РАЗМЫТИЕ СКРЫВАЕТ НЕ САМУ ИНФОРМАЦИЮ

оно распределяет пиксели,
но не удаляет данные

РАЗМЫТО



↓

ВОССТАНОВЛЕНО

TOKEN: 4B7K-9QX2

Исходные данные по-прежнему закодированы в размытии.
Ничего не удаляется — только меняется порядок.

@lionsoulbatyshchikov



ПЕРЕРИСОВКА И СОВПАДЕНИЕ

PIXELATION — ДЕТЕРМИИНИРОВАННОЕ УСРЕДНЕНИЕ

1 Выделяем ключевые элементы

шрифт, размер, фон

из чистых участков



2 Рендер каждого кандидата

та же сетка и блоки

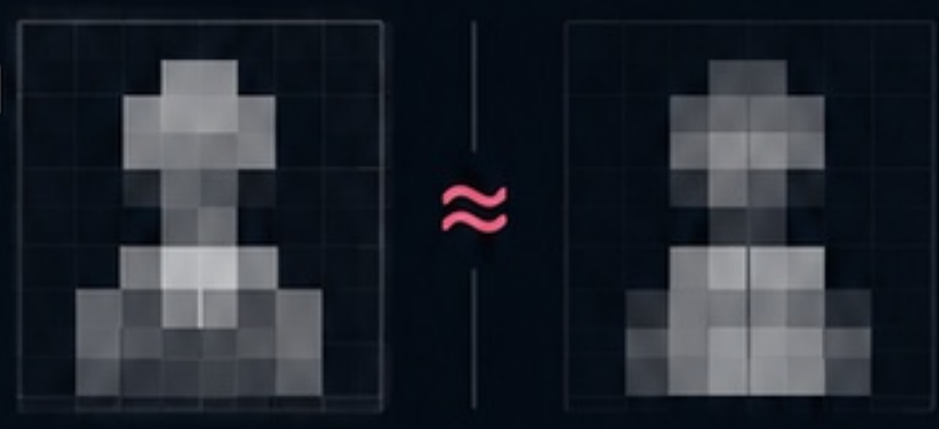
для всех вариантов



3 Pixelate и сравниваем

минимальный MSE

побеждает





Каждый мозаичный блок — это просто среднее значение (mean) всех пикселей внутри него.
Повторите тот же процесс усреднения — и кандидат с наименьшей ошибкой (MSE) будет оригиналом.

@lionsoulbatyshchikov

ДЕТЕРМИНИРОВАННЫЙ HASH



почему pixelation превращается в lookup,
а не в догадку

```
# усредняем пиксели по фиксированным блокам NxN
block_out = mean(block_in)  # детерминированно
# одинаковый вход → одинаковый выход
```

Depix: отрисовываем каждый глиф
в известном шрифте и пикселизуем тем же способом,
затем сверяем каждый блок через lookup.

> Сетка усреднённых блоков превращает
> восстановление в таблицу lookup
> по известному кабору символов.
> Геометрия решает любой блок,
> который соответствует более чем одному глифу.





@lionsoulbatyshchikov

BLUR – ЭТО СВЕРТКА (CONVOLUTION)

а у свертки есть обратная операция



РАЗМЫТИЕ = ИСХОДНОЕ ИЗОБРАЖЕНИЕ ⊛ KERNEL

ДЕКОНВОЛЮЦИЯ – УБИРАЕТ ЭФФЕКТ ⊛

Размытие можно представить как деление
по частотной области (Wiener / Richardson-Lucy).
Восстановление возможно до тех пор,
пока частота не стремится к нулю.

ГАУССОВО РАЗМЫТИЕ



TOKEN: 4B7K-9QX2

Применён гауссов фильтр.
Информация размыта равномерно.

ДЕКОНВОЛЮЦИЯ (WIENER)



