Security Engineering: часть 12. Сетевые атаки и защита, а также Авторское право и DRM

Published: 25.06.2026 11:21 | Updated: 26.06.2026 12:02

Сетевые атаки и защита (Уязвимости сетевых протоколов)

Современные атаки неразрывно связаны с сетевой связностью. Чтобы понять масштаб угрозы, достаточно рассмотреть несколько типичных сценариев, разворачивающихся в корпоративных и домашних сетях:

1. Офисный работник открывает вложение в электронном письме. Это заражает его компьютер вредоносным ПО, которое перехватывает пароли, передающиеся по локальной сети в открытом виде, и заражает другие машины в офисе.
2. Вредоносное письмо приходит от «мамы» пользователя. Вирус автоматически рассылает себя по всем адресам из адресной книги зараженной машины.
3. Злоумышленники не пытаются угадать пароль для конкретной учетной записи. Вместо этого они пробуют один и тот же пароль для миллионов аккаунтов (например, веб-почты), чтобы получить контроль над ними и рассылать спам.
4. «Google-хакинг»: злоумышленники используют поисковые системы для обнаружения веб-серверов с уязвимыми приложениями.
5. Вредоносное ПО массово заражает компьютеры. Затем оно ищет ценные данные (например, номера кредитных карт) или веб-серверы для хостинга фишинговых страниц. Остальные зараженные машины продаются по доллару за штуку операторам ботнетов, которые сдают их в аренду спамерам, фишерам и вымогателям.

6. Технология Fast-flux (быстрый поток): IP-адрес вредоносного веб-сайта меняется, например, каждые 20 минут, чтобы его было сложнее заблокировать. Разные машины в ботнете по очереди выступают хостами (или прокси), поэтому блокировка одного адреса дает лишь временный эффект. Это часто используется для фишинговых сайтов.

Уязвимости сетевых протоколов

Фундаментальные сетевые протоколы (IP, TCP, DNS, ARP, DHCP) разрабатывались в эпоху, когда сеть состояла из доверенных хостов, и вопросы безопасности не стояли на первом месте. Эта изначальная открытость заложила основу для множества векторов атак.

Атаки на локальные сети (LAN)

Если злоумышленник получает контроль хотя бы над одним ПК в локальной сети, его возможности резко расширяются:

- **Снифферинг и спуфинг.** Атакующий может установить сниффер пакетов для перехвата паролей, передающихся по сети в открытом виде, чтобы получить права администратора (root). Кроме того, он может подделать MAC- и IP-адреса (спуфинг), чтобы обойти сетевую защиту или выдать себя за другую машину.
- **Уязвимости файловых систем.** Классический пример — Network File System (NFS). NFS позволяет клиентам монтировать удаленные файловые системы. Если клиентская машина скомпрометирована, а сервер доверяет клиенту без должной проверки подлинности, злоумышленник может получить несанкционированный доступ к файлам на сервере.

Атаки с использованием интернет-протоколов и механизмов

SYN-флуд (SYN Flooding)

Протокол TCP использует трехстороннее рукопожатие для установки соединения. При SYN-флуде атакующий отправляет целевому серверу огромное количество SYN-пакетов, но намеренно игнорирует ответы сервера (SYN-ACK). Сервер выделяет ресурсы на ожидание подтверждения (создавая полуоткрытые соединения) и в итоге исчерпывает доступную память или буферы, что приводит к полному отказу в обслуживании (DoS).
Эта атака была известна теоретически с 1980-х годов, но широкую огласку

получила в 1996 году, когда с ее помощью на несколько дней был выведен из строя нью-йоркский провайдер Panix.
Контрмеры: Использование механизма «SYN-печенья» (syncookies). При этом сервер не хранит состояние полуоткрытых соединений, а вместо этого отправляет клиенту зашифрованное значение (cookie). Если клиент легитимен, он вернет cookie, и сервер восстановит состояние соединения. Эта защита реализована в Linux и других системах, однако SYN-флуды до сих пор остаются одним из самых распространенных видов атак.

Смурфинг (Smurfing)

Атака использует протокол ICMP (эхо-запросы, ping). Атакующий формирует ICMP-пакет с поддельным исходным адресом (адресом жертвы) и отправляет его на широковещательный адрес сети-усилителя (smurf amplifier). Все хосты в этой сети получают запрос и отправляют ответ на адрес жертвы, многократно усиливая трафик и перегружая ее каналы.
Контрмеры: Проблема была решена на уровне стандартов (RFC 2644), запретивших маршрутизаторам по умолчанию пересылать широковещательные ICMP-пакеты. Администраторы сетей также должны настраивать свои маршрутизаторы на игнорирование таких запросов. К 2007 году смурфинг практически сошел на нет.

