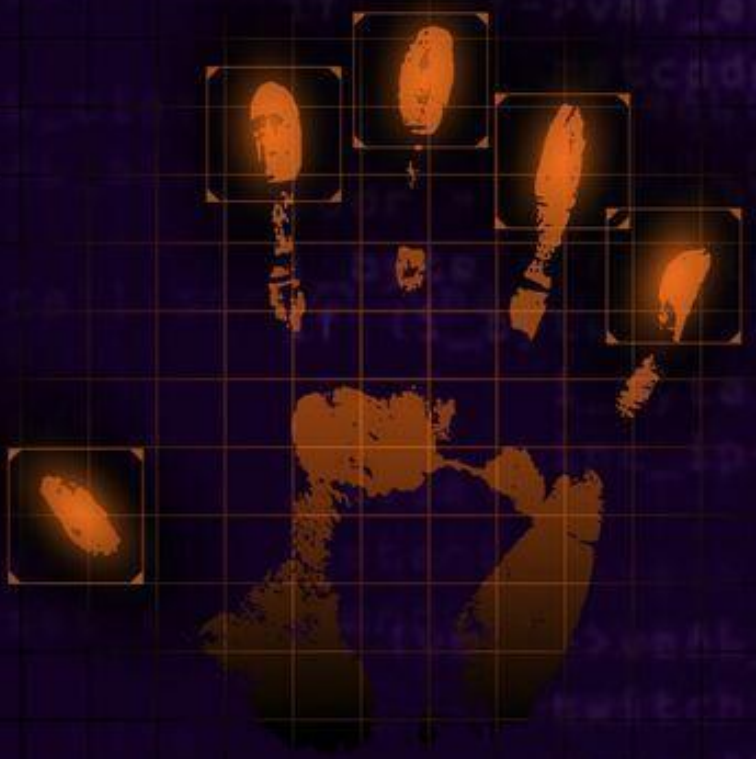
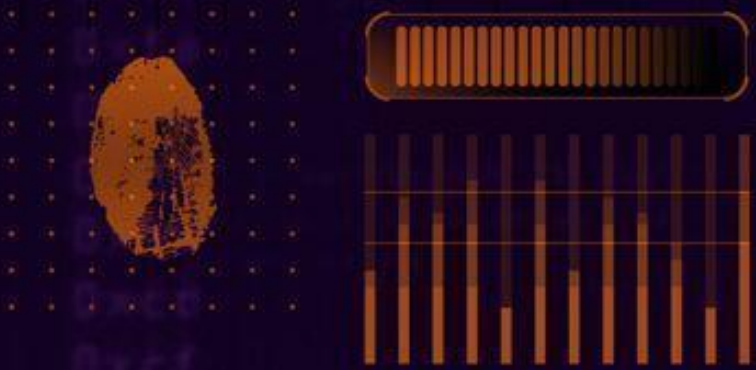
12+

Михаил Тарасов

ЛОГИ НЕ ВРУТ

практическая форензика для параноиков





Михаил Тарасов

Логи не врут. Практическая форензика для параноиков

«Издательские решения»

Тарасов М.

Логи не врут. Практическая форензика для параноиков /
М. Тарасов — «Издательские решения»,

Люди врут. Файловая система — никогда. Вы можете очистить «Корзину», включить режим Инкогнито и сжечь флешку, но операционная система помнит каждый ваш шаг.Эта книга — практический мануал по цифровому вскрытию. Внутри: поиск стёртых улик, вскрытие кэша мессенджеров, анализ реестра, извлечение паролей из ОЗУ и восстановление файлов после форматирования. Без академической воды — только рабочие инструменты и реальные методы расследования. Узнай, как найти правду, которую пытались скрыть.

Содержание

Логи не врут. Практическая форензика для параноиков

Михаил Тарасов

ISBN 978-5-0071-1609-1
Создано в интеллектуальной издательской системе Ridero

ВВОДНАЯ: АКСИОМА НУЛЕВОГО ДОВЕРИЯ

— Манифест параноика. Почему люди всегда врут, а машинный код — никогда

Добро пожаловать в профессию, где твоим лучшим другом станет недоверие.

Если ты взял эту книгу в руки, значит, ты либо хочешь научиться вытаскивать правду из цифрового небытия, либо ты тот самый парень, который пытается понять, где он прокололся. В любом случае, слушай внимательно.

За мои годы в форензике я усвоил один урок, который выжег бы татуировкой на лбу каждому стажеру, если бы это не противоречило корпоративной этике. Урок звучит просто: **Все врут.**

Это не цитата из сериала про хромого врача. Это фундамент нашей работы.

Подозреваемые врут, потому что боятся тюрьмы, штрафа или увольнения. Свидетели врут, потому что хотят казаться лучше, чем они есть, или просто забыли детали. Потерпевшие врут, потому что им стыдно, что они кликнули на ссылку «Увеличить член на 5 см» и поймали шифровальщика на сервер бухгалтерии. Даже заказчики расследования врут, потому что хотят услышать конкретный ответ, а не объективную истину.

Человеческая память — это самая ненадежная флешка в мире. Она перезаписывается эмоциями, стирается стрессом и форматируется самовнушением. Человек может искренне верить, что не отправлял то самое письмо. Он может пройти полиграф, глядя в глаза оператору, и клясться здоровьем мамы, что не копировал базу клиентов.

Но знаешь, кому плевать на его клятвы, слезы и эмоции?

Файловой системе NTFS.

Компьютер — это идеальный бюрократ. Он — тот самый зануда в офисе, который записывает каждый чих в журнал, ставит штамп с датой, подшивает в папку и дублирует копию в архив. Ему все равно, хороший ты человек или плохой. Ему все равно, хотел ты украсть деньги или просто «проверял систему безопасности».

Компьютер оперирует только двумя состояниями: 0 и 1. Было событие или не было. Записано или не записано.

И именно поэтому **машинный код никогда не врет**.

1. Анатомия лжи vs. Анатомия лога

Давай разберем классическую ситуацию.

Сидит перед тобой менеджер среднего звена. Глаза бегают, ладошки потные.

— *Я не был за компьютером в тот момент!* — кричит он. — *Я вышел покурить! Меня подставили! Это хакеры через Wi-Fi!*

Звучит убедительно? Для бабушки у подъезда — да. Для судьи, возможно, тоже. Но ты открываешь **System Event Log** (журнал системных событий) и смотришь на ID 4624 (Успешный вход в систему) или анализируешь **SRUM** (System Resource Usage Monitor).

И что ты видишь?

Ты видишь, что в ту самую минуту, когда он «курил», его учетная запись не просто была активна. Ты видишь, что курсор мыши двигался, открывался конкретный Excel-файл, а браузер Chrome обращался к облачному хранилищу.