TOKEN: 4B7K-9QX2

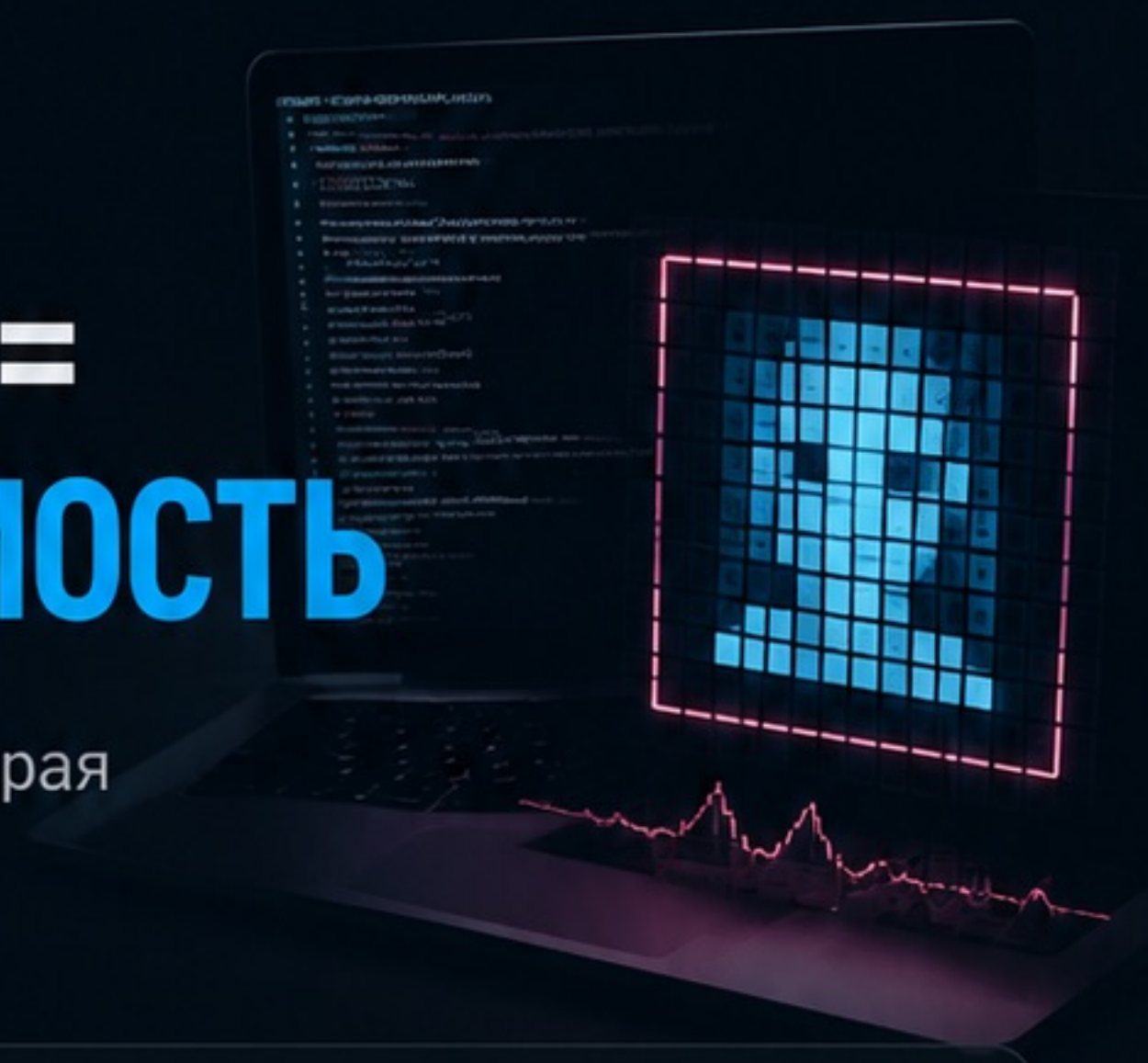
Применён Wiener deconvolution.
Текст восстановлен максимально точно.



Blur не защищает данные.
Зная kernel – восстановление вопрос времени.
Надёжная защита – только шифрование.

@lionsoulbatyshchikov

ЛИНЕЙНОСТЬ = ВОССТАНОВИМОСТЬ



единственная операция, которая
не разрушается

Σ	Blur и pixelate — линейные операции	информация переносится, а не теряется
≈	~400 кандидатов в секунду	один поток
■	Constant Fill — многие-к-одному	необратимая операция



Depix



Unredacter



Инструменты уже решают эту задачу:
Depix восстанавливает размытие,
Unredacter справляется с масками грубым перебором.

Единственная математически безопасная редакция — это **замена пикселей на константу**.



Ни blur, ни pixelate не защищают данные.
Зная **kernel** — вопрос лишь во времени.
Надёжная защита — это **шифрование**.

ЧТО ДЕЛАЕТ АНТИ-BLURRING

Open Source · Документирован
Для повышения осведомлённости

Depix

pixelation

Выполняет lookup по словарю блоков против отрендеренного набора символов. Обращает вспять линейное размытие.

Unredacter

pixelation

Алгоритм епископа Фокса. Брутфорс поиск и сопоставление рендера, устойчив к антиалиасингу.

ImageMagick / GIMP

blur

Деконволюция Wiener / Richardson-Lucy. Обращают вспять гауссово и движущееся размытие.

