdislocator, ASAN/LSAN/UBSAN

LD_PRELOAD=/path_to/libdislocator.so make check-world

или

Добавляем флаги компилятора:

-fsanitize=address

-fsanitize=leak

-fsanitize=undefined

make installcheck, make check-world – ловим падения

Позволило найти ряд несколько ошибок, связанных с выходом за границы буфера

PostgresPro

Статический анализатор Svace, сервер историй сборки Svacer

На ветке REL_11_7 – 2474 предупреждения
На ветке ENT_11_7_1 – 2918 предупреждений Ⓟ надо разбираться с ложными срабатываниями, включать/выключать чекеры, нужны специфичные чекеры для управления контекстами памяти в PostgreSQL и т.д.
Нужны профили Svace под конкретные уровни доверия!
Иерархия снапшотов сборки в Svacer – разметка предупреждений и задания на устранение.

PostgresPro

Статический анализатор Svace, сервер историй сборки Svacer

PGPRO-4024	Предупреждения svace в коде pg_transfer	⌄	WAITING FOR MERGE	24 Aug 2020
PGPRO-4015	Предупреждения svace в коде multimaster	=	NEW	30 Jul 2020
PGPRO-4014	Предупреждения svace в коде shardman	=	NEW	30 Jul 2020
PGPRO-4011	Предупреждения svace в коде Connpool	⌄	NEW	31 Jul 2020
PGPRO-4018	Предупреждения svace в коде pg_dump, pg_upgrade	=	DONE	20 Aug 2020
PGPRO-4017	Предупреждения svace в коде PGPRO-1805, rsocket, priority	=	DONE	26 Aug 2020
PGPRO-4016	Предупреждения svace в коде backend_set_config	=	DONE	11 Aug 2020
PGPRO-4010	Предупреждения svace в коде vops	=	NEW	30 Jul 2020
PGPRO-4021	Предупреждения svace в коде CERT	⌃	CLOSED	06 Jul 2020
PGPRO-4020	Предупреждения svace в коде Pathman	=	NEW	30 Jul 2020
PGPRO-4019	Предупреждения svace в коде pgbench	⌃	CLOSED	03 Aug 2020
PGPRO-4022	Предупреждения Svace в коде	=	IN PROGRESS	03 Aug 2020
PGPRO-4023	Предупреждения svace в коде 64bit xids	=	NEW	30 Jul 2020
SRV-320	Настроить возможность работы с HASP ключами на ферме для тестирования	=	DONE	16 Jul 2020
SDL-49	Анализ Стандартной и Enterprise 11 сетрификационной ветки средствами Svace в рамках сертификации 2020	=	TO DO	26 Jun 2020
PGPRO-3992	Предупреждения svace в коде aqo	=	NEW	30 Jul 2020
PGPRO-3990	Предупреждения svace в коде pg_proaudit	⌃	DONE	21 Jul 2020

~ 100 задач на исправление в JIRA

Фаззинг – тестирование: AFL++, crusher

Проблемы и подходы:
Postgres – работа несколько параллельных процессов
Ⓟsingle mode;
 пропатчить, чтобы запускалось минимум процессов
Фаззинг:

- аргументов командной строки
- Фаззинг преобразования типа данных во внутреннее представление : tsvector, hstore, array,…
- Фаззинг модуля: json

Что дальше?

Постепенная разметка предупреждений svace и включение в CI – коммит после анализатора.
Поиск целей для фаззинга внутренних процессов: оптимизатор, планировщик, исполнитель.
Улучшение покрытия кода тестами.
Исследование инструментария для высоких уровней доверия.

КОМПАНИЯ POSTGRES PRO

Протестировать
СУБД Postgres Pro:

117036, Москва, ул. Дмитрия Ульянова, 7А

8 (495) 150-06-91

info@postgrespro.ru

www.postgrespro.ru