Разработки Postgres Pro для повышения безопасности и защиты данных

PGMEETUP

4 июня 2024 г., г. Москва

Гусаков Андрей,

Старший технический консультант

Реальны ли угрозы?

- СУБД – «лакомый кусочек» для злоумышленников и «последний рубеж обороны» для сотрудников отдела ИБ (см. tadviser.ru/index.php/Статья:Утечки_данных_в_России)

  «В 2023 году из финансовых организаций в России утекло 170,3 млн записей персональных данных клиентов, что в 3,2 раза превосходит показатель предыдущего года»

  «Общий объем утекших данных оценивается в 103,4 терабайта. Попавшие в общий доступ базы данных совокупно содержат 4,8 млн строк, в том числе 225 млн номеров телефонов и 145 млн адресов электронной почты»

- Риски несанкционированного доступа, потери целостности данных и самих данных – следствия ненадлежащего управления разрешениями, атак инъекционного типа, эксплуатации уязвимостей БД, наличия теневой инфраструктуры БД, уязвимости данных резервного копирования
- В апреле 2023 г. ФСТЭК ужесточила требования к разработчикам средств защиты СУБД
- Запрос на bdu.fstec.ru/search/index?q=postgres возвращает около 40 выявленных (и устраненных) уязвимостей на стыке ОС-СУБД -> применение рекомендаций и обновлений
- Применение «накладных» средств защиты не отменяет необходимость защиты на уровне ядра СУБД
- Для построения эффективной системы защиты необходимо сотрудничество между отделами ИБ, сопровождения информационных систем, вендорами и интеграторами

Методы и стратегии защиты баз данных

PostgresPro

- оценка уровня опасности уязвимостей базы данных, что включает в себя обнаружение скомпрометированных конечных точек и классификация конфиденциальных данных;
- постоянное управление правами доступа пользователей и предотвращение наличия чрезмерных привилегий и неактивных пользователей;
- обучение сотрудников методам снижения рисков, что включает в себя получение знаний о распространенных киберугрозах, таких как фишинговые атаки и эксплуатация электронной почты;
- отслеживание всей активности, связанной с получением доступа к базе данных в режиме реального времени, для обнаружения утечек данных, несанкционированных SQL, а также атак на протоколы и системы;
- автоматизация проведения аудита с помощью специально подобранной платформы;
- блокировка вредоносных веб-запросов;
- архивация внешних данных, шифрование информации в базе данных и маскировка ее полей, чтобы скрыть конфиденциальную информацию.

cisoclub.ru/ugrozy-bezopasnosti-baz-dannyh-i-sredstva-zashhity-bd/

Рекомендации Center for Internet Security, Inc. (CIS) и Security Technical Implementation Guides (STIGs)

PostgresPro

- Разработка показателей и рекомендаций для совершенствования программ обеспечения безопасности и соответствия требованиям
- Создание базовых уровней конфигурации безопасности систем
- Сопоставление текущего состояния объекта с требуемым

Scoring Information

A scoring status indicates whether compliance with the given recommendation impacts the assessed target's benchmark score. The following scoring statuses are used in this benchmark:

Scored

Failure to comply with "Scored" recommendations will decrease the final benchmark score. Compliance with "Scored" recommendations will increase the final benchmark score.

Not Scored

Failure to comply with "Not Scored" recommendations will not decrease the final benchmark score. Compliance with "Not Scored" recommendations will not increase the final benchmark score.

Имеются готовые решения для выявления атак на СУБД, но лучше сфокусироваться на превентивных мерах...

PostgresPro

Пакет экспертизы для пользователей MaxPatrol SIEM от Positive Technologies помогает «обнаружить такие подозрительные действия, как:

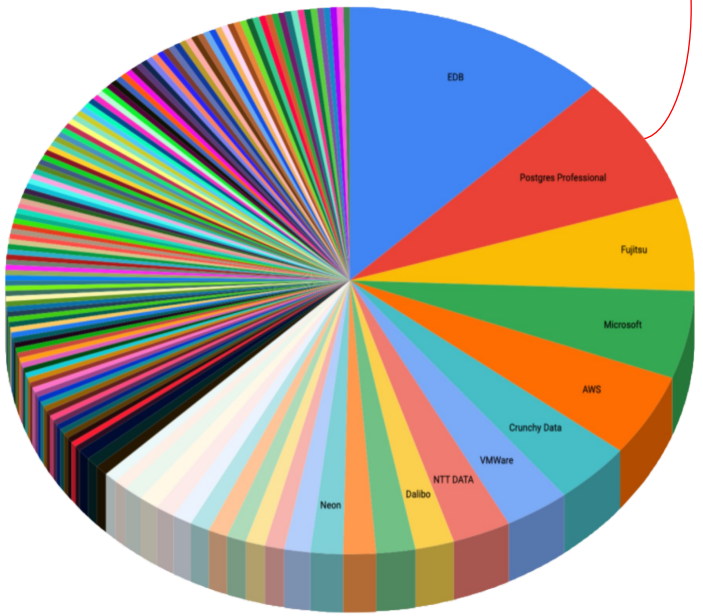
- отправка команд для определения версии БД (свидетельствует о начале атаки на СУБД),
- чтение таблиц, содержащих хеш-суммы паролей,
- отключение аудита,
- изменение уровня важности сообщений аудита для сокрытия действий,
- изменение метода аутентификации для повышения вероятности компрометации пользователя или роли в системе,
- отключение политики защиты строк для таблицы,
- отключение шифрования канала передачи данных (SSL),
- перезагрузка конфигурации сервера,
- запуск встроенных приложений операционной системы с возможностью выполнения произвольных команд.»

ptsecurity.com/ru-ru/about/news/maxpatrol-siem-vyyavlyaet-ataki-na-subd-postgresql