Другие инструменты

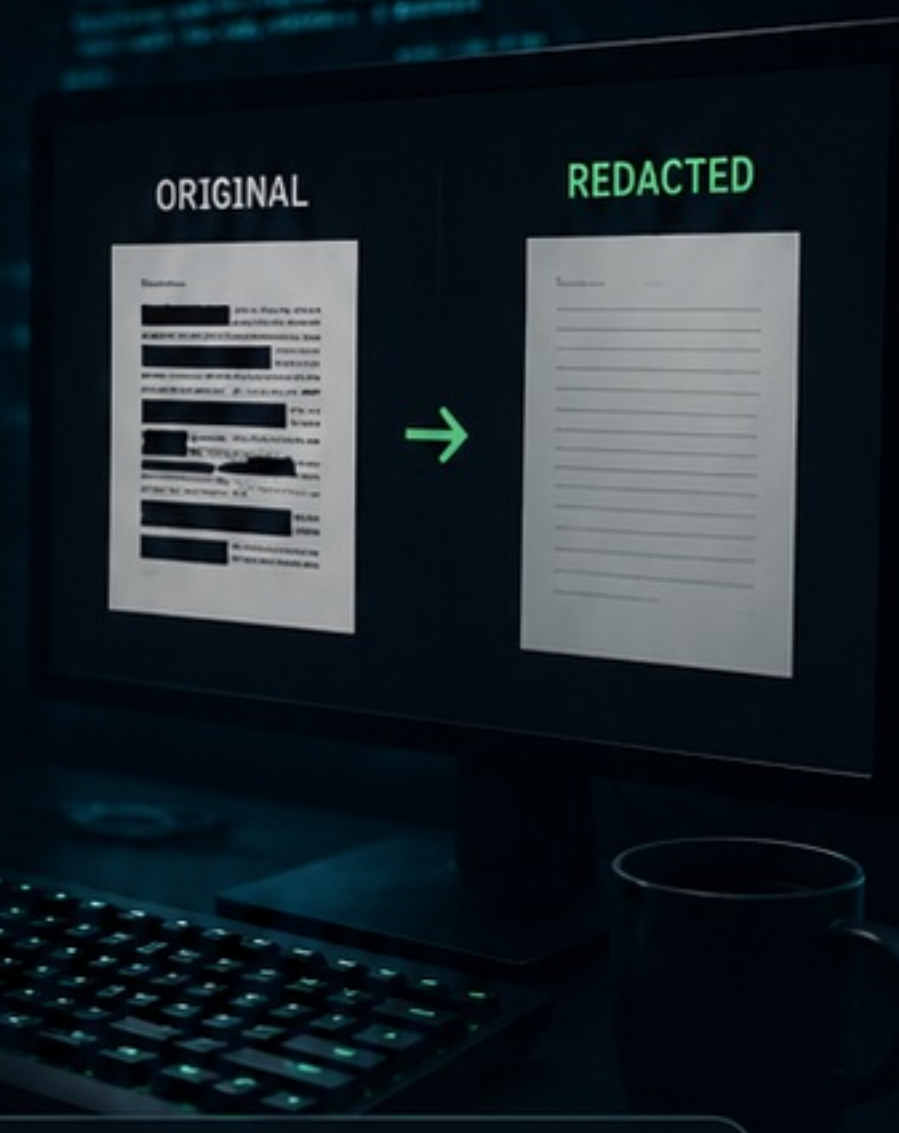
Целая экосистема утилит под разные типы искажения. Комбинируйте подходы для лучшего результата.

Если бесплатный инструмент справился за минуты — значит размытие не защищало данные.
Знайте **kernel** — управляйте временем восстановления.
Надёжная защита — это **шифрование**, а не размытие.

@lionsoulbatyshchikov

РЕДАКТИРУЙ, ЗАТЕМ ПРОВЕРЯЙ

инструменты, которые действительно
удаляют и доказывают это



Solid box + flatten

image

GIMP / Photoshop: залейте область поверх пикселей, затем выполните `Flatten Image` перед экспортом.



pdftotext `file.pdf`

verify

Выводит текстовый слой.
Если «redacted» всё ещё видно — редакция не удалась.



mat2 `file`

clean

Убирает метаданные и скрытые слои из изображений, PDF и документов за один проход.