Распределенные атаки типа «отказ в обслуживании» (DDoS)

В отличие от локальных DoS-атак, DDoS использует распределенную сеть из тысяч скомпрометированных машин (ботнет). Атакующий управляет ботнетом и заставляет все машины одновременно отправлять мусорный трафик на цель. DDoS-атаки стали публично известны в 1996 году (атака на Panix), а в конце 1990-х годов хакеры-подростки использовали их для захвата IRC-серверов. Позже атаки приобрели коммерческий характер: хакеры атаковали сайты онлайн-букмекеров перед крупными спортивными событиями и требовали деньги за прекращение атак.
Контрмеры:

- Перенос сайтов на хостинги с высокой пропускной способностью (например, Akamai), чья распределенная инфраструктура способна поглощать большие объемы трафика.
- Использование специализированных «пакетомоечных» машин (packet-washing), которые фильтруют вредоносный трафик на высокой скорости.

- Экономический подход: букмекеры объединились, выработали политику коллективного отказа платить выкуп, и полиция (в частности, в России) арестовала банду вымогателей.

Спам

Спам тесно связан с DDoS, так как также опирается на инфраструктуру ботнетов.

- **Консолидация:** Если раньше спам рассылался миллионами мелких операторов, то к 2007 году рынок консолидировался. Объемы спама стали резко колебаться в зависимости от кампаний нескольких крупных группировок.
- **Обратный выброс (backscatter):** Когда спамер подделывает адрес отправителя, сообщения об ошибках доставки (bounce messages) идут на поддельный адрес, создавая дополнительный мусорный трафик для ничего не подозревающей жертвы.
- **Контрмеры на уровне провайдеров (ISP):** Провайдеры начинают отслеживать исходящий спам со своих сетей и помещать зараженные машины в «изолированные сады» (walled gardens). Из таких сетей пользователи могут получить доступ только к странице с антивирусным ПО, что лишает ботнет возможности рассылать спам.

Безопасность DNS и фарминг

DNS (Domain Name System) преобразует удобные доменные имена в IP-адреса. Уязвимости DNS позволяют перенаправлять пользователей на вредоносные сайты.

- **Отравление кэша:** Если атакующий может подделать ответ и отравить кэш DNS-сервера, он может заменить IP-адрес легитимного сайта на подконтрольный ему адрес.
- **Фарминг (Pharming):** Атака, при которой пользователи перенаправляются на поддельные (например, фишинговые) сайты даже при вводе правильного URL.
- **Drive-By Pharming:** Особый вид атаки на домашние роутеры. Злоумышленник заманивает пользователя на веб-страницу с JavaScript-кодом, который через уязвимости в роутере меняет настройки DNS-сервера роутера на подконтрольный атакующему сервер. После этого все устройства в домашней сети будут перенаправляться на фишинговые сайты.

- **Контрмеры:** Пользователям необходимо менять заводские пароли на роутерах и обновлять их прошивки. На более высоком уровне требуется внедрение криптографической защиты DNS (DNSSEC), хотя на момент описания этих технологий она еще не была массово развернута.

Сетевые атаки и защита (Вредоносный код)

Эксперты по компьютерной безопасности давно осведомлены об угрозе вредоносного кода. Терминология в этой области иногда вызывает споры, но общепринятые определения звучат следующим образом:

- **Троян (Троянский конь)** — программа, которая выполняет какие-либо вредоносные действия (например, перехватывает пароли), когда ничего не подозревающий пользователь запускает её.
- **Червь** — это самореплицирующаяся вредоносная программа.
- **Вирус** — это червь, который реплицируется, прикрепляясь к другим программам.
- **Руткит** — piece of software (часть ПО), которая после установки на машину тайно передает ее под удаленный контроль.

Руткиты могут использоваться для целенаправленных атак (например, правоохранительными органами для превращения ноутбуков подозреваемых в прослушивающие устройства) или для финансового мошенничества (оснащаясь кейлоггерами для кражи паролей). Главная особенность современных руткитов — скрытность: они пытаются спрятаться от операционной системы, чтобы их нельзя было обнаружить и удалить стандартными средствами.

Ранняя история вредоносного кода

Вредоносный код появился практически одновременно с самими компьютерами.