...с помощью наших решений в области ИБ

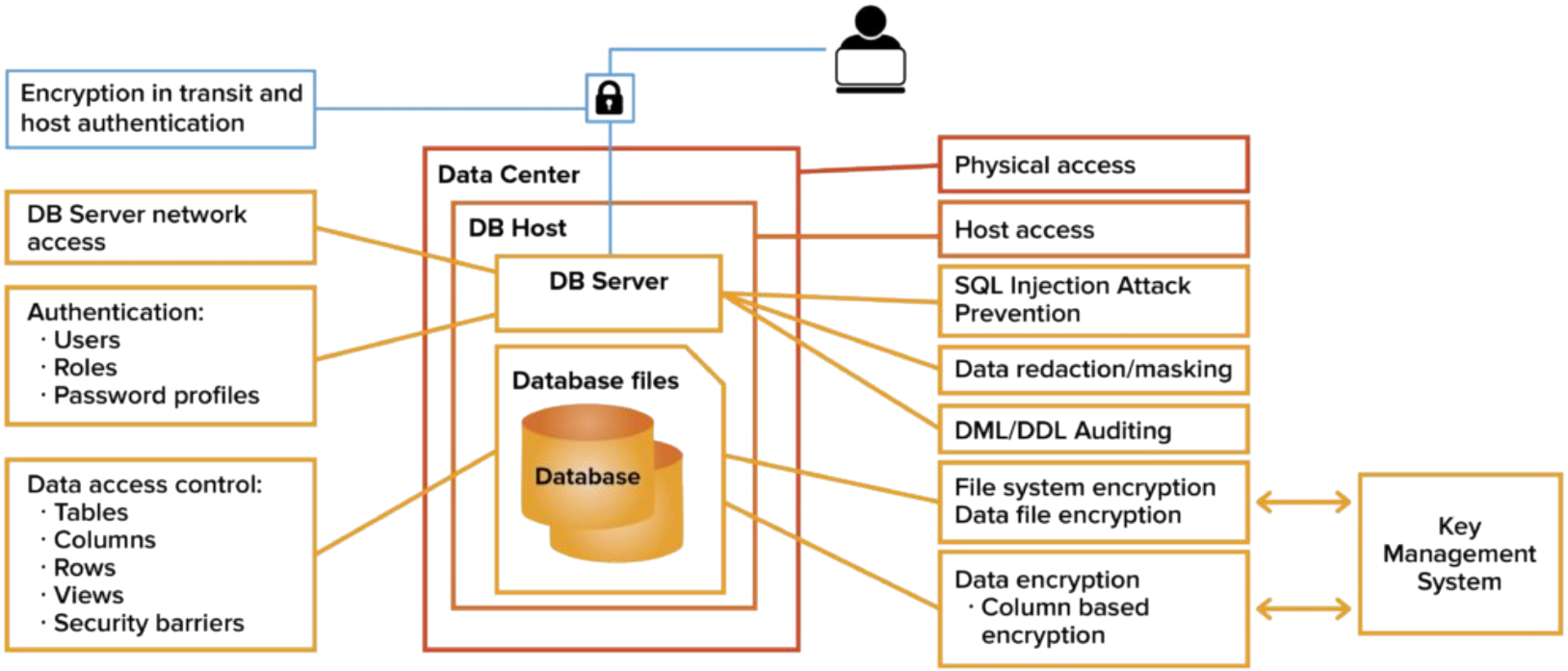
- ООО "Постгрес Профессиональный" – разработчик и правообладатель ПО класса СУБД на основе PostgreSQL в соответствии с реестровой записью №104 от 18.03.2016 в reestr.digital.gov.ru
- ООО «Постгрес Профессиональный» является обладателем лицензии ФСТЭК на деятельность по разработке и производству средств защиты конфиденциальной информации (см. reestr.fstec.ru/reg2)
- Сертификация (подтверждение характеристик) проводится с 2016 года; сейчас – по «Требованиям по безопасности информации к системам управления базами данных», утвержденным приказом ФСТЭК России от 14 апреля 2023 г. N 64, самого высокого для защиты конфиденциальной информации четвертого класса защиты и уровня доверия (продукту и его производству)
- Postgres Pro – контрибьютер #2 в код PostgreSQL в мире (после EnterpriseDB)
- Совместимость ПО – это crowdtesting + дальнейшее развитие и поддержка + заплатки для устранения уязвимостей (см. postgresql.org/support/security)

Вклад PostgresPro в разработку PostgreSQL



В PostgreSQL имеется множество решений ИБ

- Управление привилегиями
  - Ролевая модель
  - Row Level Security
  - Маскирование данных
- Безопасность подключений
- Идентификация и аутентификация пользователей
- Встроенный аудит
- Встроенная криптозащита



itopstimes.com/itsec/creating-a-multi-layered-security-architecture-for-your-databases/

Почему этого недостаточно для защиты информации?

PostgresPro

- Проблема суперпользователя – бесконтрольность, невозможность отследить опасные действия, риски использования superuser для работы сервисов
- Проблема доступа к данным привилегированных пользователей
- Недостаточная гибкость аудита и повышенная нагрузка от него на СУБД
- Риски безопасности при несанкционированном изменении файлов СУБД
- Неудобная криптография – непрозрачно для приложений, утилит, имеются уязвимости

- Растущие требования регуляторов

Контроль привилегированных пользователей

16.1 PostgresPro

- Задача – снизить вероятность того, что суперпользователь в одиночку прочитает и/или изменит «неположенные» ему данные для предотвращения утечек

- Устанавливаются правила логирования любых действий с пользователями, изменений конфигурации СУБД, изменений хранимых процедур, любых DDL

«Замена» суперпользователя администратором СУБД

PostgresPro

- При создании Администратора СУБД ему выдаются права на следующие встроенные роли
  - pg_read_all_settings
  - pg_read_all_stats
  - pg_stat_scan_tables
  - pg_monitor

  } чтение различных полезных параметров конфигурации, статистики и другой системной информации

  - pg_signal_backend
  - pg_checkpoint

  } выполнение экстренных действий на уровне инстанса

  - *pg_create_tablespace (новая предопределенная роль)*
  - *pg_manage_profiles (новая предопределенная роль)*

  } создание табличных пространств и управление профилями

- При создании Администратора СУБД ему выдаются права на следующие системные функции
  - pg_reload_conf()
  - pg_rotate_logfile()

  } управление конфигурацией и логированием

  - pg_create_restore_point()
  - pg_backup_start()
  - pg_backup_stop()

  } управление бэкапированием и восстановлением

  - pg_switch_wal()
  - pg_promote()
  - pg_wal_replay_pause()
  - pg_wal_replay_resume()

  } управление журналом предзаписи и репликацией

habr.com/ru/companies/postgrespro/articles/788268/

Разграничение прав администраторов – реализация

- Создана роль PGPRO_DBMS_ADMIN
  - CREATEDB, CREATEROLE, REPLICATION, INHERIT
  - Создаёт новые БД
  - Создаёт пользователей – Администраторов БД
  - Создаёт tablespaces
  - Меняет настройки СУБД
  - Создаёт пользователей для репликации
  - Управляет репликацией
- Создана роль PGPRO_DB_ADMIN
  - CREATEROLE, INHERIT
  - Создаёт таблицы и хранимые функции в БД
  - Создаёт пользователей БД
  - pg_dump / pg_restore
- Доработка pg_integrity_check
- Доработка pg_proaudit
- Ограничены права ролей с атрибутом CREATEROLE и их возможности изменять другие роли (back port из 16 версии в 13, 14 и 15)
- Обычным пользователям запрещены любые изменения кода информационных систем. Они не могут создавать или изменять код хранимых процедур, функций, пакетов, триггеров. Это позволено только администратору БД

	13*	14*	15*	16
Postgres Pro Standard		14.10.1+	15.5.1+	16.1.1+
Postgres Pro Enterprise	13.13.1+	14.10.1+	15.5.1+	16.1.1+

* - выполнен бэкпорт кода из 16-й версии

Как ставить расширения без superuser?