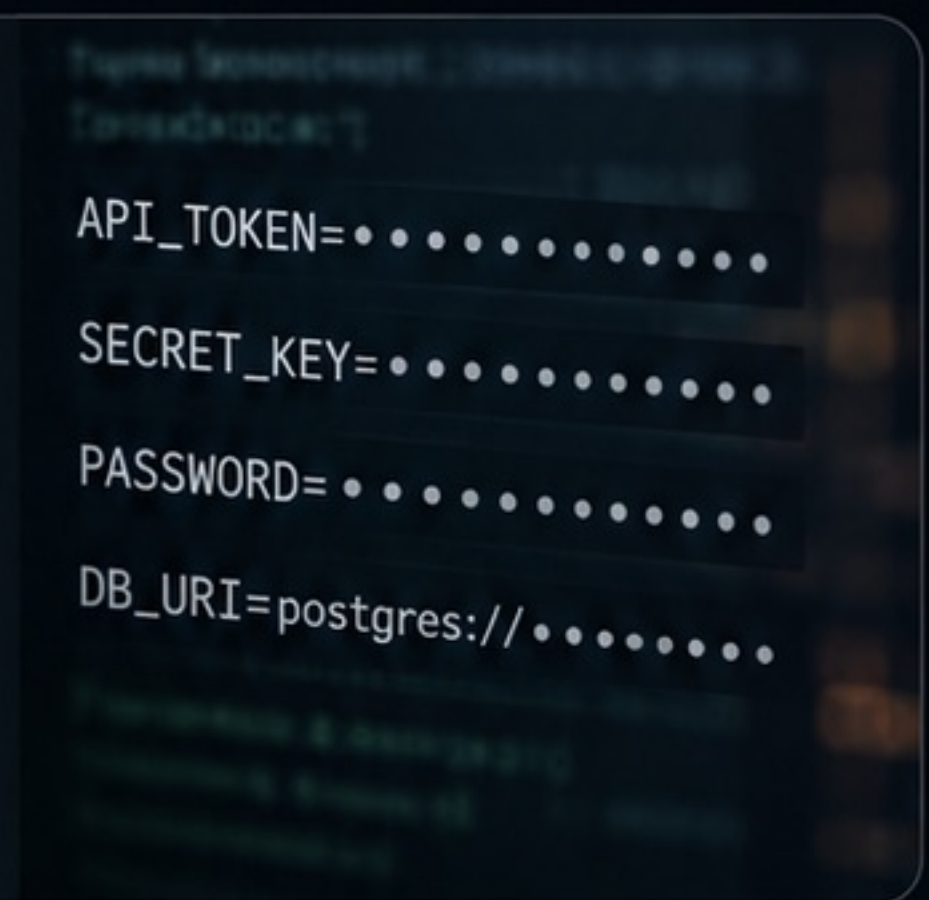
Сначала редактируйте, затем проверяйте.
Повторно извлекайте данные с помощью `pdftotext` или `exiftool`.
Доверяйте верификации, а не чёрному прямоугольнику.

@lionsoulbatyshchikov

ЧТО ЗАБЛУРИВАЮТ ЛЮДИ

Токены, ключи, пароли

Перед публикацией скриншота для форума или тикета — всё чувствительное должно быть скрыто.