Компьютер записал таймстемп с точностью до миллисекунды. Он записал хеш-сумму запущенного процесса. Он сохранил в **Prefetch** информацию о том, что программа для стирания файлов была запущена за 10 минут до твоего прихода.

Машина не умеет сговориться. Машина не умеет «забыть». Если запись есть в $MFT (Master File Table) — значит, действие было. Если в реестре остался ключ *UserAssist* — значит, программа запускалась. Точка. Никаких «но» и «если».

Наша работа — это перевод с языка бездушных фактов на язык человеческих доказательств. Мы не психологи. Нам не нужно гадать по мимике. У нас есть кое-что получше: **артефакты**.

2. Иллюзия пустоты (Почему «Удалить» — это миф)

Люди думают, что цифровой мир работает так же, как физический. Если ты сжег бумажку — её нет. Пепел развеяли, концы в воду.

Они нажимают «Очистить корзину» и с чувством выполненного долга идут пить кофе. Они думают, что перехитрили систему.

Наивные.

Для операционной системы (особенно Windows) удаление файла — это слишком затратная операция. Зачем стирать гигабайт нулей и единиц, тратить ресурс диска и время процессора? Система просто помечает в оглавлении (MFT): «Это место свободно. Если нужно будет записать что-то новое — пиши сюда».

Сами данные лежат на месте. Они лежат там час, день, неделю, иногда годы. Пока их случайно не перезапишет новая серия сериала или отчет за квартал.

И даже если они использовали шредер файлов (вайпер), который перезаписывает сектора нулями…

Остаются тени.

— **LNK-файлы (Ярлыки):** Windows заботливо создала ярлык для файла, чтобы пользователю было удобно его открывать. Файл стерт, а ярлык в папке *Recent* остался. И в нем — путь к файлу, время открытия и даже размер.

— **Jumplists (Списки переходов):** В панели задач сохранилась история.

— **Thumbnails (Миниатюры):** В базе thumbcache_*.db лежат маленькие копии картинок. Оригинала нет, а превьюшка есть. И по ней прекрасно видно, что это скан паспорта или фото с места преступления, а не «просто картинка с котиком».

Параноик внутри тебя должен ликовать. Система спроектирована так, чтобы **помнить**. Твоя задача — знать, где эта память хранится.

3. Эффект наблюдателя и принцип Локара

В криминалистике есть **принцип Эдмонда Локара**: *«Каждый контакт оставляет след»*. Преступник уносит с места преступления частицы пыли и оставляет свои отпечатки.

В цифровом мире этот принцип возведен в абсолют. Нельзя войти в систему и не наследить. Нельзя скопировать файл, не изменив время доступа (Last Access Time). Нельзя подключить флешку, не прописав её серийный номер в ветку реестра USBSTOR.

Но здесь кроется и главная опасность для тебя самого.

Как только ты прикасаешься к исследуемому компьютеру, ты становишься частью системы. Ты — загрязнитель.

Если ты включишь компьютер подозреваемого и просто загрузишь Windows, чтобы «посмотреть», ты изменишь сотни файлов. Обновятся логи запуска, временные метки, создадутся новые записи в реестре.

Поздравляю, ты только что уничтожил доказательства. Адвокат защиты порвет тебя в суде, как тузик грелку. Он скажет: *«Эти изменения внес эксперт. Откуда нам знать, что он не подбросил улики?»*. И будет прав.

Поэтому первое правило параноика-форензика: **Не трогай, пока не защитил.**

Мы используем блокираторы записи (Write Blockers). Мы работаем не с оригинальным диском, а с его побитовой копией (образом). Мы работаем в стерильных условиях виртуальных машин. Мы — хирурги, которые боятся занести инфекцию в рану, которая и так уже гноится.

4. Психология цифрового следа

Почему я называю этот подход «Манифестом параноика»?

Потому что ты должен подозревать каждый байт.

Пользователь переименовал virus. exe в report.docx?

Обычный юзер увидит иконку Word и успокоится. Параноик посмотрит на **сигнатуру файла** (Magic Bytes) в hex-редакторе и увидит MZ (исполняемый файл) вместо PK (архив документа Office).

Логи не врут. Расширение файла — врет. Иконка — врет.

Пользователь изменил дату создания файла на 2010 год, чтобы скрыть, что документ создан вчера?

Обычный юзер посмотрит в «Свойства» и поверит. Параноик полезет в атрибуты $STANDARD_INFORMATION и $FILE_NAME внутри записи MFT и увидит расхождение во времени (Timestomp).

Системное время можно скрутить. Но миллисекунды в журнале USN Journal сдать тебя с потрохами.

Мы ищем аномалии. Мы ищем несостыковки.

Если в логах чистота и порядок, как в операционной — это самый тревожный знак. Это значит, что кто-то *очень старался* замести следы. А следы затирания следов — это, пожалуй, самая громкая улика из всех возможных. Нормальные люди не чистят Event Logs перед сном. Нормальные люди не запускают CCleaner с 35-кратной перезаписью свободного места, если им нечего скрывать.

5. Грань закона (Важное предупреждение)

И последнее в этой вводной части, но первое по важности.

Мы — исследователи, а не взломщики. Мы — цифровые детективы, а не черные хакеры.

Вся информация в этой книге — это оружие. Обоюдоострое.

Знание того, как найти следы, автоматически дает знание, как их спрятать (хотя, как мы выяснили, спрятать *всё* невозможно).

Но если ты решишь применить эти знания, чтобы взломать ноутбук бывшей, прочитать переписку конкурента или залезть в корпоративную сеть, куда тебя не звали — закрой книгу и сожги её.

Потому что тогда ты переходишь из категории «охотник» в категорию «дичь».

Статья 272 УК РФ (Неправомерный доступ к компьютерной информации), 138 УК РФ (Нарушение тайны переписки) и еще букет смежных статей — это не страшилки. Это реальность. И иронично будет, если тебя поймает такой же параноик, как я, используя методы из этой самой книги.

Он найдет твои логи. Он найдет твои удаленные инструменты. Он докажет твою вину.

Потому что ты — человек, и ты ошибаешься. А логи — нет.

Резюме раздела

— **Доверяй только данным.** Слова — шум. Нех-код — сигнал.

— **Компьютер помнит всё.** Даже то, что ему приказали забыть.

— **Не навреди уликам.** Работай с копиями, используй блокираторы.

— **Смотри глубже.** Интерфейс Windows создан для домохозяек. Истина лежит в реестре, базах данных и бинарных файлах.

— **Оставайся на светлой стороне.** Тюремная роба плохо сочетается с цветом лица айтишника.

Готов? Тогда надевай перчатки (метафорически), выключай жалость к пользователям и добро пожаловать в Лабораторию Параноика.

Переворачивай страницу. Мы начинаем Setup.

