

ТИНЬКОФФ

То, чем стоило заняться вчера

**защита данных, когда их стало слишком
много**





Глеб Марченко | Data Security Lead



g.v.marchenko@tinkoff.ru



[@KeepItAwesome](https://t.me/KeepItAwesome)

Почему данные вообще нужно защищать?



Потому что законы

- Штрафы за утечки ПДн
- Холд на оперриски от ЦБ
- А может и прокуратура



Потому что репутация

- Клиенты - это вообще-то важно
- Негативный пиар негативен



Потому что деньги

- Могут срываться проекты
- Инсайдеры сосут нашу кровь
- Устранение последствий – дорогое удовольствие

Продуктовые команды

- Безопасность мешает удобно работать и увеличивает T2M
- Приходится разрабатывать неинтересный функционал
- Планы уже составлены, бэклог бэклог капасити
- Трудно измерима в понятных метриках

Информационные безопасники

- Данные – такой же элемент ландшафта
- Некому заниматься доработками и анализом
- Нужен нетривиальный набор скиллов
- Готовых решений нет

Владельцы и топы

- Утечка данных – инцидент масштаба всей компании
- Страдает общий бюджет
- Репутация, штрафы, vip-клиенты, проекты, финансы, подставь свое
- Не могут вникать в каждый конкретный процесс

Топы

Осознают необходимость
и формируют запрос

Бизнес

Принимают
необходимость изменений

ИБ

Формируют подходы и
правила игры

IT

Реализуют в железе и коде

**Для защиты
данных нужна
инициатива
сверху и
ответственность
снизу**

От чего нужно защищать данные?

Внутри

- ▶ Пробивы
- ▶ Доступ к данным пачкой
- ▶ Халатность
- ▶ Куча времени внутри
- ▶ Data Poisoning
- ▶ SuperUser Misbehave

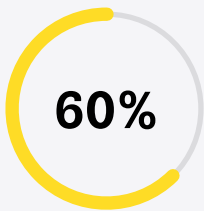


Снаружи

- ▶ Удаление
- ▶ Шифрование
- ▶ Порча

AND

- ▶ Все то же, что и внутри

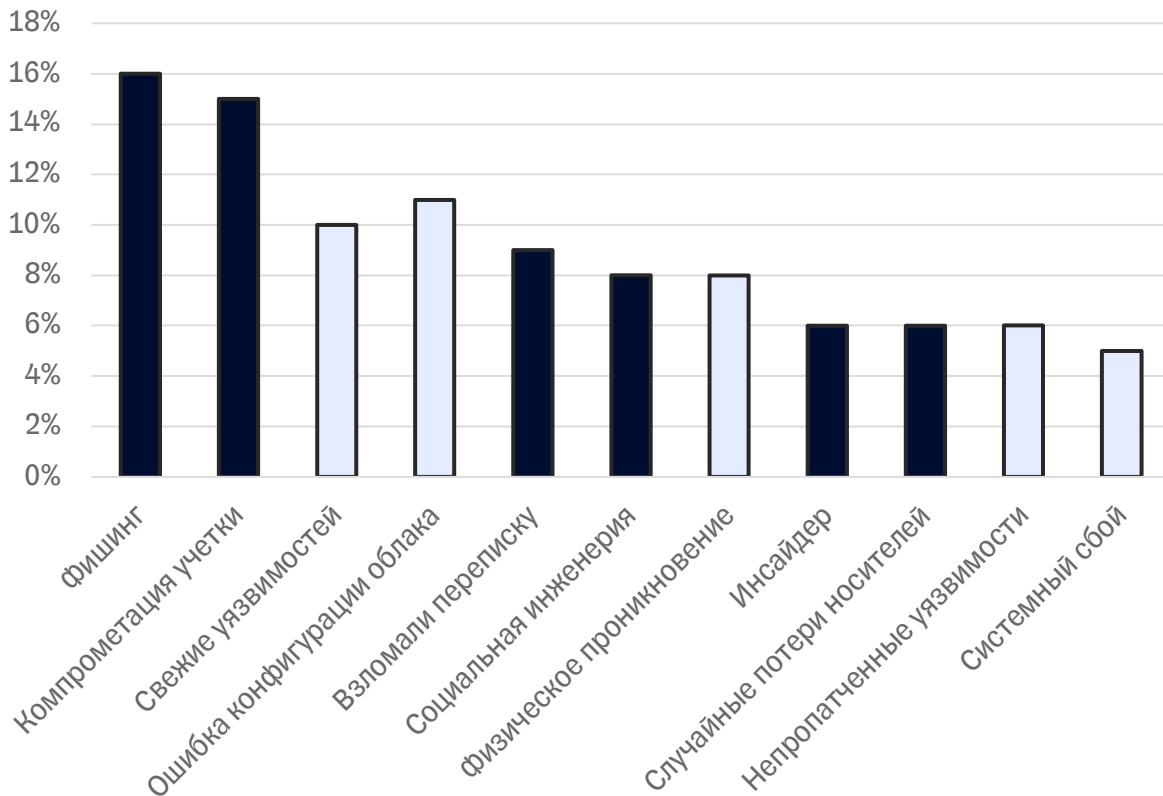


утечек связаны с
пользователями



дней требуется в
среднем на обнаружение
таких утечек

Что стало причиной утечки?