- Используем то, что trusted-расширение ставится от имени postgres
- SQL-файлы расширения выполняются от имени postgres без login к базе, внутренними механизмами
- Безопасность достигается необходимостью вовлечение нескольких человек в установку расширения: Администратора СУБД и Администратора безопасности (инфраструктуры)

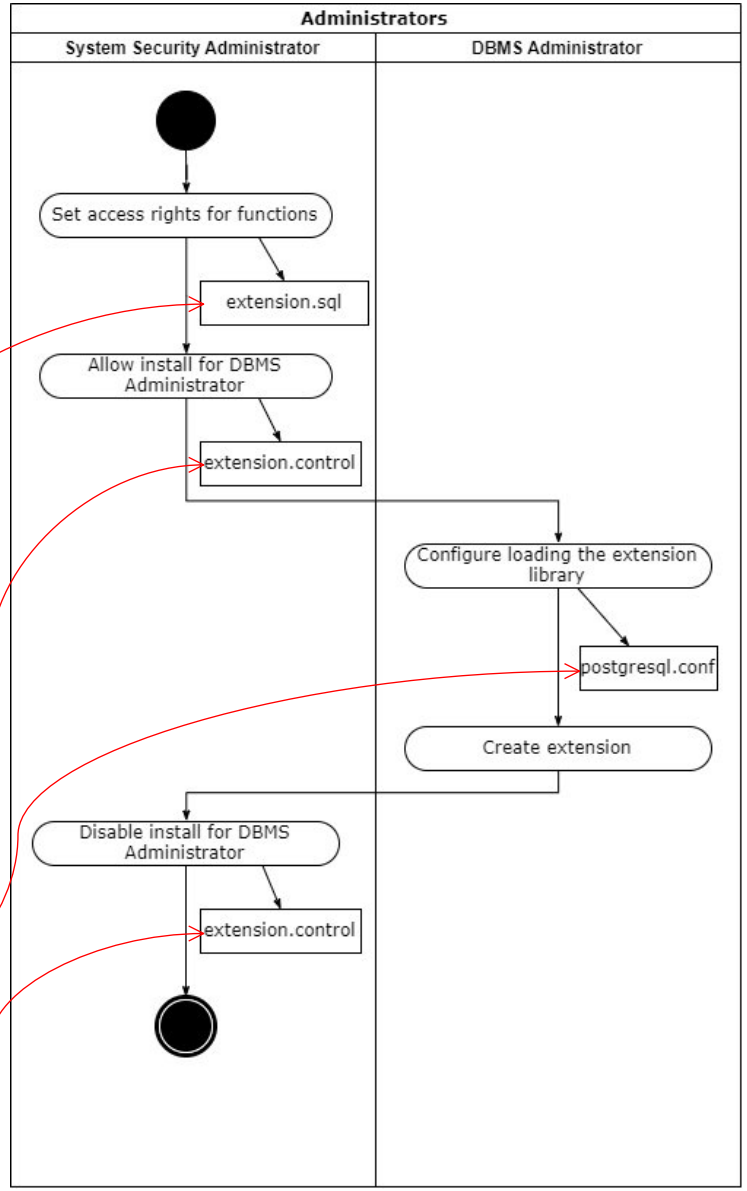
Пример выдачи разрешения Администратору СУБД для установки расширения pg_proaudit

```
-- Create new versions of objects
CREATE FUNCTION pg_proaudit_show()
RETURNS TABLE(
  db_name text,
  event_type text,
  object_type text,
  object_oid oid,
  role_name text)
AS 'MODULE_PATHNAME', 'pg_proaudit_show_conf'
LANGUAGE C VOLATILE;
REVOKE ALL ON FUNCTION pg_proaudit_show() FROM public;
GRANT ALL ON FUNCTION pg_proaudit_show() TO PGPRO_DBMS_ADMIN;
```

trusted = true

add file to shared_preload_libraries + systemctl restart postgresql

trusted = false

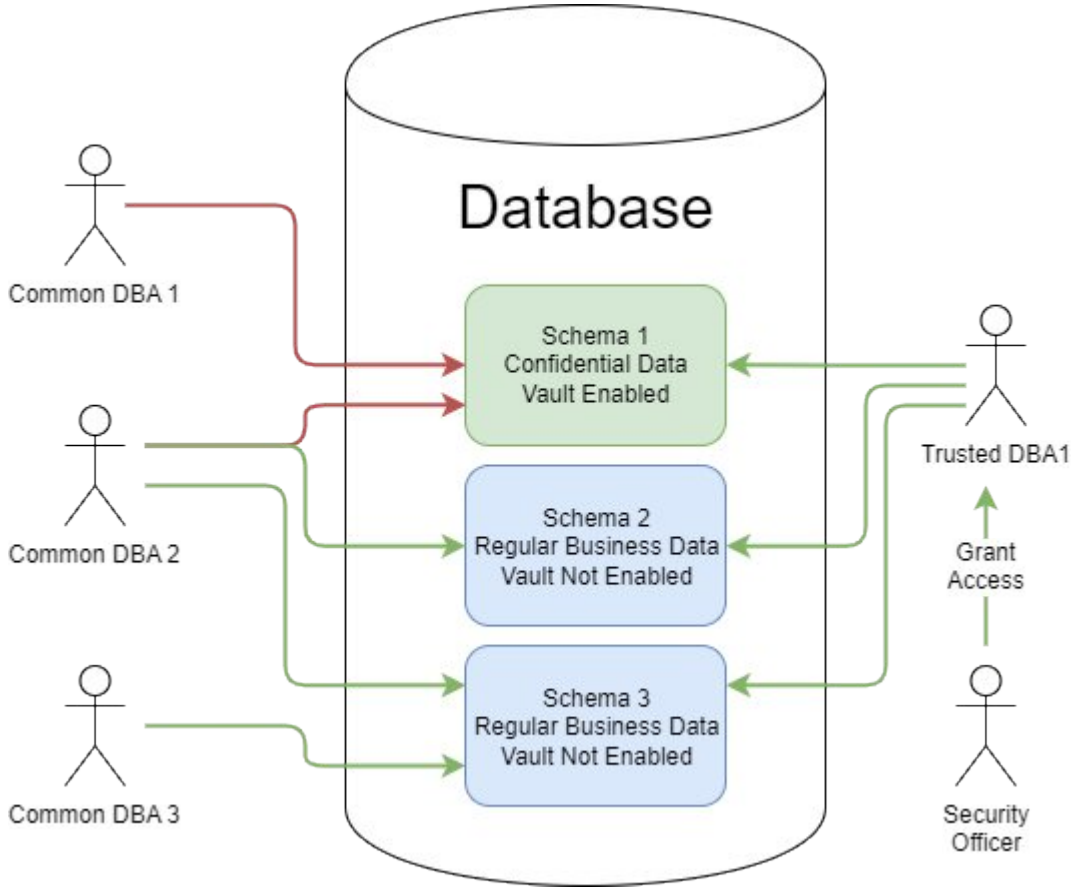


Как ограничить доступ администратора к данным?

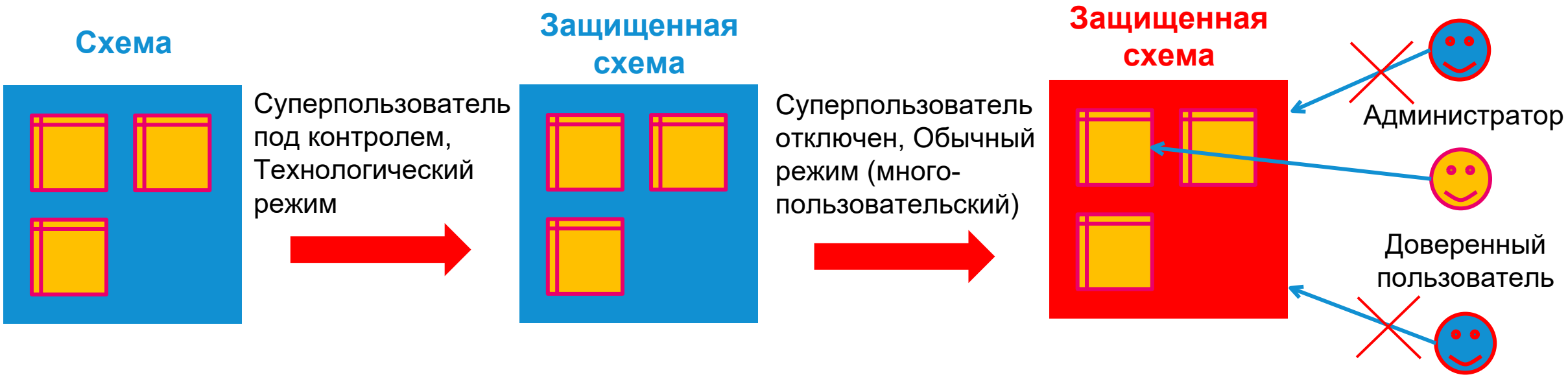
16.1 PostgresPro

- Задача – предоставить доступ только доверенным бизнес-пользователям!