— Принцип Локара. «Каждый контакт оставляет след». Даже если ты просто посмотрел на файл

В 1910 году французский криминалист Эдмон Локар сформулировал принцип, который стал библией для всех сыщиков мира: *«Любой контакт между двумя объектами приводит к взаимному обмену следами»*.

В физическом мире это работает понятно. Вор разбил окно — на его одежде остались микрочастицы стекла, а на раме — волокна его свитера. Убийца прошел по ковру — на подошве остался ворс, на ковре — грязь с улицы. Это физика. Пыль, грязь, пот, кровь.

В цифровом мире всё намного хуже.

В цифровом мире не существует понятия «просто посмотреть». В цифровом мире акт наблюдения — это акт изменения. Это, черт возьми, квантовая механика, помноженная на бюрократию операционной системы.

Новички (и плохие оперативники) часто говорят одну и ту же фразу, которая становится эпитафией на могиле их расследования:

«Я просто включил компьютер, чтобы проверить, тот ли это ноутбук, и быстренько открыл папку „Документы“. Я ничего не менял! Я даже не сохранял ничего!»

Поздравляю, Шарик, ты балбес. Ты только что уничтожил целостность доказательства. Ты не просто посмотрел — ты растоптал цифровую сцену преступления своими грязными ботинками. И я объясню тебе, почему.

1. Эффект наблюдателя: Ты — главный вирус

Как только ты нажимаешь кнопку питания на исследуемом устройстве, ты запускаешь цепную реакцию. Ты будишь монстра.

Современная операционная система (особенно Windows 10/11) — это живой организм. Она не сидит и не ждет твоих команд. Она живет своей жизнью.

В момент загрузки:

— **Меняются логи системы.** Event Logs записывают событие включения (Event ID 12, 6005 и прочие). Ты сдвинул таймлайн. Теперь последнее действие на компьютере совершил не преступник, а ты.

— **Запускаются фоновые службы.** Индексация поиска, антивирус, обновление Windows, дефрагментация. Система начинает бешено перелопачивать файлы, обновляя метаданные.

— **Монтируются тома.** Даже если ты ничего не открывал, система прописывает флаги «Dirty bit» или обновляет журнал транзакций NTFS ($LogFile), фиксируя состояние файловой системы.

Ты еще даже не тронул мышку, а на диске уже изменились тысячи байт. Оригинальное состояние системы, каким оно было в момент изъятия, утеряно навсегда.

2. Ловушка «Просто посмотреть»: Таймстемпы MAC

Допустим, ты загрузился. Ты видишь файл План_захвата_мира.docx. Рука тянется кликнуть. Ты думаешь: *«Я просто открою, прочитаю и закрою. Содержимое ведь не изменится»*.

Ты забываешь про **MAC-времена**. Это святая троица метаданных файловой системы NTFS:

— **M (Modified):** Время изменения содержимого файла.

— **A (Accessed):** Время последнего доступа к файлу.

— **C (Created):** Время создания файла.

Как только ты открываешь файл, операционная система услужливо обновляет метку **Last Accessed**.

«Ну и что?» — спросишь ты.

А то. Представь, что алиби подозреваемого строится на том, что он не открывал этот файл в день убийства. Экспертиза могла бы показать, что последний доступ был месяц назад. Но ты открыл его сегодня. Теперь дата доступа — **сегодня**.

Ты только что стер доказательство невиновности (или вины). В суде адвокат скажет: *«Дата доступа совпадает с днем, когда ноутбук был у следователя. Откуда мы знаем, что это не следователь открыл файл?»*.

И судья выкинет эту улику в мусорку. А тебя — с работы.

3. ShellBags: Ты наследил, даже не открывая файлы

«Ладно,» — скажешь ты. — *«Я не буду открывать файлы. Я просто зайду в папку и посмотрю на список файлов»*.

Ха. Ты думаешь, ты умнее Microsoft?

Знакомься: **ShellBags**. Это ключи реестра (в ветке USRCLASS. DAT), которые запоминают настройки отображения папок. Размер окна, позиция иконок, тип сортировки.

Зачем это нужно? Чтобы Windows «помнила», как тебе удобно смотреть пор… простите, документы.

Как только ты открываешь папку «Секреты», Windows создает или обновляет запись в ShellBags: *«Пользователь заглянул в эту папку в 14:05»*.

Ты не открыл ни одного файла. Но ты оставил след в реестре, что папка была просмотрена. Ты изменил реестр. Ты изменил **User Hives**. Хэш-сумма файла реестра изменилась.

Целостность данных нарушена. Шах и мат.

4. Хэш-сумма: Цифровая ДНК и принцип хрупкости

Чтобы понять масштаб катастрофы, нужно понять концепцию **Хэш-суммы**.

Мы поговорим об этом подробнее в главе про финализацию, но сейчас ты должен усвоить базу.

Хэш (MD5, SHA-1, SHA-256) — это уникальный цифровой отпечаток файла или всего диска.

Если у тебя есть файл «Война и мир» и ты изменишь в нем хотя бы одну запятую, или даже просто поменяешь одну букву с заглавной на строчную — хэш-сумма изменится до неузнаваемости. Это называется «лавинный эффект».

В профессиональной форензике процесс выглядит так:

— Изъяли диск.

— Подключили через защиту.

— Посчитали хэш-сумму всего диска. Записали в протокол.

— Сделали копию.

— Посчитали хэш копии. Он должен **совпадать байт-в-байт**.

Если ты, умник, включил компьютер подозреваемого *до* снятия образа, ты изменил состояние диска.

Допустим, ты потом передал этот диск в лабораторию ФСБ. Эксперт считает хэш, и он не совпадает с тем, что могло быть зафиксировано ранее (или просто логи показывают активность после изъятия).

Любое изменение — это сомнение. В юриспруденции: **«Все сомнения толкуются в пользу обвиняемого»**.

Твоё любопытство подарило преступнику свободу.

5. USB-гигиена: Не суй куда попало

Принцип Локара работает и в обратную сторону. Не только ты оставляешь следы на компьютере, но и компьютер оставляет следы на твоем оборудовании.

Или ты оставляешь следы своего оборудования на компьютере.

Классическая ошибка: *«Я вставлю свою флешку, чтобы скинуть туда найденные документы»*.

Бум! Ты только что прописал серийный номер, вендора и ID своей флешки в ветку реестра HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR.

Теперь этот компьютер знает твою флешку.

А на флешке система создаст скрытую папку System Volume Information и попытается проиндексировать содержимое.

Ты связал себя с уликой. Если на твоей флешке были файлы с других дел (или, не дай бог, твои личные фото), и ты занес их следы или вирусы на комп подозреваемого — ты загрязнил улику. Это называется **Cross-contamination** (перекрестное загрязнение). В биолабораториях за такое увольняют. В форензике — сажают за фальсификацию.