Структура

Понимаем, где и какие
данные используются

Процессы

Знаем, кто и для чего
использует разные данные

Доступ

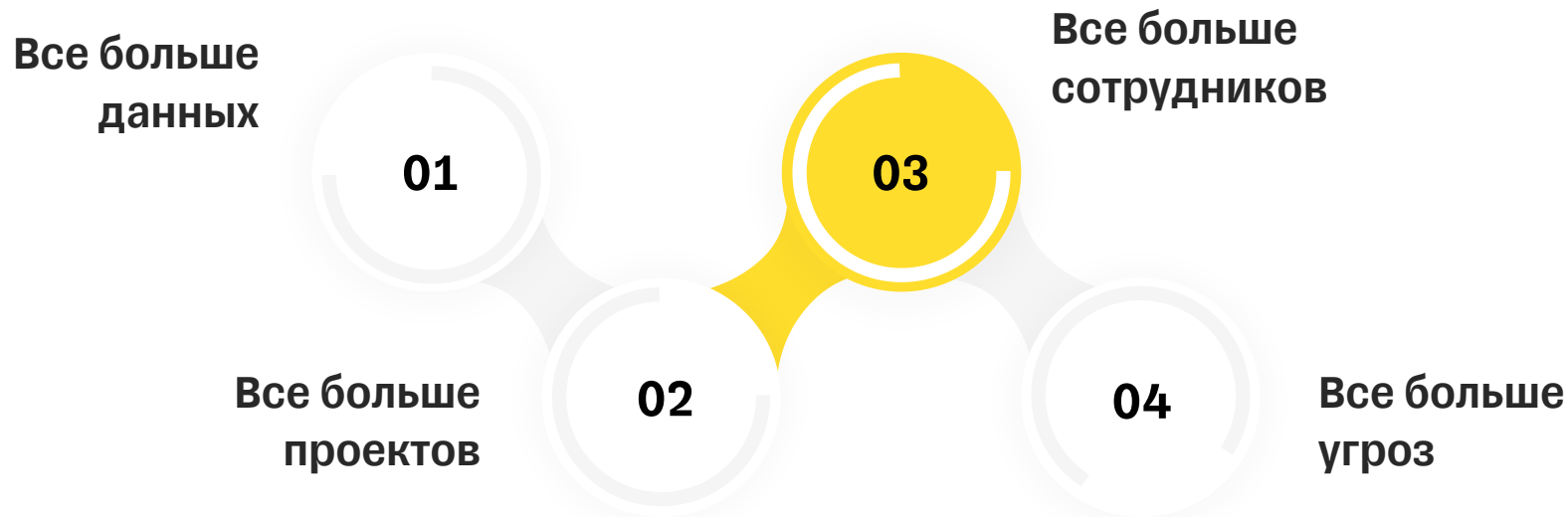
Даем ровно столько,
сколько нужно

Контроль

Следим, чтобы все было
правильно

**Как должно
быть в идеале**

А что делать, если бизнес сильно обогнал безопасность?



Two-step ladder to success

- 01 срочно найти все
чувствительные данные
- 02 и немедленно
ограничить к ним доступ

Для этого мы придумали Hound



(это не настоящий логотип)

Hound – это инструмент для поиска чувствительных данных



Поддерживает

GreenPlum/PostgreSQL

Oracle

Cassandra

ClickHouse

Hadoop



Как ищем

ML-модели

Правила и регулярки

Метаданные

Lineage

Hound – это инструмент для поиска чувствительных данных



Как работает

1. Получает задание на сканирование БД
2. Подключается к БД
3. Собирает список таблиц для сканирования
4. В цикле по таблицам запускает проверку
 - а. Получает семпл данных из таблицы - важно для больших БД с терабайтами данных
 - б. Проверяет семпл на наличие чувствительных данных
5. Сохраняет результаты сканирования для дальнейшего анализа в внутреннюю БД

Нound – это инструмент для поиска чувствительных данных



Что мы умеем искать

- ФИО
- Даты рождения
- Телефон
- Email
- Номер банковской карты
- СНИЛС
- Паспорта РФ, загранники, ID карты, свидетельства о рождении
- Серия и номер военного билета
- Серия и номер СТС, ПТС
- Секреты
- Ссылки на социальные сети
- IMSI, IMEI, IP
- Названия ЮЛ