- **Начало 1960-х:** Первые программы-трояны появились на раннем компьютере EDSAC. Студенты писали компьютерные игры с «троянским» кодом внутри, который проверял, запущена ли программа с правами root, и если да — создавал дополнительную привилегированную учетную запись с известным паролем.

- **1970-е годы:** Крупные системы с разделением времени в университетах стали мишенью для различных хулиганских выходок с использованием троянов.
- **1978 год:** Джон Шох и Джон Хупп из Xerox PARC написали программу, которую назвали «червем». Она реплицировалась по сети в поисках незанятых процессоров, чтобы назначать им задачи.
- **1984 год:** Кен Томпсон опубликовал классическую статью «On Trusting Trust» («О доверии к доверию»), в которой показал, что даже если исходный код системы тщательно проверен и не содержит уязвимостей, ловушку (trapdoor) можно встроить в компилятор.
- **1984 год:** Фред Коэн в рамках своей дипломной работы провел серию экспериментов с различными операционными системами, доказав, что код может распространяться с одной машины на другую и даже пересекать границы в многоуровневых защищенных системах. Это вызвало тревогу и привело к появлению первых реальных вирусов «в дикой природе» примерно через три года. Почти все они были вирусами для ПК, так как DOS была доминирующей ОС. Вирусы распространялись, когда пользователи делились программами на дискетах или через доски объявлений (BBS).

Интернет-червь

Первый случай, привлекший внимание прессы и широкой общественности, произошел в ноябре 1988 года. Это был интернет-червь, написанный Робертом Моррисом-младшим. Программа использовала несколько уязвимостей для распространения с одной машины на другую:

1. **Общие уязвимости:** подбор из 432 распространенных паролей и эксплуатация файлов `.rhosts`.
2. **Специфические уязвимости:** дыры в `sendmail` и `fingerd`.

Червь предпринял шаги для маскировки: он называл себя sh (shell) и шифровал свои строки данных с помощью шифра Цезаря. Однако в его коде была ошибка: он не мог корректно определить, что машина уже заражена, и продолжал reinfect (повторно заражать) её. Это привело к тому, что машины начали генерировать огромный объем сетевого трафика, полностью забив Интернет. Несмотря на то, что Интернет (и его предшественник Arpanet) проектировался с высочайшим запасом устойчивости к атакам, включая стратегический ядерный

удар, программа, написанная студентом, смогла его парализовать. К счастью, последствия были устранены за день-два, и червь поразил в основном системы на базе Berkeley Unix и их производные.

Как работают вирусы и черви

Вредоносная программа обычно состоит из двух компонентов: **механизма репликации** и **полезной нагрузки (payload)**.
Механизм репликации:

- Червь просто копирует себя в другое место (например, взламывает другую систему или рассылает себя по адресам из адресной книги зараженной машины).
- Вирус (в эпоху DOS) добавлял себя в конец исполняемого файла и патчил его так, чтобы путь выполнения сначала переходил к коду вируса, а затем возвращался к оригинальной программе.
- Специфический трюк: если в системе был файл `ACCOUNTS.EXE`, вирус мог создать файл `ACCOUNTS.COM`. Поскольку DOS исполняла `.COM` файлы в приоритете, пользователь, пытаясь запустить программу, неосознанно запускал вирус.

Полезная нагрузка (Payload):
Обычно активируется по триггеру (например, по дате) и может выполнять одно или несколько вредоносных действий:

1. Избирательное или случайное изменение состояния защиты машины.
2. Изменение пользовательских данных (например, шифрование жесткого диска с требованием выкупа за расшифровку).
3. Блокировка сети (например, начало репликации с максимальной скоростью).
4. Выполнение злонамеренных задач (например, использование процессора для перебора ключей DES).
5. Заставление модема звонить на номера с повышенным тарифом для перевода денег телефонным мошенникам.
6. Установка шпионского ПО (spyware) или рекламного ПО (adware) для кражи данных или показа рекламы.
7. Установка руткита для превращения машины в часть ботнета.

История развития вредоносного ПО

В конце 1980-х и начале 1990-х годов ПК-вирусы стали настолько серьезной проблемой, что породили целую индустрию производителей антивирусного ПО. Многие надеялись, что переход на «настоящие» 32-битные операционные системы (такие как Windows) решит эту проблему, но этого не произошло.