6. Как не стать слоном в посудной лавке?

Параноик выживает, потому что знает правила игры. Чтобы обойти принцип Локара (точнее, минимизировать его последствия), мы используем «цифровые презервативы».

А. Блокираторы записи (Write Blockers)

Это железная коробочка, которая ставится между диском подозреваемого и твоим компьютером.

Она физически перерезает провод «Запись». Команды на чтение проходят. Команды на запись (Write, Delete, Format) — блокируются на уровне контроллера.

С этой штукой ты можешь хоть обсмотреться. Windows будет посылать команды «обнови дату доступа», «создай Thumbs. db», но блокиратор скажет: *«Пошел нахрен, бро»*. Диск останется девственно чистым.

Если у тебя нет «железного» блокиратора — используй программный (в реестре Windows WriteProtect или монтирование в Linux в режиме Read-Only). Но железо надежнее.

Б. Работа с мертвой системой (Post-Mortem)

Мы почти никогда не работаем с «живым» включенным компьютером, если только не ищем ключи шифрования в оперативной памяти.

Стандарт: выдернуть шнур из розетки (да, жестко, но иногда необходимо, чтобы не сработали скрипты самоуничтожения). Достать диск. Подключить через блокиратор. Снять образ.

Вернуть диск на место. Положить в пакетик. Опечатать.

Работать ТОЛЬКО с образом.

В. Стерильная среда

Твой компьютер для анализа должен быть чист. Если ты анализируешь малварь на машине, где у тебя залогинен личный Telegram и лежит скан паспорта — ты идиот.

Используй Виртуальные машины (VM). Нагадил — удалил снапшот — откатился. Чистота — залог свободы.

Резюме раздела

Принцип Локара в цифре беспощаден.

— Ты посмотрел на файл? Ты изменил метаданные.

— Ты открыл папку? Ты изменил реестр.

— Ты подключил флешку? Ты наследил в драйверах.

— Ты просто загрузил систему? Ты изменил сотни логов.

Запомни: **В форензике нет пассивного наблюдения.** Любое твое действие — это вторжение.

Твоя задача — сделать это вторжение контролируемым, задокументированным и, по возможности, обратимым. Или работать с копией, которую не жалко.

Прежде чем нажать любую кнопку, задай себе вопрос параноика: *«Что именно изменится в системе, когда я это сделаю?»*.

Если ты не знаешь ответа — убери руки от клавиатуры.

Теперь, когда я тебя достаточно запугал, давай поговорим о том, когда эти правила МОЖНО нарушать. Да, иногда приходится работать «по-живому». Но для этого нужно знать красные флажки.

Переходим к следующему разделу.

— Красные флажки. Три ситуации, когда нужно закрыть ноутбук и звонить адвокату

Мы подошли к самой неприятной части. К той черте, которая отделяет уважаемого исследователя информационной безопасности от фигуранта уголовного дела.

В фильмах про хакеров главный герой взламывает сервер Пентагона за 30 секунд, а потом пьет кофе. В жизни, если ты случайно зайдешь не туда, к тебе в 6 утра придут вежливые люди в масках, положат лицом в пол и изымут всю технику, включая умный чайник.

Форензика — это работа с информацией. А информация бывает разной. Бывает открытой, бывает личной, а бывает охраняемой законом так строго, что даже попытка на неё посмотреть — это уже состав преступления.

Ты можешь быть тысячу раз прав этически («Я же искал доказательства измены!» или «Я просто проверял безопасность фирмы!»), но Уголовному кодексу плевать на твои мотивы. Ему важен **состав деяния**.

Ниже — три «Красных флажка». Три ситуации-триггера. Если в процессе работы ты натыкаешься на одну из них — **STOP**. Немедленно. Убери руки от клавиатуры. Не сохраняй ничего. Не копируй. Закрой ноутбук. И думай, как грамотно «съехать» с темы, пока не поздно.

Красный флаг №1. Чужая тайна без ордера (Или: «Я просто хотел помочь»)

Ситуация:

Тебя попросил знакомый бизнесмен: *«Слушай, проверь комп моего бухгалтера, мне кажется, она ворует»*.

Ты, как крутой спец, приходишь, снимаешь образ диска, начинаешь копать. И находишь переписку бухгалтера в Telegram. Не про деньги фирмы. А личную. Интимную. Или медицинские документы.

В чем проблема:

Ты только что нарушил **Статью 138 УК РФ (Нарушение тайны переписки)** и, возможно, **Статью 137 УК РФ (Нарушение неприкосновенности частной жизни)**.

Разбор полетов:

Компьютер может принадлежать компании. Но *Личность* сотрудника охраняется Конституцией.

Если бухгалтер подписала бумагу, что «Работодатель имеет право мониторить рабочие средства связи», ты еще можешь попытаться оправдаться (и то, суды часто встают на сторону работника, если речь о *личной* почте, открытой в *рабочем* браузере).

Но если такой бумажки нет, или ты полез в личный смартфон, который лежал на столе — ты преступник.

Грань очень тонкая.

— Найти файл otkat_schema. xlsx на рабочем столе — это форензика (корпоративное расследование).

— Подобрать пароль к архиву Photos_from_Turkey. rar, найти там голые фотки и показать заказчику — это уголовка.

Что делать параноику:

Прежде чем начать корпоративное расследование, требуй у заказчика **Письменное согласие на обработку данных** или **Приказ по организации**, где четко прописано, что техника принадлежит фирме и приватности на ней нет.

Если ты работаешь как фрилансер по просьбе «друга» — никогда, слышишь, НИКОГДА не лезь в личные папки (Photos, Social Media History), если они явно не связаны с предметом поиска (хищением денег). Увидел личное — отверни. Не скринь. Не сохраняй. Забудь.

Красный флаг №2. Выход за периметр (Или: «А что там на сервере?»)

Ситуация:

Ты анализируешь компьютер, зараженный вирусом. Ты видишь в логах, что вирус «стучится» на какой-то IP-адрес (C2-сервер злоумышленников).

В тебе просыпается охотничий азарт. *«Сейчас я узнаю, кто это сделал!»* — думаешь ты.

Ты запускаешь Nmap, сканируешь порты этого удаленного сервера, находишь уязвимость, проникаешь туда, чтобы скачать логи хакера и доказать его вину.

В чем проблема:

Поздравляю, Робин Гуд. Ты только что заработал **Статью 272 УК РФ (Неправомерный доступ к компьютерной информации)**. Возможно, группой лиц. Возможно, с использованием служебного положения.

Разбор полетов:

В тот момент, когда ты вышел за пределы *своего* (или доверенного тебе) компьютера и полез на *чужой* сервер (пусть даже сервер преступника) — ты сам стал хакером.