- В больших организациях слишком много администраторов имеют доступ к серверу СУБД, так как им надо производить её регулярное обслуживание, резервное копирование и т.д.
- Надо отнять у них права доступа к чувствительным данным и коммерческой тайне, передав им «доверенным» администраторам
- В больших организациях доступ к БД раздаёт отдел безопасности, а не Админы СУБД. Нужно, чтобы доступ к чувствительным данным и коммерческой тайне давали тоже только они



Алгоритм организации защиты данных в схеме



Владелец: **DB_Admin**

Менеджер прав доступа: отсутствует

Владелец: **DV_OWNER**

Менеджер прав доступа: **SEC_OFFICER**

Владелец: **DV_OWNER**

Менеджер прав **SEC_OFFICER**

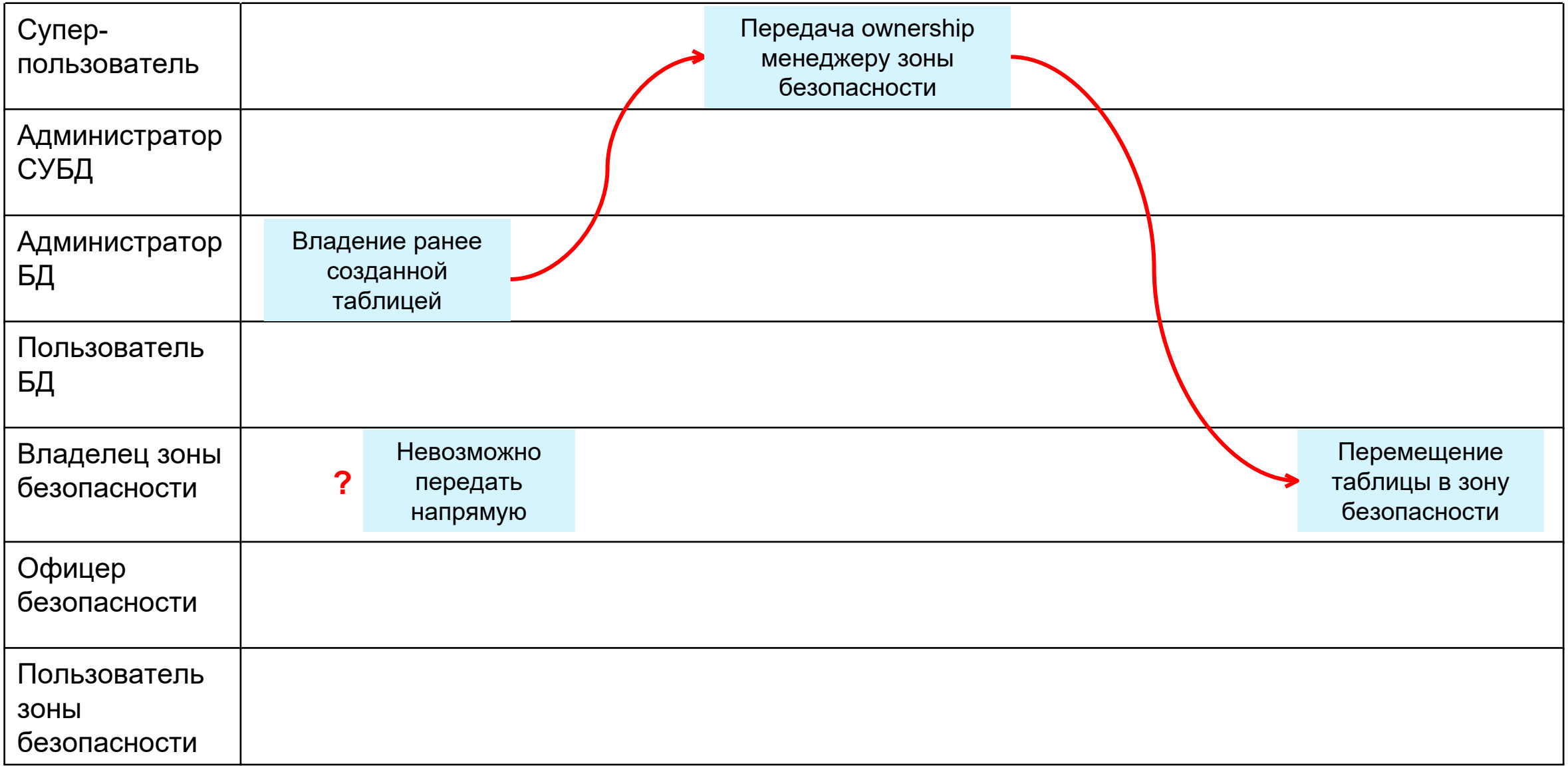
Назначение с привлечением суперпользователя

ADMIN OPTION – проблема при предоставлении доступа (создании) пользователю защищенной зоны

Решение проблемы ADMIN OPTION при предоставлении доступа (создании) пользователю защищенной зоны

Решение проблемы отсутствия общей роли при переносе существующей таблицы в защищённую зону

PostgresPro



Администратор без доступа к данным – реализация

PostgresPro

- Вводится понятие защищенной зоны на уровне схемы, которую создает суперпользователь
- Вводятся специальные роли для использования в защищенной схеме – владельца и офицера безопасности, которых тоже создает суперпользователь
- Администратор БД (не СУБД!) создает бизнес-пользователей БД, суперпользователь разрешает подключения к БД с защищенной зоной владельцу и офицеру безопасности и «очищает» УЗ бизнес-пользователей БД от ADMIN OPTION
- Владелец защищенной зоной создает в ней таблицы, а офицер безопасности разрешает доступ и определяет привилегии пользователей защищенной зоны в явном виде. Администраторы СУБД и БД по умолчанию не имеют доступа к данным защищенной зоны и не могут определять привилегии пользователей защищенной зоны
- Используется стандартный механизм ACL
- Изменения:
  - Код выдачи прав на доступ к объектам – pg_class, pg_proc, pg_type, pg_collation
  - В структуре каталога pg_namespace - новое поле nspsecofficer, аналогичное nspowner
  - Роль SECURITY OFFICER внесена в Schema в явном виде
  - В синтаксисе ALTER SCHEMA появилось «ALTER SCHEMA … SECURITY OFFICER TO …»
  - В pg_dump – возможность выгрузки только прав доступа для pg_restore под Security Officer