- **Интерпретируемые языки:** Распространение интерпретируемых языков предоставило хакерам плодотворную почву. В конце 1990-х годов процветали вредоносные Java-апплеты, эксплуатирующие уязвимости в браузерах. К 2000 году основными переносчиками угроз стали макроязыки в продуктах вроде Microsoft Word. Макровирусы доминировали благодаря тому, что документы пересылались по электронной почте.
- **Червь «Love Bug» (2000 год):** Стал переломным моментом. Это был самораспространяющийся червь, который рассылал себя всем из адресной книги жертвы с темой письма «I love you».
- **«Вспышечные черви» (Flash worms):** Появились такие угрозы, как Code Red и Slammer, которые сканировали весь Интернет на наличие уязвимых машин и заражали их за считанные часы или минуты.
- **Переход к организованной преступности (~2004 год):** С этого момента мотивация сместилась от хулиганства и хвастовства к созданию организованной криминальной экономики. Целью авторов вредоносного ПО стало не создание «глючных» вирусов, которые быстро обнаруживаются, а скрытный набор машин для последующей продажи операторам ботнетов.
- **Доминирование троянов:** Трояны стали встречаться чаще, чем черви. Если червь рассылает себя по электронной почте, он привлекает внимание жертвы. Трояны же распространяются напрямую операторами ботнетов, которые используют спам или фишинг.
- **Ботнет Storm (2007 год):** Один из крупнейших ботнетов, насчитывающий около миллиона машин. Его операторы использовали профессиональный онлайн-маркетинг для распространения троянов (например, спам с предложениями скачать трекер футбольных матчей). Ботнет использовал P2P-архитектуру и сдавался в аренду для мошеннических схем с акциями (pump-and-dump) и продажи поддельных лекарств.

Контрмеры

Борьба с вредоносным кодом напоминает гонку вооружений.

1. **Сканеры и контрольные суммы:** Первые антивирусы искали известные сигнатуры или проверяли контрольные суммы файлов.
2. **Полиморфизм:** В ответ авторы вирусов начали применять полиморфизм. Вирус шифровал свой код простым шифром, а в начале файла оставлял небольшой заголовок с кодом расшифровки. При каждой репликации вирус менял ключ шифрования и модифицировал код расшифровки (заменяя инструкции на эквивалентные), чтобы обойти сигнатурный анализ.
3. **Современная реальность:** Антивирусное ПО становится все менее эффективным. Операторы крупных ботнетов используют автоматизированные инструменты для тестирования эксплойтов. Руткиты не только скрываются от ОС, но и получают «послепродажное обслуживание» от своих создателей: как только появляется инструмент для удаления конкретного руткита, его авторы быстро выпускают обновленную, невидимую для этого инструмента версию.

Защита от сетевых атак и топология сетей

Обеспечение безопасности в сетевой среде опирается на четыре основных инструмента: управление конфигурацией, фильтрация, обнаружение вторжений и шифрование. Рассмотрим каждый из них, а также влияние топологии сети на её уязвимость.

Управление конфигурацией и операционная безопасность

Большинство технических уязвимостей в корпоративных сетях эксплуатируются не в день их обнаружения, а спустя месяцы, когда патчи уже выпущены, но не установлены. Цикл управления уязвимостями стал критическим процессом.

- **Патчинг:** Существует напряжение между необходимостью быстрого закрытия уязвимостей и необходимостью их тщательного тестирования. Microsoft внедрила практику «Patch Tuesday» (выход обновлений во

второй вторник каждого месяца), чтобы корпоративные клиенты могли планировать тестирование и развертывание. Злоумышленники, в свою очередь, научились реверс-инжинирингу патчей для создания эксплойтов «нулевого дня» против систем, которые еще не были обновлены.
- **Сокращение поверхности атаки:** Стандартной практикой стало удаление всех ненужных сетевых сервисов и демонов с серверов. Чем меньше портов открыто, тем меньше векторов для атаки.
- **Операционная безопасность:** Люди остаются слабым звеном. Обучение пользователей не нажимать на подозрительные ссылки в электронной почте важно, но недостаточно, так как многим сотрудникам по роду деятельности приходится кликать по ссылкам. Операционная безопасность должна включать не только инструкции, но и механизмы, ограничивающие ущерб от успешной компрометации (например, запуск браузеров с ограниченными привилегиями).

Фильтрация: Файрволы и прокси

Фильтры анализируют потоки пакетов и отбрасывают или модифицируют нежелательный трафик. Они делятся на три основных класса:

1. Фильтрация пакетов (Packet Filtering)

Работает на сетевом уровне (IP). Маршрутизатор или файрвол проверяет заголовки пакетов (IP-адреса источника/назначения, номера портов).