Закон не делает различий между «хорошим взломом ради справедливости» и «плохим взломом».

Несанкционированный доступ есть? Есть.

Ты не имеешь права «хакать в ответ» (Hack Back). Это прерогатива государственных спецслужб, и то с кучей оговорок.

Что делать параноику:

Твоя юрисдикция заканчивается на сетевой карте исследуемого компьютера.

Все, что улетело в сеть — это уже зона OSINT (разведки по открытым источникам).

— Посмотреть Whois домена хакера? Можно (это открытая инфа).

— Найти этот IP в базах угроз (VirusTotal)? Можно.

— Просканировать порты или пытаться подобрать пароль к админке хакера? **КРАСНЫЙ ФЛАГ.**

Если ты видишь, что следы ведут на внешний сервер, ты фиксируешь этот факт в отчете и пишешь: *«Дальнейший анализ требует полномочий правоохранительных органов»*. И ставишь точку. Не геройствуй.

Красный флаг №3. ЦП (Запрещенный контент)

Ситуация:

Это самый страшный сон любого форензика.

Ты анализируешь диск на предмет, скажем, финансового мошенничества. Прогоняешь карвинг картинок (восстановление удаленных фото).

И вдруг среди восстановленного мусора ты видишь ЭТО. Материалы с участием несовершеннолетних (CSAM — Child Sexual Abuse Material). Или экстремистские материалы.

В чем проблема:

Хранение, распространение и оборот. **Статья 242.1 УК РФ** и смежные.

Проблема в том, что как только эти файлы появились на *твоем* мониторе и сохранились на *твоем* диске (в папке результатов Autopsy), ты технически стал обладателем этого контента.

Если ты просто удалишь это и сделаешь вид, что не видел — ты можешь попасть под «Укрывательство» или «Несообщение о преступлении» (в зависимости от тяжести).

Если ты сохранишь это «для отчета», чтобы показать заказчику — ты занимаешься распространением.

Разбор полетов:

Это минное поле. Любое неверное движение может сломать тебе жизнь.

Обычные гражданские договора о неразглашении (NDA) здесь не работают. Если ты нашел криминал федерального масштаба, никакой NDA не защитит тебя от обязанности сообщить, и никакой NDA не защитит тебя от обвинения в хранении, если у тебя дома найдут этот жесткий диск.

Что делать параноику:

— Как только увидел превью — **СТОП**. Не открывай файл на полный экран.

— Ничего не копируй себе. Не пересылай заказчику.

— Зафиксируй факт (время, место, имя файла, хэш-сумму, если успел посчитать).

— Если ты работаешь официально в компании — немедленно пиши докладную записку руководителю службы безопасности. Пусть у него голова болит.

— Если ты фрилансер — это ситуация цугцванга. По закону ты должен сообщить в органы. Технически, это значит сдать своего заказчика. Этически — это единственный верный выбор, потому что педофилия — это зло, с которым мы не договариваемся.

— Но главное для тебя — **не храни это у себя**. Диск с таким контентом должен покинуть твой дом/офис как можно быстрее (передан полиции или возвращен владельцу под протокол с отказом от работы).

Резюме раздела: Техника безопасности свободы

Форензика — это не игра. Мы копаемся в грязном белье, и иногда это белье токсично.

Запомни три правила выживания:

— **Не копай глубже ордера.** Если тебе разрешили искать Excel-файлы, не лезь в папку с порно. Лишние знания — лишние сроки.

— **Не выходи из комнаты.** Не взламывай внешние сервера, даже если они вражеские.

— **Не становись хранилищем запрещенки.** Увидел криминал — сбрасывай ответственность на тех, у кого есть погоны.

В этой книге я буду учить тебя искать. Но я не могу дать тебе мозги, чтобы вовремя остановиться. Этот навык тебе придется прокачать самому.

Паранойя — это не только поиск врагов. Паранойя — это, прежде всего, защита себя от глупых ошибок.

А теперь, когда мы разобрались с законами и моралью, и ты все еще не передумал…

Давай наконец запустим эту чертову Виртуальную Машину и начнем делать вещи.

Конец вводной части.

(Переходи к Главе 1. Лаборатория параноика)

МОДУЛЬ 1. АНАТОМИЯ WINDOWS: ОПЕРАЦИОНКА-СТУКАЧ

Глава 1. Реестр (Registry): ДНК системы

— Почему Реестр помнит программы, которые удалили год назад

Если файловая система (NTFS) — это скелет Windows, то Реестр — это её мозг, память и подсознание одновременно. И, как у любого сложного организма с кучей детских травм (привет, Windows 95), у этого подсознания есть одна паршивая черта: оно ничего не забывает.

Большинство пользователей живут в сладкой иллюзии, которую им продала Microsoft. Эта иллюзия называется «Установка и удаление программ».

Пользователь нажимает «Удалить», видит прогресс-бар, нажимает «Готово» и думает, что программа исчезла. Он думает, что очистил диск. Он думает, что стер следы.

Наивный.

С точки зрения операционной системы, процесс деинсталляции — это ленивая уборка подростка перед приходом родителей. Он запихал грязные носки под кровать (удалил. exe файл), смахнул пыль с видного места (убрал ярлык с Рабочего стола), но оставил горы мусора в шкафу.

Этот шкаф — **Системный Реестр (Windows Registry)**.

В этом разделе мы разберем, почему Windows ведет себя как маньяк-коллекционер, где именно она хранит воспоминания о давно удаленном софте, и как мы, параноики-форензики, используем это против неё.

1. Философия цифрового мусора: Почему Windows так делает?

Прежде чем лезть в ветки HKEY_, нужно понять логику врага. Почему Microsoft не чистит реестр «под ноль»? Это не заговор рептилоидов (хотя…), это банальная инженерная лень и борьба за совместимость.

— Скорость важнее чистоты. Удаление ключей реестра требует времени. Если инсталлятор будет вычищать за собой каждую запись, процесс удаления займет 20 минут вместо 2. Пользователь будет ныть. Поэтому разработчики софта пишут деинсталляторы так: *«Удали папку Program Files и пару главных ключей, остальное — черт с ним»*.

— User Experience (UX). Если ты удалил игру, а через месяц решил поставить её снова, Windows хочет, чтобы твои настройки сохранились. Ей удобно помнить, где было открыто окно в последний раз.

— DLL Hell и Shared Components. Многие программы используют общие библиотеки. Если деинсталлятор удалит запись о библиотеке, могут сломаться другие программы. Поэтому золотое правило Windows: *«Не уверен — не удаляй»*.

В итоге, компьютер, которому больше года — это кладбище мертвых ключей. Для оптимизатора системы (типа CCleaner) это мусор. Для нас — это **золотая жила**.