Разметили - разграничиваем



Соккрытие и маскирование

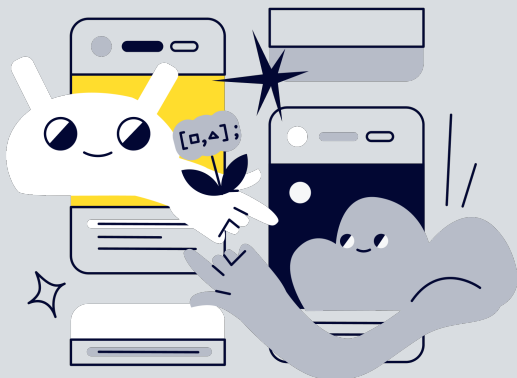
По умолчанию, чувствительные данные недоступны для просмотра с основной учетки на рабочем компе – разработали разные типы маскирования и соккрытия



Отдельный контур

А для работы с неприкрытыми чувствительными данными нужна VDI и специальная учетная запись

Контур для безопасной работы с чувствительным и данными









```
// сделали сами
изолированный пул VDI
ограниченный буфер обмена
отдельный фаервол и сетевые политики
специальные учетки, которые валидны только в контуре
ограничения доступа и функционала во внутренних сервисах

// пришлось доработать в системах
ЧД без маскирования и сокрытия - только в контуре
отдельные пользовательские песочницы
отдельные инстансы BI-инструментов
ограничены функции отправки данных
контроль пользовательского кода

// нам важно не сильно тормозить бизнес
выгрузка данных:
    только через автоматизированные проверки
or
    по согласованию с ИБ
```

Чего мы добились

-  Ограничили доступ в моменте
-  Сформировали процесс контроля и ресертификации доступа
-  Провели инвентаризацию огромного объема данных
-  Контролируем данные в источнике и при переиспользовании
-  Не усложняем жизнь бизнесу без причины
-  Дополнительная защита только там, где это необходимо

А самое главное – теперь можно выкинуть лишнее из процессов



**Как оказалось, бизнесу зачастую вообще не нужны
чувствительные данные – а значит, от них можно избавиться**

и уменьшить поверхность атаки

Выходит, проблема решена?!



Так чего тогда ты нас пугал?

Не так все просто и вот почему

ДЕЛИ ШЕС

/else



Приходится полагаться на дата-инженеров и админов



И пока никуда от этого не деться



Постоянные исключения

Потому что приходится подстраиваться
под имеющиеся процессы



Много легаси

От которого очень трудно избавиться (а
исправить почти невозможно)



Нет общего решения

И приходится много анализировать,
работать напильником и идти на
компромиссы

О чем стоит подумать заранее

Анализ безопасности данных лучше
сдвинуть как можно левее

Оценить масштаб компании
и количество сотрудников

Состав данных – ключевой
критерий

Сетевая сегментация,
разделение хранилищ,
резервное копирование et al.

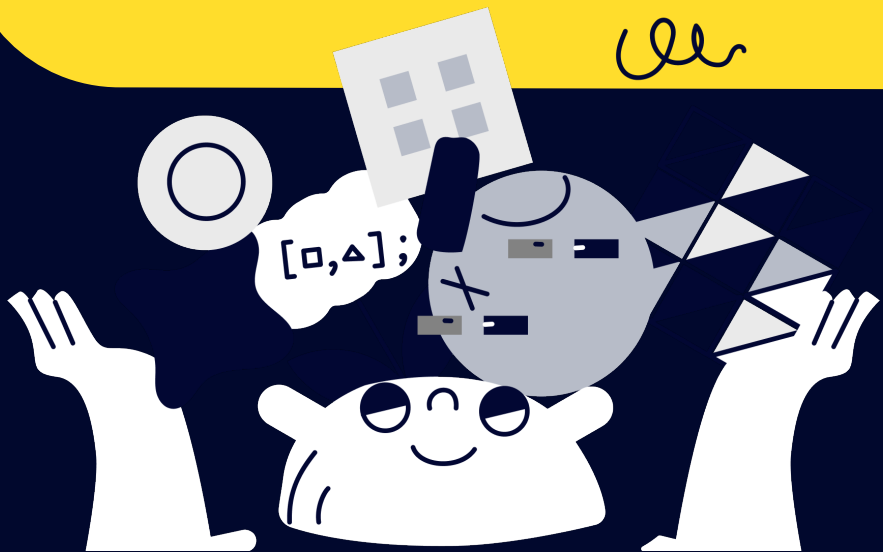
А не выкинуть ли вообще
чувствительные данные из
процесса?

Пользовательские сценарии

Разграничение доступа



Всегда пригодятся



- Bug Bounty
- Data Guard
- DLP
- Awareness
- немножечко доверия

Tinkoff CTF



- С 20 по 21 апреля 2024 г.
- 1-3 человека в команде
- Призы до 420 000 рублей на команду

***на правах рекламы**

ТИНЬКОФФ