- **Защита от спуфинга:** Файрволы могут блокировать исходящие пакеты с поддельными внутренними IP-адресами (защита от smurf-атак) и отбрасывать входящие пакеты из внешних сетей, если они претендуют на внутренние адреса.
- **Ограничения:** Простая фильтрация уязвима к фрагментации пакетов, когда злоумышленник разделяет пакет так, чтобы заголовок с запрещенным портом попал в один фрагмент, а полезные данные — в другой, обходя правила фильтрации.

2. Шлюзы цепей (Circuit Gateways)

Работают на транспортном уровне (TCP). Они устанавливают сессию, проверяя её легитимность, и затем пропускают трафик.

- **VPN:** Шлюзы часто используются для создания виртуальных частных сетей (VPN), где трафик между удаленными офисами шифруется и инкапсулируется, проходя через публичный Интернет как обычный поток данных.

3. Прикладные шлюзы и ретрансляторы (Application Relays)

Работают на уровне приложений. Они полностью понимают протокол (например, SMTP для почты или HTTP для веба) и могут фильтровать контент.

- **Почтовые фильтры:** Удаляют вложения с исполняемыми файлами или макросами.
- **Веб-прокси:** Блокируют загрузку вредоносного кода или доступ к определенным категориям сайтов.
- **Гонка вооружений:** Злоумышленники обходят такие фильтры, упаковывая исполняемые файлы в ZIP-архивы, а затем и в зашифрованные архивы с паролем, который передается в теле письма.

Ingress и Egress фильтрация

- **Ingress (входящий):** Традиционная защита периметра от внешних атак.
- **Egress (исходящий):** Относительно новая концепция, направленная на контроль трафика, покидающего сеть. Провайдеры начинают использовать egress-фильтрацию для обнаружения зараженных машин, рассылающих спам. Зараженные хосты помещаются в «изолированные сады» (walled gardens), откуда у них есть доступ только к порталу с антивирусом, что лишает ботнет возможности рассылать мусор.

Архитектура

Сложные сети используют многоуровневую архитектуру. Классический пример — **DMZ (демилитаризованная зона)**. Внешний файрвол отделяет DMZ (где стоят публичные серверы) от Интернета, а внутренний файрвол защищает корпоративную сеть от DMZ.
Пример: На авиабазе Хэнском (Hanscom Air Force Base) была внедрена сложная система файрволов, разделяющая сети с разным уровнем доверия. Это показало,

что даже при наличии четкой политики, интеграция различных систем и управление правилами требуют постоянного внимания, так как ошибки в конфигурации сводят на нет всю защиту.

Обнаружение вторжений (Intrusion Detection)

Системы обнаружения вторжений (IDS) мониторят сеть или хосты на предмет аномалий или известных сигнатур атак.

- **Обнаружение злоупотреблений (Misuse detection):** Ищет известные шаблоны (сигнатуры) атак, например, специфические последовательности байтов в сетевом трафике, соответствующие эксплойту.
- **Обнаружение аномалий (Anomaly detection):** Строит профиль нормального поведения системы или пользователя и сигнализирует об отклонениях.
- **Ограничения:** Главная проблема IDS — высокий уровень ложных срабатываний (false positives). Если система генерирует слишком много ложных тревог, администраторы перестают на них реагировать. Кроме того, IDS бесполезны против трафика, зашифрованного с помощью TLS или IPsec, так как они не могут проанализировать содержимое пакетов.

Шифрование

Шифрование защищает конфиденциальность и целостность данных, но не решает всех проблем (например, оно не защищает от фишинга или вредоносного ПО на конечных точках).

SSH (Secure Shell)

Разработанный Тату Йлоненом в 1995 году, SSH заменил собой небезопасные протоколы удаленного доступа (telnet, rsh). SSH обеспечивает шифрование трафика и аутентификацию. Однако его безопасность зависит от правильной настройки: использование слабых ключей или уязвимостей в реализации (например, недостаточная энтропия при генерации ключей) может привести к компрометации.