Изменения в программе для создания резервных копий

PostgresPro

- В программе для создания резервных копий базы данных **pg_dump** предусмотрена раздельная выгрузка (а) структуры и данных и (б) прав доступа защищенной зоны

	Порядок выгрузки данных	Порядок загрузки данных
Администратор БД	1. выгрузить все данные, кроме защищённой схемы	1. загрузить все данные, кроме защищённой схемы
Владелец зоны безопасности	2. выгрузить данные защищённой схемы 3. выгрузить права доступа	2. загрузить данные защищённой схемы
Офицер безопасности		3. загрузить права доступа к защищённой схеме

Регистрация событий безопасности

16.2 PostgresPro

- Задача – расширить охват аудита и снизить нагрузку на СУБД:
  pgAudit -> pg_proaudit (v1.0 быстрый, но сложные настройки) -> **pg_proaudit (v2.0)**
- Оптимизированный механизм поиска – правило для аудирования подбирается фильтрацией правил по комбинации
  - Имя базы данных
  - event_type
  - object_type
  - object_name
  - role_name
- Легко настраивать правила
- Логирует, какой пользователь выполнил действие
- Есть логирование по группе пользователей, а не только по конкретному пользователю
- Может логировать любую команду SQL
- Выше быстродействие благодаря параллельной обработке

Типовой «рекомендованный» набор правил

- Удовлетворяет требованиям сертификации ФСТЭК и известным нам типовым требованиям ИБ
- Имитирует типовую настройку с учётом наличия Secured Schema

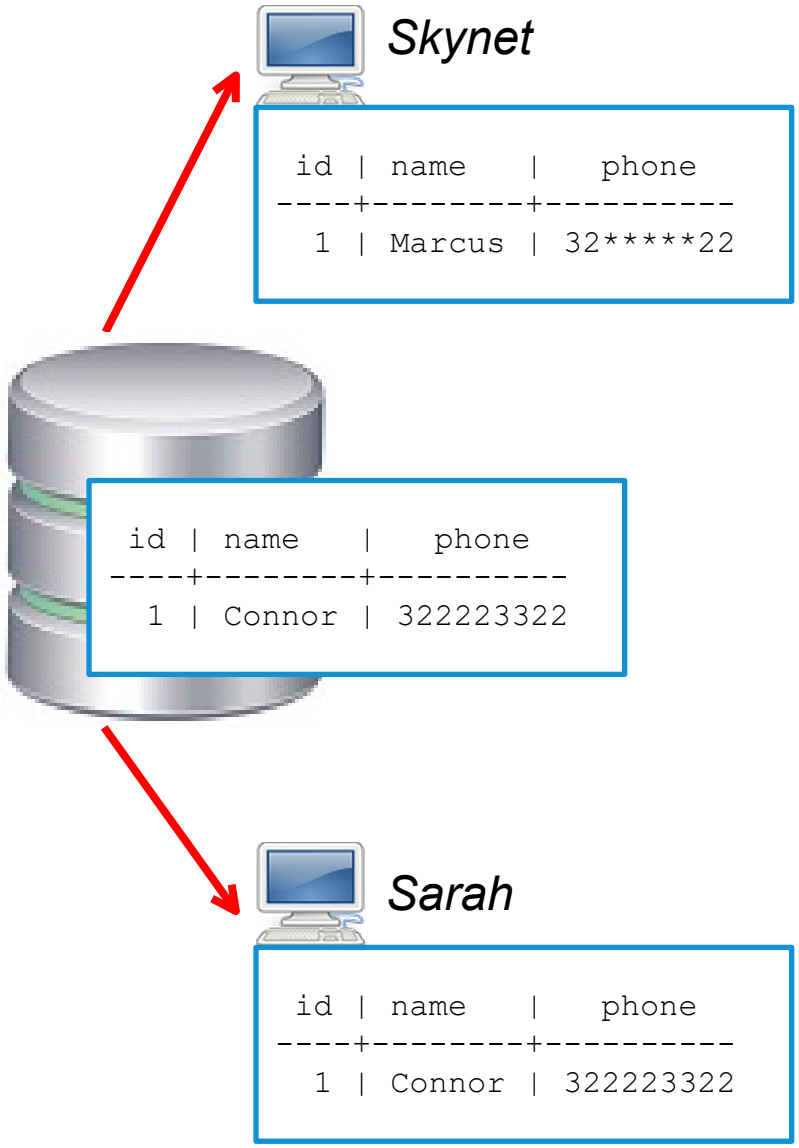
DB	event_type	object_type	object_name	role_name	comment
	AUTHORIZE				все новые авторизации (соединения)
	DISCONNECT				все окончания соединений
	ALL	ROLE			все действия над пользователями и ролями - create/alter/drop user/role/group
	ALTER SYSTEM				все изменения системной конфигурации
	ALL_DDL				все создания и модификации баз данных, таблиц, представлений, хранимых функция и процедур, ...
	GRANT				все разрешения доступов
HR_DB	ALL_DML	SCHEMA	HR_VAULT		весь доступ к данным внутри безопасной схемы
HR_DB	ALL_PROC	SCHEMA	HR_VAULT		все вызовы хранимок из безопасной схемы
	ALL			SUPPORT	все действия инженеров поддержки
	ALL			PGPRO_DBMS_ADMIN	все действия Администраторов СУБД

Маскирование данных

15.1 PostgresPro

- Во многих случаях раскрытие данных СУБД нежелательно, и для защиты коммерческих или персональные сведений их надо заменить либо фейковой, либо частичной информацией
- Оригинал при этом остается нетронутым
- pgpro_anonymizer – расширение для маскирования или замены конфиденциальных данных внутри экземпляра PostgresPro
- В проекте используется декларативный подход к анонимизации. Вы можете объявлять правила маскирования, используя язык описания данных (DDL), и задавать свою стратегию анонимизации внутри определения таблицы
- Используются следующие основные стратегии:
  - Динамическое маскирование изменяет представление реальных данных, не модифицируя их. Некоторые пользователи могут читать только замаскированные данные, а другие могут получить доступ к исходной версии
  - Статическое маскирование полностью заменяет конфиденциальную информацию несвязанными данными. После такой обработки исходные данные не могут быть восстановлены

Пример динамического маскирования данных



Правила маскирования задаются метками безопасности:

```
SELECT anon.start_dynamic_masking();

SECURITY LABEL FOR anon ON COLUMN people.name
   IS 'MASKED WITH FUNCTION anon.fake_first_name()';
SECURITY LABEL FOR anon ON COLUMN people.phone
   IS 'MASKED WITH FUNCTION anon.partial(phone,2,$$******$$,2)';

SECURITY LABEL FOR anon ON ROLE Skynet, Mark IS 'MASKED';
```