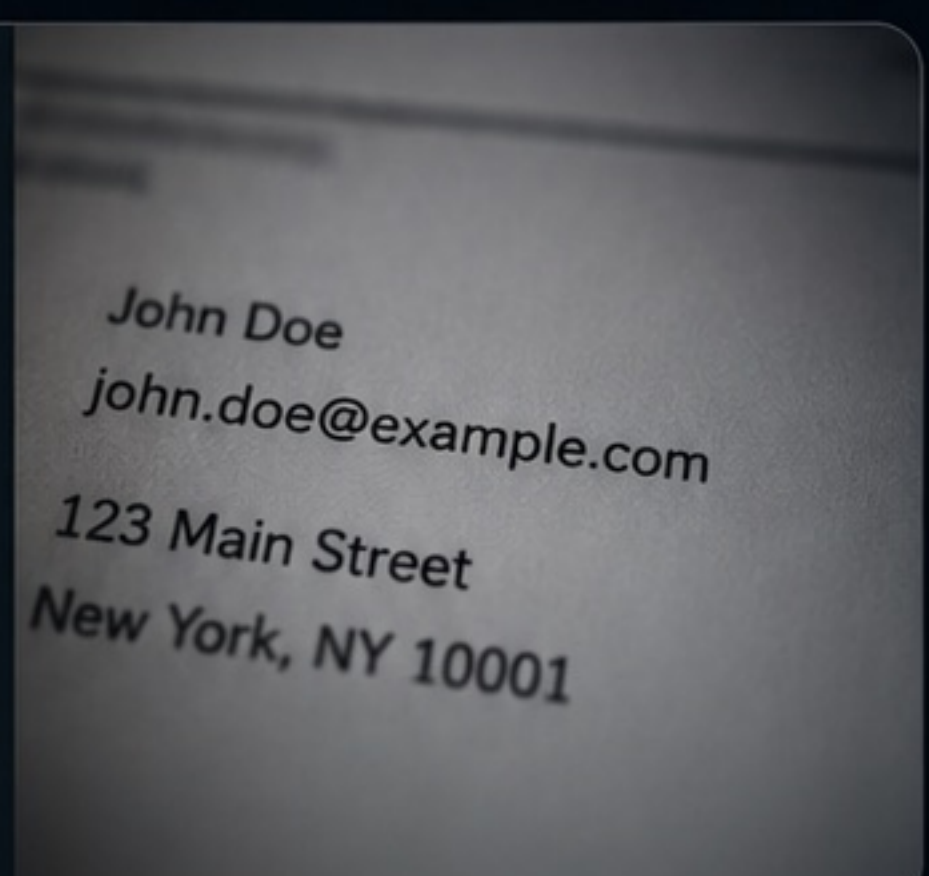
Номера карт и аккаунтов

Короткий и привычный формат — самый уязвимый для brute-force атак.



Имена, почты, адреса

В общих документах такие данные могут быть заблюрены, но при этом остаются легко восстанавливаемыми из того же источника.



Блюр ≠ защита.

Он скрывает — но не удаляет.
Данные можно восстановить.
Думайте как атакующий.



НЕ ВЕСЬ BLUR ОДИНАКОВО БЕЗОПАСЕН

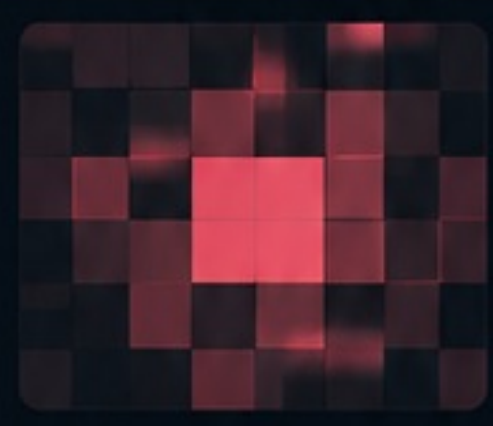
но по умолчанию ни один из них не защищён



Пикселизация / мозаика

восстанавливается

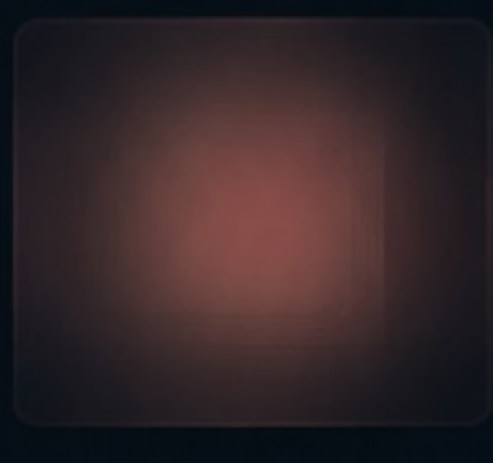
Обычно поддаётся **восстановлению** при наличии оригинала.



Размытие по Гауссу / box blur

часто восстанавливается

Часто **восстанавливается** при достаточном количестве данных.