2. Ульи (Hives): Где копать?

Реестр — это не один файл. Это набор файлов, которые называются «Ульи» (Hives). Они лежат на диске, и именно их мы вытаскиваем при анализе (даже если система не загружается).

Нас интересуют два главных направления:

— **SYSTEM** и **SOFTWARE** (лежат в C:\Windows\System32\config\). Это глобальная память компьютера. Тут записано то, что касается всех.

— **NTUSER. DAT** (лежит в C:\Users\%Username%\). Это святая святых. Личный файл каждого пользователя. Именно здесь хранится история конкретного человека: что он открывал, что запускал и какие настройки менял.

Когда ты работаешь с живой системой, ты видишь их как HKEY_LOCAL_MACHINE и HKEY_CURRENT_USER. Но форензик чаще работает с файлами. Запомни эти пути. Если ты не смог вытащить NTUSER. DAT подозреваемого — ты зря потратил время.

3. Артефакты бессмертия: Топ-3 предателя

А теперь переходим к «мясу». Где именно искать следы программы, которую удалили полгода назад?

А. BAM/DAM (Background Activity Moderator) — Стукач нового поколения

Путь: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ bam\State\UserSettings\ {SID_Пользователя}

Появился в Windows 10 (версия 1709) и стал подарком для экспертов. Изначально эта служба нужна, чтобы экономить батарею и ресурсы. Она следит за фоновыми процессами.

Но побочный эффект великолепен: BAM записывает **полный путь к исполняемому файлу** и **время последнего запуска**.

Даже если пользователь скачал hack_tool. exe, запустил его один раз, а потом удалил и перезаписал свободное место 35 раз — запись в ветке BAM останется.

Ты увидишь:

— C:\Users\Admin\Downloads\hack_tool. exe

— Таймстемп: *2025-10-05 14:23:11 UTC*

Это железобетонное доказательство. Файла нет. Но факт запуска зафиксирован службой ядра. Оспорить это в суде практически невозможно.

Б. MUICache — Память имен

Путь: HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft \Windows\Shell\MuiCache

Windows хочет быть дружелюбной. Когда ты запускаешь программу, система лезет внутрь. exe файла, ищет там красивое название (например, не vlc. exe, а «VLC Media Player») и кэширует его, чтобы красиво отображать в меню.

Этот кэш называется MUICache.

Фишка: Записи оттуда почти никогда не удаляются автоматически.

Если ты видишь в этом списке строчку:

C:\Temp\PortableApps\TorBrowser\firefox. exe — «Tor Browser»

Значит, Tor запускался. Даже если сейчас папки Temp уже не существует.

Это отличный способ ловить пользователей на использовании **Portable-софта** (который не требует установки). Они думают: *«Раз я не устанавливал, значит, в „Установке и удалении“ этого нет, значит, я чист»*.

MUICache смеется им в лицо.

В. UserAssist — Шифр Цезаря в 21 веке

Путь: HKEY_CURRENT_USER\Software\Microsoft\Windows \CurrentVersion\Explorer\UserAssist

Это мой любимый артефакт, просто потому что он показывает уровень «безопасности» от Microsoft.

UserAssist нужен, чтобы рисовать меню «Пуск» и показывать «Часто используемые программы».

Эта ветка хранит:

— Имя файла.

— Счетчик запусков (сколько раз запускали).

— Время последнего запуска.

Но есть нюанс. Microsoft решила «зашифровать» имена программ. Знаешь как? Алгоритмом **ROT13**.

Это когда буква A меняется на N, B на O и так далее (сдвиг на 13 символов). Это уровень шифрования «Записка на уроке в 5 классе».

Форензик-тулзы (или даже простые онлайн-декодеры) расшифровывают это мгновенно.

Находя в UserAssist запись о Veracrypt. exe, ты получаешь не просто факт наличия. Ты получаешь **паттерн поведения**.

— *Запусков: 1*. Значит, скачал, посмотрел, не понял, удалил. Возможно, любопытство.

— *Запусков: 542*. Значит, это рабочий инструмент. Человек жил в этой программе. Тут уже не скажешь «меня подставили».

4. ShimCache (AppCompatCache) — Король доказательств

Если бы мне разрешили взять на необитаемый остров только один артефакт реестра, я бы взял **ShimCache**.

Путь: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager \AppCompatCache

Это часть механизма совместимости приложений (Application Compatibility Shim). Windows отслеживает проблемы совместимости файлов, чтобы старый софт работал на новой ОС.

Почему это бомба:

— Охват: Он хранит информацию о до **1024** последних исполняемых файлах.

— Живучесть: Данные хранятся в памяти и сбрасываются в реестр при выключении компьютера. Удалить запись оттуда вручную невероятно сложно (нужно чистить ветку реестра, что требует прав SYSTEM).

— Следы: Он фиксирует имя файла, путь, размер и **время последней модификации файла** (не запуска, а именно изменения самого файла).

— Главное: Он помнит файлы, которые даже не существовали на диске долгое время (например, запущенные с флешки).

Кейс из жизни:

Подозреваемый утверждает, что никогда не работал с программой для подделки печатей Stamp. exe.

Мы лезем в ShimCache. Находим там запись E:\Soft\Stamp. exe.

Диск E: — это флешка. Флешки сейчас в компьютере нет.

Но ShimCache помнит. Он говорит нам: *«Эй, бро, файл Stamp. exe лежал на диске E:, и у него была дата изменения 01.01.2024»*.

Всё. Связь установлена. Мы доказали, что софт запускался с внешнего носителя.

5. Иллюзия «Чистильщиков» (CCleaner и аналоги)

«А если я запущу CCleaner?» — спрашивает каждый второй «мамкин хакер».

Да, чистильщики реестра удаляют «битые» ссылки. Они могут подчистить MUICache (иногда). Они могут грохнуть списки недавних документов.

Но они **почти никогда** не трогают глубокие системные ветки типа BAM или ShimCache, потому что это опасно для стабильности системы.

Более того, сам факт наличия и запуска CCleaner оставляет такие жирные следы в том же Prefetch и UserAssist, что для форензика это красный флаг.

Если я вижу, что в 18:00 был запущен cleaner. exe, а инцидент произошел в 17:50 — это называется **«Противодействие следствию»** и уничтожение улик. Юридически это часто хуже, чем то, что пытались скрыть.

Практический вывод для параноика

Реестр — это книга жизни твоей системы. Каждая установленная программа, каждый запущенный portable-софт, каждая вставленная флешка прописывается в его ДНК.

Удалить файл с диска — это как вырвать страницу из книги. Текст пропал, но нумерация страниц нарушена, корешок книги поврежден, а в оглавлении (Реестре) глава всё еще значится.