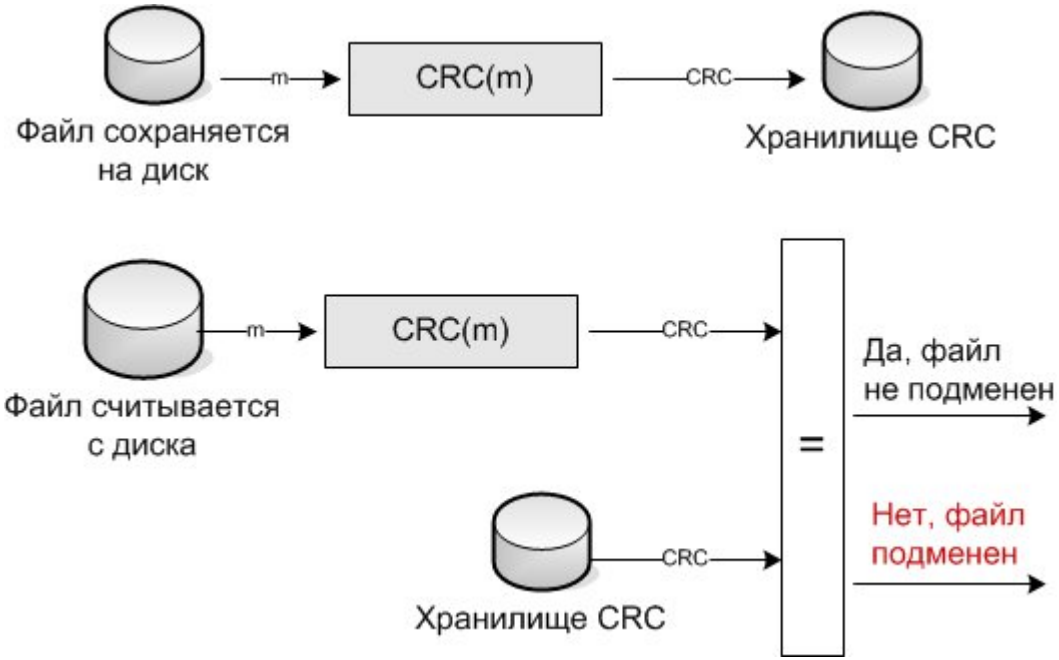
В результате доверенный пользователь с ролью Sarah получит немодифицированные данные, а недоверенный пользователь с ролью Skynet – строку с фейковым именем и частично сокрытым номером телефона

Варианты изменения данных при маскировании

- Удаление просто удаляет данные.
- Статическая замена последовательно заменяет данные одним и тем же значением. Например: замена всех значений столбца типа text на значение «КОНФИДЕНЦИАЛЬНО».
- Отклонение «сдвигает» даты и числовые значения. Например, после применения отклонения +/- 10% к столбцу зарплаты набор данных не потеряет смысл.
- Обобщение снижает точность данных, заменяя их диапазоном значений. Вместо «Бобу 28 лет» можно сказать «Бобу от 20 до 30 лет». Этот метод полезен для аналитики, так как данные остаются верными.
- Перестановка перемешивает значения в рамках столбца. Исходные данные могут быть восстановлены, если алгоритм перестановки будет расшифрован.
- Рандомизация заменяет конфиденциальные данные случайными, но правдоподобными значениями.
- Частичное скрытие аналогично статической подстановке, но оставляет часть данных нетронутыми
- Пользовательские правила предназначены для изменения данных в соответствии с особыми требованиями (например, заменить одновременно почтовый индекс и название города случайными значениями, чтобы они оставались согласованными).
- Псевдонимизация защищает персональные данные, скрывая их с помощью дополнительной информации. Шифрование и хеширование — два примера методов псевдонимизации. Псевдонимизированные данные, тем не менее, по-прежнему остаются связаны с исходными данными.

Контроль целостности

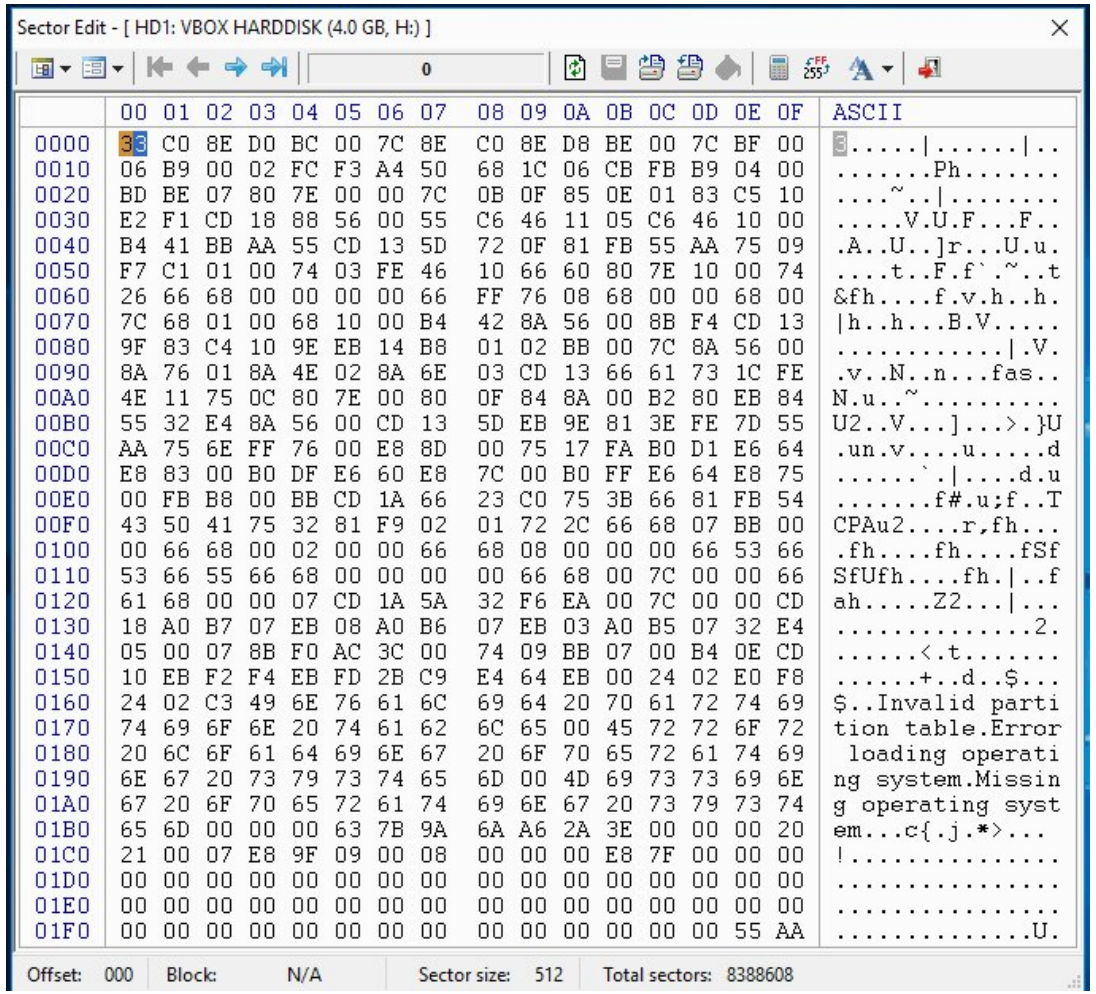
- Требования предъявляются не к данным СУБД, а к конфигурации СУБД!
- Реализуется утилитой **pg_integrity_check**
- Расчет контрольных сумм учитывает содержимое файла, его атрибуты, время изменения
- Имеются отдельные настройки для неизменяемых файлов (бинарники) и файлов, которые могут быть изменены администратором баз данных (конфигурационные файлы, системные таблицы).
  Для первых контрольные суммы поставляются готовыми, для вторых необходимо провести расчеты при вводе в промышленную эксплуатацию
- При старте СУБД контроль целостности бинарников выполняется автоматически; при несовпадении контрольных сумм требуется вмешательство root для принятия изменений или отката