WiFi (IEEE 802.11)

- **WEP (Wired Equivalent Privacy):** Изначальный протокол шифрования WiFi. Оказался фатально уязвимым из-за слабостей в алгоритме RC4 и использовании коротких векторов инициализации (IV). Атака Флуэрера, Мантина и Шамира (Fluhrer, Mantin, Shamir) позволила взламывать WEP-ключи за считанные минуты, собирая трафик.
- **WPA / WPA2:** Были разработаны как временное (WPA) и постоянное (WPA2) решение. WPA2 использует AES в режиме счетчика с CBC-MAC (CCMP), что обеспечивает надежную защиту, если используется сложный пароль (pre-shared key) или корпоративная инфраструктура с RADIUS-сервером.

Bluetooth

Изначально проектировался для персональных сетей (PAN) с низким энергопотреблением.

- **Уязвимости:** Протоколы сопряжения (pairing) подвергались атакам. Исследователи (Шакед и Вул) показали, что пин-код Bluetooth можно подобрать, перехватывая трафик во время процесса сопряжения. Более новые версии спецификации (Secure Simple Pairing) используют эллиптические кривые для защиты от пассивного прослушивания, но атаки типа «человек посередине» (man-in-the-middle) все еще возможны, если устройства не имеют надежного способа верификации (например, сравнения цифрового кода на экранах).

HomePlug

Технология передачи данных по силовой электропроводке.

- **Безопасность:** Поскольку сигнал выходит за пределы квартиры (в общую электросеть), шифрование обязательно. HomePlug AV использует AES. Ключи управления вводятся пользователем вручную при настройке сети. Исследования показали, что пользователи часто пренебрегают настройкой шифрования, оставляя сеть открытой для соседей или злоумышленников, имеющих физический доступ к розетке.

IPsec

Набор протоколов для защиты IP-соединений на сетевом уровне. Поддерживает режимы туннелирования (для VPN) и транспорта (для защиты соединений).

- **IKE (Internet Key Exchange):** Протокол для безопасного обмена ключами. IPsec широко используется в корпоративных средах, но его сложность часто приводит к ошибкам в конфигурации, которые сводят на нет преимущества шифрования.

TLS (Transport Layer Security)

Наследник SSL, обеспечивает безопасность на транспортном уровне (обычно поверх TCP). Используется для защиты веб-трафика (HTTPS), электронной почты и других приложений.

- **Механизм:** Клиент и сервер согласовывают криптографические алгоритмы (ciphersuite), обмениваются сертификатами для аутентификации и генерируют общий секретный ключ сессии.
- **Уязвимости:** TLS защищает канал, но не конечные точки. Если сервер или клиент скомпрометирован, шифрование не поможет. Кроме того, внедрение TLS в веб-браузеры привело к тому, что пользователи привыкли игнорировать предупреждения о недействительных сертификатах, что используется фишерами.

PKI (Public Key Infrastructure)

Инфраструктура открытых ключей должна была решить проблему доверия в Интернете, связывая открытые ключи с реальными личностями или организациями через центры сертификации (CA).

- **Реальность:** Во время бума доткомов PKI рассматривалась как панацея, и компании оценивались в миллиарды долларов. Однако на практике PKI столкнулась с проблемами масштабируемости, управления отзывом сертификатов (CRL) и юзабилити.
- **Ограничения:** Большинство пользователей не понимают, как работают сертификаты. Коммерческие CA иногда выдавали сертификаты мошенникам из-за ошибок в процедуре проверки (due diligence). В результате PKI не стала универсальным решением для защиты электронной коммерции, оставшись нишевым инструментом для корпоративных сетей и специфических приложений.

Топология

Топология сети (способ соединения узлов) критически влияет на её устойчивость к атакам и распространению вредоносного кода.

- **Масштабно-свободные сети (Scale-free networks):** Исследования Альберта, Чонга и Барабаши (Albert, Jeong, Barabasi) показали, что многие реальные сети (включая Интернет и социальные сети) имеют степенное распределение связей. В таких сетях есть небольшое количество «хабов» (узлов с огромным количеством связей) и множество слабо связанных узлов.
- **Устойчивость к отказам и атакам:** Такие сети крайне устойчивы к случайным сбоям (отключение случайного узла не разрушит сеть). Однако они **критически уязвимы к целенаправленным атакам**. Если злоумышленник идентифицирует и выведет из строя ключевые хабы (например, магистральные маршрутизаторы или крупные автономные системы), сеть распадется на изолированные фрагменты.
- **Распространение вредоносного кода:** Топология также определяет скорость распространения червей. В социальных сетях (где узлы — это люди или их аккаунты) черви могут использовать доверие между узлами для быстрого распространения. Понимание топологии необходимо для разработки стратегий сдерживания эпидемий вредоносного кода (например, путем локальной фильтрации трафика на уровнях провайдеров).