Сплошное заполнение + выравнивание

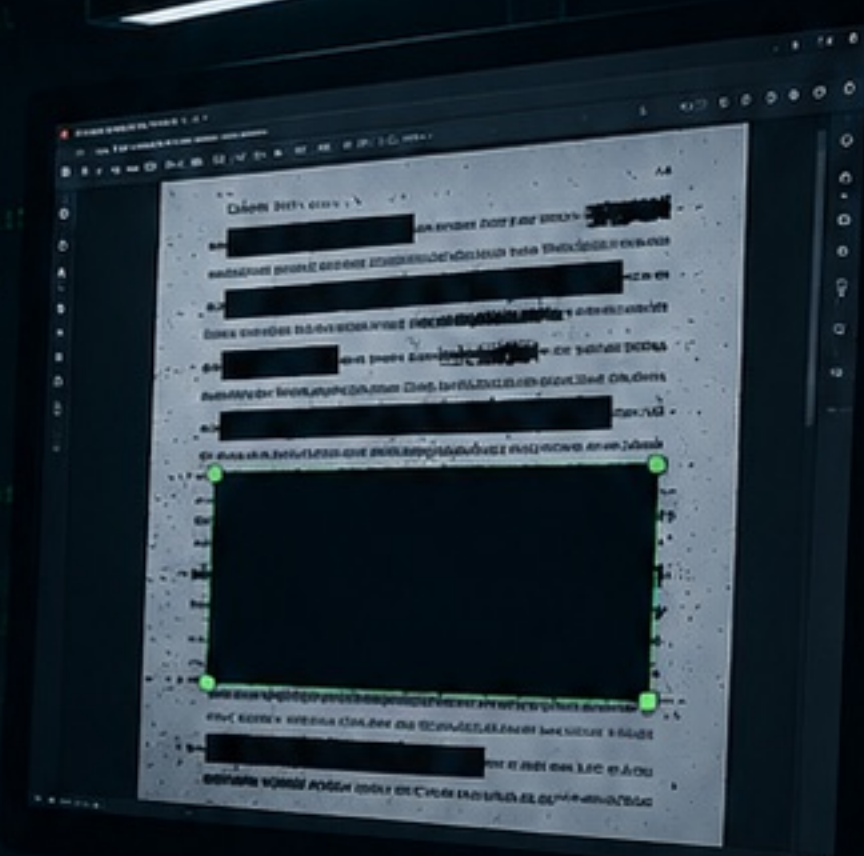
безопасно

Наиболее **безопасный** метод. Информация утрачивается.

Чем **сильнее blur**, тем выше риск восстановления. Единственный надёжный подход — **не искажать**, а **заменять пиксели**.

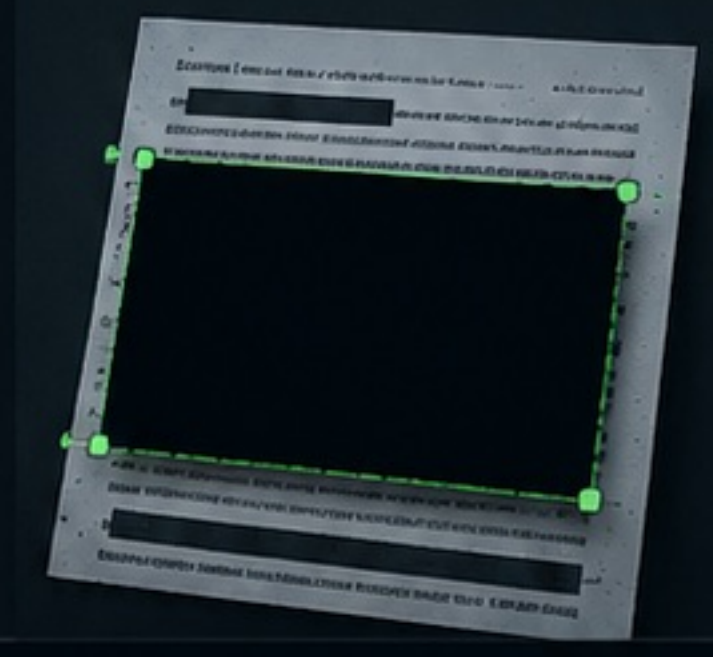
СКРОЙ, ЗАТЕМ ВЫРОВНЯЙ

заменяй пиксели, а не размывай их



Нарисуй сплошной непрозрачный блок

Закрой чувствительные области сплошным цветом.



Выровняй / экспортируй в изображение

Объедини слои и экспортируй в image, чтобы зафиксировать изменения.