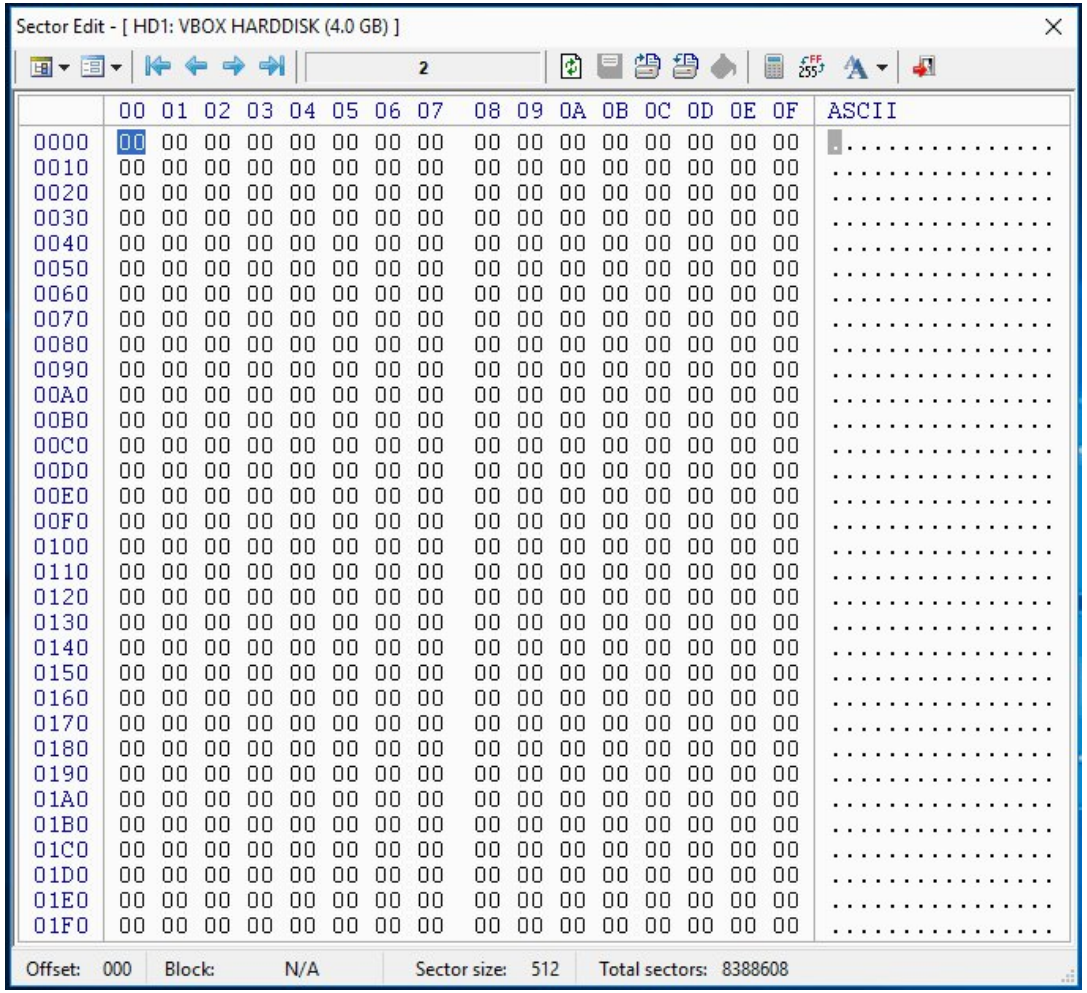
Очистка памяти (имеется в виду – на дисках)

14.1 PostgresPro

Файл помечен, как удаленный, но информацию можно восстановить



Проведена очистка удаленного файла с заполнением адр. пространства нулями



Очистка памяти – уточнение задачи СУБД

PostgresPro

- Удаление баз данных и журналов, используемых СУБД, путём многократной перезаписи уничтожаемых файлов – не задача Postgres; в Linux реализуется например командами **wipe** или **shred**
- Postgres может заполнить пустышками (нулевыми байтами) модифицированные участки файлов при выполнении команд DELETE и VACUUM (см. postgrespro.ru/docs/enterprise/16/memory-purge-cert)
- Внимание! Очень ресурсоёмкая операция...
- Немного разные решения используются для
  - Удаления файлов из внешней памяти
  - Очистки страниц (Multiversion Concurrency Control –> VACUUM)
  - Очистки блоков памяти в ОЗУ (MemoryContexts)
  - Очистки файлов WAL (удаление или перезапись сегментов)

Доступная криптозащита данных

- Это не шифрование по определению ФСБ! (см. п.4 в clsz.fsb.ru/clsz/license.htm)
- Шифрование паролей
  - Предпочтительный метод – SCRAM
- Шифрование избранных столбцов
  - Модуль **pgcrypto** позволяет хранить в зашифрованном виде избранные поля. Чтобы прочитать эти поля, клиент передаёт дешифрующий ключ, сервер расшифровывает данные и выдаёт их клиенту. Уязвимо в процессе расшифровывания и передачи данных из-за потенциального перехвата системным администратором
- Шифрование раздела данных
  - Шифрование хранилища данных можно реализовать на уровне файловой системы или на уровне блоков. Уязвимо при хранении ключа на компьютере, который монтирует ФС
- Шифрование данных при передаче по сети
- Проверка подлинности сервера SSL
- Шифрование на стороне клиента
  - Вся логика д.б. на стороне клиента (приложения) – потенциальный удар по производительности!