Чтобы реально вычистить программу из реестра, нужно понимать его структуру лучше, чем инженеры Microsoft, и править HEX-коды вручную. 99,9% преступников этого не умеют.

Они надеются на кнопку «Удалить».

А мы надеемся на их глупость. И, как показывает практика, логи (и реестр) не врут, а надежды нас не подводят.

— UserAssist и MUICache: Как доказать, что софт не просто лежал на диске, а запускался

В уголовном праве и корпоративных расследованиях есть гигантская пропасть между двумя понятиями: **«Владение»** и **«Использование»**.

Представь ситуацию. Ты находишь на компьютере главного бухгалтера файл password_cracker. exe.

Бухгалтер делает большие глаза и кричит:

— Это не моё! Это вирус скачал! Или сисадмин Вася скинул мне на флешку год назад вместе с фотками с корпоратива, я даже не знала, что это там лежит! Я не хакер, я жертва!

И знаете что? Если у тебя нет доказательств запуска, адвокат развалит обвинение за пять минут. Владение хакерским софтом (если это не спецсредства для негласного съема информации) само по себе редко является преступлением. Может, человек коллекционирует вирусы. Хобби у него такое.

Чтобы прижать подозреваемого к стенке, нам нужно доказать **Intent (Умысел)** и **Action (Действие)**. Нам нужно доказать, что курсор мыши навел именно человек, что он кликнул два раза, и что программа отработала.

Для этого в Реестре Windows существуют два главных предателя: **UserAssist** и **MUICache**. Они работают в тандеме, как плохой и хороший полицейский, и вместе они поют такую песню, от которой у защиты вянут уши.

1. UserAssist: Счетовод вашей паранойи

Если бы операционная система была человеком, UserAssist был бы тем самым назойливым менеджером, который стоит у тебя за спиной с секундомером и блокнотом.

Эта функция является частью оболочки Windows Explorer. Её официальная цель — определять, какие программы вы используете чаще всего, чтобы заботливо подсовывать их вам в меню «Пуск» или на Панель задач.

Но для форензика UserAssist — это **Святой Грааль поведенческого анализа**.

Где живет:

HKEY_CURRENT_USER\Software\Microsoft\Windows
\CurrentVersion\Explorer\UserAssist

Внутри этой ветки ты увидишь странные папки с названиями типа {CEBFF5CD-ACE2—4F4F-9178-9926F41749EA}. Это GUID-ы (Globally Unique Identifier).

Нас интересуют два основных:

— {CEBFF5CD-…} — здесь хранятся записи об Исполняемых файлах (.exe).

— {F4E57C4B-…} — здесь лежат данные о Ярлыках (.lnk).

Секрет Полишинеля: Шифрование ROT13

Когда ты откроешь эту ветку, ты увидишь список значений, похожий на бред сумасшедшего:

P:\Gbe\GbeOebmjre\gbe. rkr

F:\Uvqqra\fgrnyre. rkr

Не пугайся. Microsoft, в приступе гениальной «защиты от дурака», решила зашифровать имена запускаемых файлов алгоритмом **ROT13**.

ROT13 (Rotate by 13 places) — это шифр, который использовал Юлий Цезарь, только чуть модифицированный. Буква A меняется на N, B на O.

Это не защита. Это шутка. Любой форензик-инструмент (Registry Explorer, RegRipper) декодирует это на лету. И «бред» превращается в улики:

P:\Gbe… -> C:\Tor\TorBrowser\tor. exe

F:\Uvqqra… -> S:\Hidden\stealer. exe

Самое вкусное: Run Count и Last Execute Time

Расшифровав имя, мы смотрим внутрь бинарного значения (Binary Data). Там лежит не просто мусор, а структурированные данные. Самые важные для нас байты отвечают за два параметра:

— Run Counter (Счетчик запусков).

— Это число показывает, сколько раз пользователь запускал эту конкретную программу.

— И это — бомба.

— — Если счетчик = **1**, подозреваемый может сказать: *«Я случайно открыл, испугался и закрыл»*. Это правдоподобно.

— — Если счетчик = **54**, оправдания *«Я не знаю, что это такое»* больше не работают. Ты не можешь «случайно» запустить программу для взлома Wi-Fi 54 раза. Это система. Это рабочий процесс.

— Кстати, счетчик начинает отсчет с 0 (первый запуск) или 1, в зависимости от версии Win, но суть одна. Число больше 5 — это приговор версии о «случайности».

— Last Execute Time (Время последнего запуска).

— Таймстемп в формате FILETIME. Показывает дату и время, когда программу открывали в последний раз.

— Сравниваем это время с журналом входов (Security Log). Кто был залогинен в 14:15? Иванов. Программа запущена в 14:16. Вопросы есть? Вопросов нет.

— Focus Time (Время в фокусе).

— В некоторых версиях Windows (и в зависимости от настроек) UserAssist записывает не просто факт запуска, а сколько времени окно программы было *активным* (в фокусе).

— Это позволяет отсеять ситуации: *«Я запустил и ушел пить чай»*. Если Focus Time составляет 40 минут, значит, человек сидел и работал в этом окне. Вбивал настройки, читал мануалы, хакал.

Важный нюанс параноика:

UserAssist отслеживает только то, что запущено через **Windows Explorer** (Проводник).

— Двойной клик по иконке? Запишет.

— Запуск через «Выполнить» (Win+R)? Запишет.

— Запуск через командную строку (CMD/PowerShell)? **НЕТ.**

Если хакер профи и запускает всё через консоль, UserAssist будет молчать о конкретной утилите (но он запишет запуск самого cmd. exe или powershell. exe).

2. MUICache: Кладбище имен

Если UserAssist — это счетчик, то MUICache — это библиотекарь.

Полное название: **Multilingual User Interface Cache**.

Где живет:

HKEY_CURRENT_USER\Software\Classes\Local Settings\Software\Microsoft\Windows \Shell\MuiCache

(В старых системах путь может немного отличаться, но ищите Shell\MuiCache).

Зачем он нужен системе:

Когда вы запускаете файл word. exe, Windows хочет показать вам в меню или в диспетчере задач красивое название: «Microsoft Word», а не просто имя файла.

Система извлекает это имя из метаданных (ресурсов) самого. exe файла и сохраняет его в реестр, чтобы в следующий раз не сканировать файл заново.

Почему это важно для нас:

— Он помнит Portable-софт.

— Программы, которые не требуют установки, не оставляют следов в ветке Uninstall. Но как только пользователь кликает по «портативке», она попадает в MUICache.

— Нашли там запись G:\Soft\RubberDucky. exe со значением USB Attack Tool?

— Поздравляю. Вы доказали, что программа запускалась с внешнего диска G:.

— Устойчивость к удалению.

— Пользователь удалил программу. Пользователь почистил папку Recent. Пользователь даже (о ужас) очистил UserAssist специальным скриптом.