Открой заново и проверь

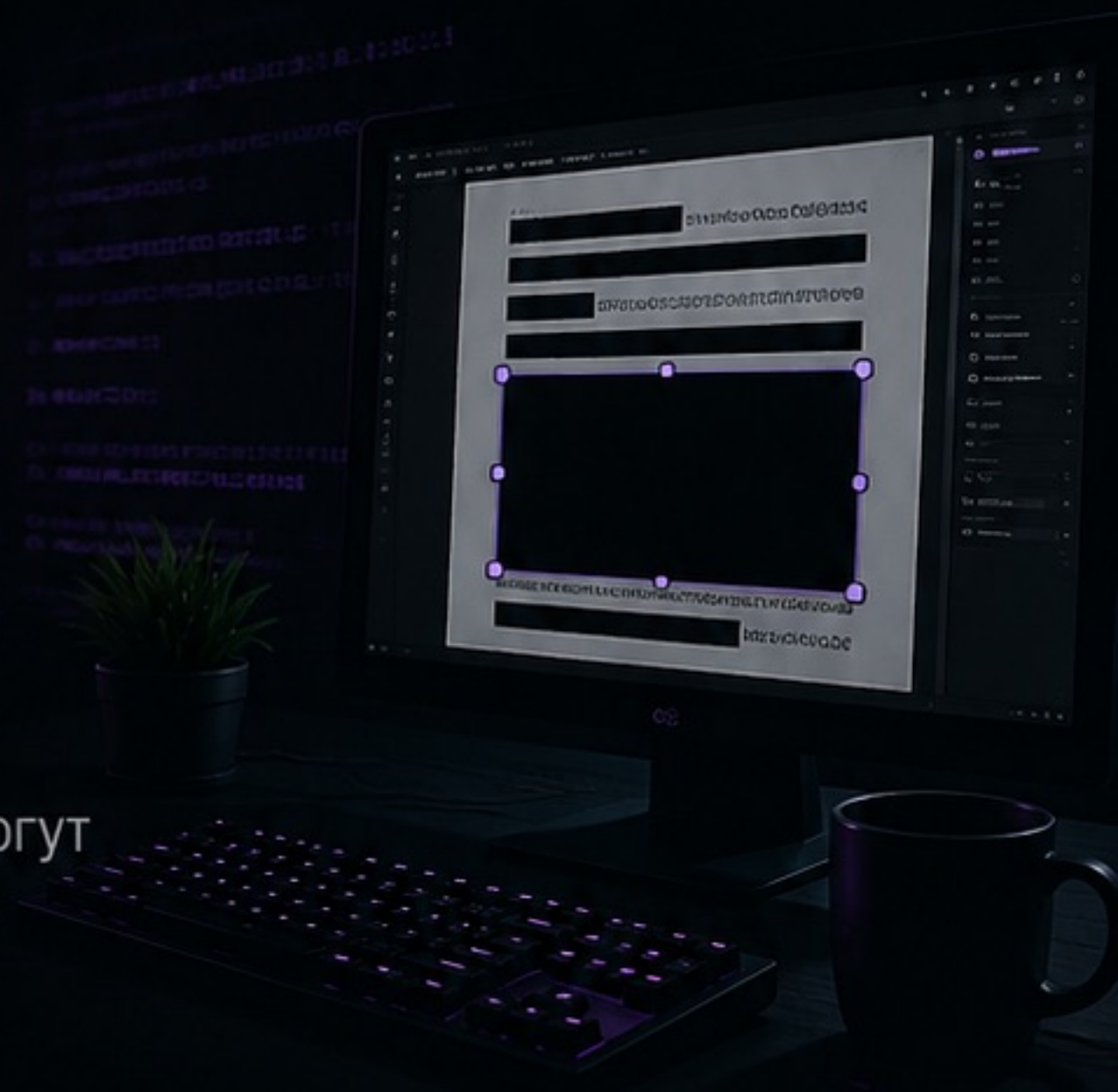
Убедись, что под блоком ничего не осталось.



Чёрный блок на **слоистом** файле (PDF, PSD) сохраняет текст под собой.
Всегда **выравнивай** и **перепроверяй**.

ЧЕК-ЛИСТ

быстрая проверка



Небольшие действия, которые помогут
не оставить лишнюю информацию
в ваших данных.

Не применяйте blur или pixelate для скрытия текста

Такие эффекты легко обратимы — текст
всё равно можно восстановить.



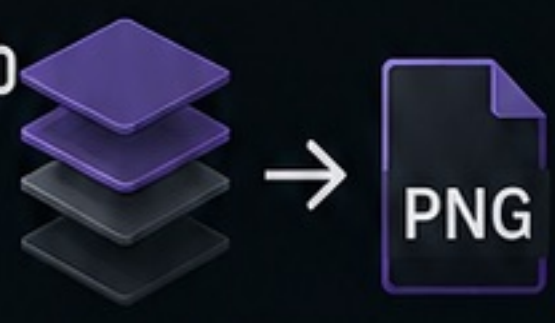
Используйте сплошной заливочный блок

Чёрный непрозрачный прямоугольник надёжно
закрывает информацию.



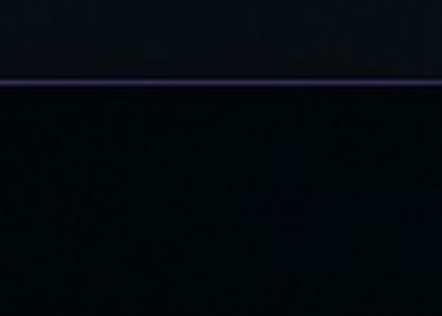
Выравнивайте и экспортируйте аккуратно

Перед экспортом `flatten` (сведение слоёв)
изображения.