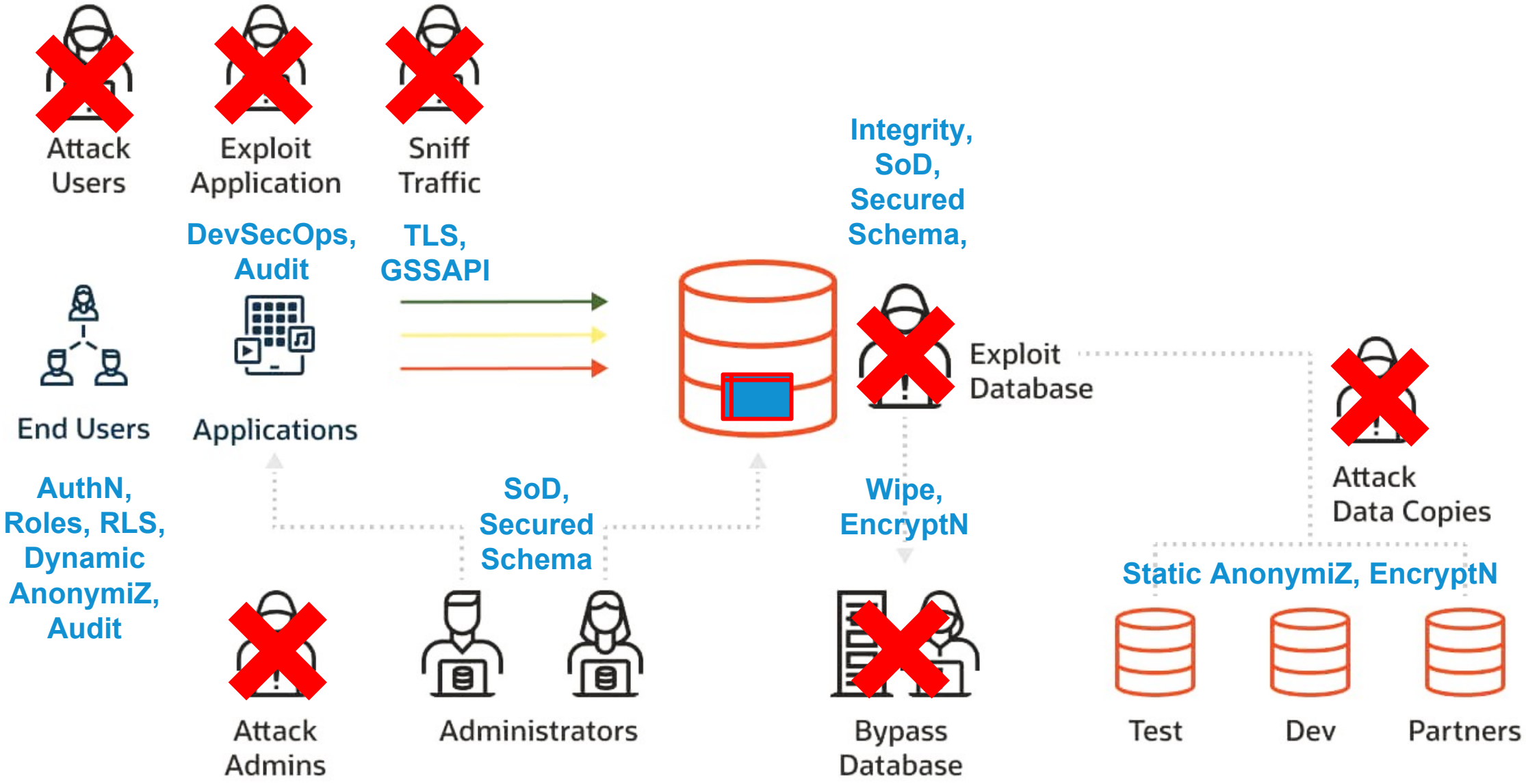
postgrespro.ru/docs/enterprise/16/encryption-options

В планах – удобная криптозащита данных

- Наш вриант реализации Transparent Data Encryption для защиты данных в состоянии покоя
- В разработке:
  - Механизмы AES-шифрования разных типов данных – страниц данных, файлов данных отношений, индексов, материализованных представлений, временных таблиц, файлов WAL
  - Механизмы работы с к ключами – иерархия, места хранения, ротация, вызов неактивных ключей
  - Авторизация нескольких утилит на использование криптографии в фоне
  - Инструкции по изменениям в работе с кластером СУБД, репликами и резервными копиями
  - Порядок развертывания решения и смены алгоритма шифрования
- Цель - разработка решения с модульной архитектурой, обеспечивающей возможность интеграция с различными СКЗИ для использования альтернативных алгоритмов шифрования

Нейтрализация угроз средствами защиты СУБД

PostgresPro



Сертифицированные редакции СУБД

PostgresPro

Standard

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

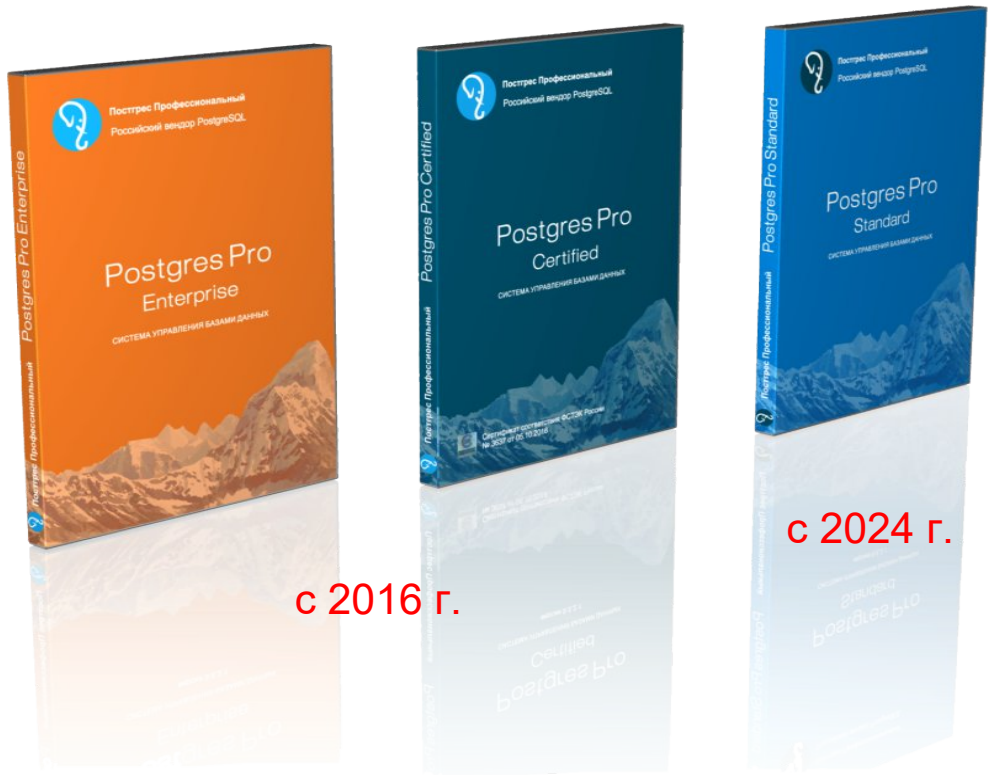
Enterprise

Наиболее полнофункциональная СУБД с высокой производительностью и масштабируемостью

Shardman

Распределенная СУБД, предоставляющая строгие гарантии целостности данных

соответствуют «Требованиям по безопасности информации к СУБД» ФСТЭК России самого высокого для защиты конфиденциальной информации четвертого класса защиты и уровня доверия



с 2016 г.

с 2024 г.

с ежеквартальным обновлением в рамках инспекционного контроля

Где можно использовать наши сертифицированные версии

PostgresPro

- Системы управления базами данных, соответствующие 4 классу защиты, применяются
  - в значимых объектах критической информационной инфраструктуры 1 категории значимости,
  - в государственных информационных системах 1 класса защищенности,
  - в автоматизированных системах управления производственными и технологическими процессами 1 класса защищенности,
  - в информационных системах персональных данных при необходимости обеспечения 1 уровня защищенности персональных данных,
  - в информационных системах общего пользования II класса

fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/trebovaniya-po-bezopasnosti-informatsii-utverzhdeny-prikazom-fstek-rossii-ot-14-aprelya-2023-g-n-64

Все это – безопасность инвестиций!

PostgresPro

Безопасность разработки

Востребованный функционал, Стабилизация, Вклад в комьюнити-версии

Безопасность эксплуатации

Быстрая доступность обновлений, Скорость и качество поддержки

Юридическая безопасность

Сертифицированные версии, Собственная лаборатория, Инспекционный контроль

Средства безопасности СУБД

Ролевая модель, Маскирование данных, Ограничение прав суперпользователя, Продвинутый аудит, Защита данных

PostgresPro

Спасибо за внимание!