Авторское право и DRM (Введение, защита ПО, книг, аудио и видео)

Введение

Авторское право и управление цифровыми правами (DRM) стали одними из самых острых и конфликтных вопросов цифровой эпохи. Как метко заметил Джон Перри Барлоу: «Дело не в цензуре, а в определяющей битве за цензуру в киберпространстве». В то же время Джон Гилмор предупреждал: «Радуйтесь, что ваш ПК небезопасен — это означает, что после покупки вы можете взломать его и установить любой софт, который захотите. То, что хотите ВЫ, а не то, что хотят Sony, Warner или AOL».
Конфликт между голливудской индустрией и компьютерной индустрией

перешел в правовую плоскость. Опасения индустрии музыки и кино по поводу потерь от цифрового копирования привели к лоббированию «удобных» для них законов, таких как Digital Millennium Copyright Act (DMCA) в США и серии директив по интеллектуальной собственности (IP) в Европе. Эти законы предоставили особую юридическую защиту механизмам DRM, что позволило использовать их не только для борьбы с пиратством, но и для самых разных целей — от блокировки фишинговых сайтов до запрета на перезаправку картриджей для принтеров.

Авторское право

История показывает, что любые попытки контролировать распространение информации встречают сопротивление. Долгое время индустрии пытались вводить налоги на чистые аудио- и видеокассеты, чтобы компенсировать правообладателям потери от любительского копирования. Однако технологический прогресс постоянно обходил эти ограничения.

Программное обеспечение

В ранние дни вычислительной техники защита авторских прав на ПО не была проблемой. Программное обеспечение распространялось бесплатно либо производителями оборудования (например, IBM), либо самими пользователями. IBM даже учредила схему, позволяющую пользователям обмениваться написанными ими программами, хотя большинство из них были слишком специфичными.
Ситуация изменилась с появлением миникомпьютеров в 1960-х годах. Стоимость ПО стала значительной, и компании начали продавать не просто «железо», а комплексные решения. Сначала это были уникальные системы, написанные под заказ, но со временем они превратились в тиражируемые «пакеты» (например, системы бухгалтерского учета).
Появилась необходимость защиты от копирования. Первым инструментом стал анализ «программных водяных знаков» (software birthmarks) — уникальных черт стиля кода, по которым можно было доказать, что конкурент украл исходный код.
С приходом эры микрокомпьютеров и ПК защита стала более агрессивной. Разработчики использовали:

- **Аппаратные ключи (донглы):** устройства, подключаемые к портам ПК, без которых программа не запускалась.

- **Поврежденные сектора на дискетах:** программа проверяла наличие данных в специально «битых» секторах, которые нельзя было скопировать стандартными утилитами.
- **Проверку конфигурации ПК:** привязка к уникальному оборудованию.

Применялись и психологические методы: вывод предупреждающих сообщений или использование «временных бомб» (хотя последние часто вызывали юридические споры).
В конечном итоге индустрия пришла к выводу, что борьба с каждым пользователем экономически нецелесообразна. Многие компании перешли к бизнес-моделям, где сам продукт отдается бесплатно или дешево, а деньги зарабатываются на сопутствующих услугах или расходных материалах (например, принтеры продаются дешево, а чернила стоят дорого). Также набрало силу движение открытого исходного кода (Open Source), которое доказало, что совместная разработка может быть эффективнее проприетарной.

Книги

История XIX века преподносит полезный урок для издателей. Тогда появились «циркулирующие библиотеки» (прокат книг), и издатели боялись, что это уничтожит продажи. Однако библиотеки лишь разожгли аппетит читателей: люди, прочитавшие книгу в прокате, часто хотели купить ее в свою коллекцию. В итоге продажи книг резко выросли.

Аудио

Музыканты всегда пытались контролировать свои произведения. Никколо Паганини хранил свои партитуры в секрете, чтобы их не украли.
В эпоху аудиокассет индустрия снова запаниковала. В ответ был внедрен SCMS (Serial Copy Management System) — система, добавляющая один бит, который запрещал копировать копию (то есть вы могли сделать копию с оригинала, но не с копии). Эта система работала плохо, но это не имело значения: продажи кассет и так были огромными.
Позже появился формат DAT (цифровая аудиокассета), а затем и MP3. Алгоритмы сжатия MP3 позволили легко передавать музыку по интернету, что привело к взрывному росту пиринговых (P2P) сетей и новой волне конфликта с правообладателями.