— Но про MUICache забывают почти все «чистильщики». Записи там висят годами.

— Это отличный способ поймать лжеца на: *«У меня никогда не стоял Telegram на рабочем компе!»*. Открываем MUICache, находим Telegram. exe -> Telegram Desktop. Шах и мат.

— Атрибуция малвари.

— Иногда вирусы маскируются под системные процессы. Например, файл называется svchost. exe, но лежит в папке C:\Temp\.

— В MUICache мы увидим: C:\Temp\svchost. exe.

— Настоящий svchost живет только в System32. Любой другой путь — это аномалия. MUICache подсвечивает эти аномалии как прожектором.

3. Синтез: Как связать их в цепочку доказательств

По отдельности эти артефакты сильны. Вместе — непобедимы.

Вот классический алгоритм работы форензика при анализе запуска софта:

Шаг 1. Экспортируем ветки NTUSER. DAT и USRCLASS. DAT.

Шаг 2. Загружаем их в **Registry Explorer** (инструмент Эрика Циммермана, бесплатен и гениален).

Шаг 3. Идем в UserAssist. Сортируем по Run Count.

— Видим Eraser. exe (программа для уничтожения данных).

— Запусков: 12.

— Последний запуск: вчера в 18:00.

Вывод: Пользователь намеренно использовал вайпер.

Шаг 4. Идем в MUICache.

— Ищем следы программ, которых нет в UserAssist (возможно, запускались давно и вытеснены ротацией, или запускались другим способом).

— Видим D:\Games\Doom. exe.

Вывод: На корпоративном ноутбуке играли. Даже если сейчас папка Games удалена.

Шаг 5. (Для профи) Сравниваем с **Prefetch**.

Prefetch (о котором будет отдельная глава) хранит следы выполнения за последние N дней/запусков.

Если в UserAssist есть запись о программе, а файла Prefetch для неё НЕТ — это **Красный Флаг**.

Это значит, что кто-то намеренно чистил папку C:\Windows\Prefetch, пытаясь скрыть следы.

Обычный пользователь туда не лезет. Удаление Prefetch — это признак квалифицированного противодействия расследованию. А это уже отягчающее обстоятельство.

Практическое задание для параноика

Прямо сейчас (не на боевой машине, а на своей домашней или виртуалке) сделай следующее:

— Открой regedit. exe.

— Перейди в HKEY_CURRENT_USER\Software\Microsoft\Windows \CurrentVersion\Explorer\UserAssist.

— Зайди в подпапку GUID (которая начинается на CE…).

— Посмотри на правую панель. Видишь эту абракадабру?

— Скопируй любое имя (например, P:\Jvaqbjf\flfgr32\pzq. rkr).

— Зайди на сайт rot13.com и вставь туда.

— Узнай, что ты запускал.

Ты удивишься, сколько мусора хранит твоя система. Там будут инсталляторы драйверов пятилетней давности, игры, которые ты удалил и забыл, и утилиты, которые ты запускал «только один разок».

Резюме главы:

UserAssist — это счетчик грехов. Он превращает версию «оно само» в пыль.

Пока пользователь думает, что он самый умный, потому что удалил ярлык с рабочего стола, ROT13 в реестре тихо хранит всю хронологию его падения.

И наша работа — просто расшифровать эту историю.

В следующей главе мы спустимся еще ниже. Мы поговорим о том, как доказать подключение устройства, которого физически уже нет в комнате.

— USBSTOR: История подключений флешек. Как найти ту самую флешку, которую «никто никогда не вставлял»

Есть три вещи, на которые можно смотреть бесконечно: как горит огонь, как течет вода и как подозреваемый бледнеет, когда ты называешь ему серийный номер флешки, которую он «потерял» полгода назад.

В мире корпоративных расследований 80% утечек информации происходит старым дедовским способом: сотрудник вставляет флешку, копирует файлы и уносит их домой. Никаких тебе сложных хакерских схем через сервис на три буквы или Tor. Просто физический доступ.

И сотрудник уверен в своей безнаказанности. Он думает: *«Я выдернул флешку, унес её, утопил в реке. Доказательств нет»*.

О, как же он ошибается.

Windows любит USB-устройства. Она их обожает. Как только ты вставляешь любой накопитель (флешку, внешний жесткий диск, даже телефон в режиме передачи данных), операционная система начинает бурную деятельность по его опознанию и установке драйверов.

И, как истинный бюрократ, она записывает всё: марку, модель, уникальный серийный номер и время первого и последнего подключения.

Эти записи хранятся в ветке реестра **USBSTOR**. И это — твой главный козырь против инсайдеров.

1. Механика предательства: Что происходит при подключении?

Когда флешка касается порта USB, происходит магия Plug and Play (PnP).

— **Опрос устройства:** Контроллер USB спрашивает: *«Кто ты, воин?»*.

— **Ответ:** Флешка радостно сообщает: *«Я Kingston DataTraveler 3.0, мой серийный номер 123456789»*.

— **Запись в реестр:** Windows создает под эту флешку уникальную ветку в реестре, чтобы в следующий раз узнать её и не устанавливать драйвер заново.

— **Монтирование:** Флешке присваивается буква (E:, F: и т.д.) и создается ссылка в системе.

Весь этот процесс оставляет неизгладимые следы в **HKEY_LOCAL_MACHINE \SYSTEM**.

Это не зависит от желания пользователя. Это работает на уровне ядра. Отключить это — значит сломать работу USB в принципе.

2. Триада Артефактов: Где искать следы?

Чтобы построить железобетонное доказательство, нам нужно заглянуть в три ключевые ветки реестра (System Hive).

А. USBSTOR — Паспортный стол

Путь: HKLM\SYSTEM\CurrentControlSet\Enum\USBSTOR

Здесь лежит список всех накопителей, которые КОГДА-ЛИБО подключались к этому компьютеру.

Структура проста:

— **Vendor & Product:** Например, Disk&Ven_Kingston&Prod_DataTraveler_3.0.

— **Instance ID (Серийный номер):** Внутри папки вендора будет папка с уникальным именем, например 50E549C688C3BE7189940366&0.

Важный момент для параноика:

Символ &0 в конце серийника означает, что это **Уникальный** серийный номер, прошитый на заводе.

Если ты видишь &1 или другой символ — значит, у устройства нет уникального номера (часто бывает у дешевых китайских ноунейм-флешек), и Windows сгенерировала его сама на основе PnP-свойств.

В суде уникальный номер (&0) — это царь-улика. Он привязывает конкретное физическое устройство к конкретному компьютеру.

Б. MountedDevices — Карта местности

Путь: HKLM\SYSTEM\MountedDevices

Здесь хранится связь между «железом» и буквой диска.

Конец ознакомительного фрагмента.