Обрезайте секреты полностью

Удаляйте конфиденциальные данные целиком,
как только это возможно.



Blur — это визуальный эффект, а не redaction.

Цель — удалить данные без возможности восстановления.

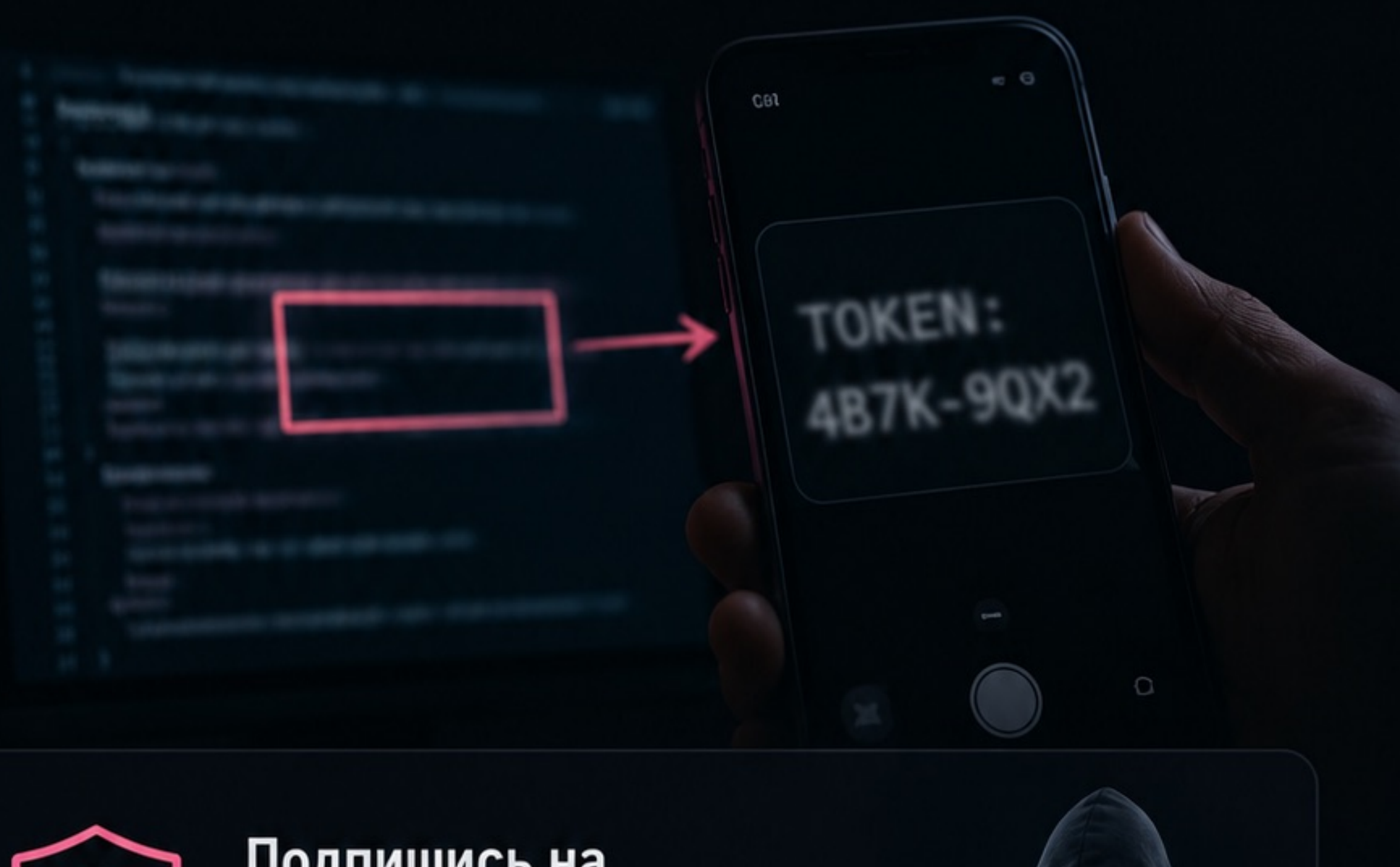
ПРОВЕРЬ СВОЙ СКРИНШОТ

безопасный способ проверки

текст читается

TOKEN: 4B7K-9QX2

Увеличьте любой **размытый текст** на скриншоте. Если вы можете угадать его содержимое — используйте скрипт.



Подпишись на @lionsoulbatyshchikov

Публикую безопасные инструменты, которые не раскрывают ваши данные. Отправляйте этот пост тем, кто размывает пароли на скриншотах.