Видео и Pay-TV

С появлением видеокассет Голливуд снова был в ужасе, опасаясь, что домашнее копирование убьет кинотеатры. Однако, как и в случае с книгами и аудио, видеокассеты стали огромным рынком, и голливудские студии начали зарабатывать на продажах больше, чем на прокате.
Ситуация с платным телевидением (Pay-TV) была иной. Операторам спутникового и кабельного ТВ необходимо было ограничивать доступ к каналам только для подписавшихся пользователей. Для этого были разработаны системы **условного доступа (conditional access)**.
Типичная архитектура систем защиты видео: Система использует гибрид аналоговых и цифровых технологий. Видеосигнал передается в аналоговом виде, но управление доступом — цифровое. Архитектура включает три компонента:

1. **Служба управления подпиской:** шифрует видеопоток и внедряет в него сообщения управления правами (EMM — Entitlement Management Messages) и сообщения управления доступом (ECM — Entitlement Control Messages).
2. **Приставка (Set-top box):** принимает сигнал, расшифровывает видео.
3. **Смарт-карта:** персонализирует приставку. Она хранит криптографические ключи, интерпретирует ECM и предоставляет приставке «слова управления» (control words) для расшифровки текущего кадра.

Методы скремблирования видео

Ранние системы (например, Videocrypt) использовали аналоговые методы, такие как инвертирование сигнала или добавление помех. Позже перешли к цифровым методам, но с аналоговым выводом.
Одним из известных алгоритмов был **cut-and-rotate (вырезание и вращение)**. Строка видеосигнала разрезалась в определенном месте, и половины менялись местами. Точка разреза определялась криптографическим ключом.
Однако такие методы были уязвимы к атакам с использованием обработки сигналов: зная алгоритм, можно было восстановить исходное изображение, анализируя артефакты скремблирования.

Атаки на гибридные системы скремблирования

Пираты быстро нашли способы обхода защиты приставок и смарт-карт.

1. **Атака с перехватом журнала ключей (Key-log attack):** Поскольку слово управления (control word) было одинаковым для всех приставок, смотрящих один канал, пираты устанавливали устройство между смарт-картой и приставкой, записывали поток ключей и выкладывали его в интернет. Люди, записавшие зашифрованную передачу, могли скачать ключи и посмотреть видео позже.
2. **Криптоанализ:** Многие системы использовали регистры сдвига с линейной обратной связью (LFSR). Пираты находили «узкие каналы» (narrow pipes) в алгоритмах, что позволяло восстанавливать ключи.
3. **Психологические и юридические меры:** Операторам ТВ приходилось постоянно обновлять алгоритмы, чтобы пираты не успевали адаптироваться. Также применялось «дерево отзыва» (revocation tree) — бинарное дерево групповых ключей, позволяющее эффективно отзывать скомпрометированные карты, не рассылая миллионы индивидуальных сообщений.

Юридическая борьба также была жесткой. Известен случай Криса Кэри (Chris Cary), ирландского пирата, который массово производил клонированные карты. Его поймали не только с помощью технических средств, но и благодаря тщательному финансовому расследованию и работе undercover-агентов.

DVB (Digital Video Broadcasting)

Переход на полностью цифровое телевидение (стандарты DVB) потребовал новых подходов. В DVB используется Общий Алгоритм Скремблирования (Common Scrambling Algorithm).
Поскольку цифровой сигнал можно копировать без потерь, индустрия разработала систему **CPCM (Content Protection & Copy Management)**. Ее цель — контролировать, что можно делать с контентом после того, как он был расшифрован внутри дома (например, можно ли записать его на жесткий диск, можно ли скопировать на другой носитель). Это вызвало новые споры о том, насколько далеко должны заходить технологии DRM в контроле действий пользователя внутри его желанной сети.

Общие платформы DRM (WMRM, OMA, P2P-сети, защита полупроводниковой IP)

С переходом к эпохе интернета перед индустрией встала задача контроля продажи и распространения цифровых товаров напрямую потребителям. Изначально эти технологии разрабатывались для онлайн-подписок на газеты и научные журналы, но быстро стало ясно, что музыка и видео станут основными объектами защиты.
Базовая проблема заключается в том, что персональный компьютер — это универсальное устройство, способное в принципе скопировать любой файл и отправить его куда угодно. В отличие от аналогового копирования, цифровые копии идеальны, и с одного оригинала можно сделать миллионы копий. Усугубляется проблема тем, что с точки зрения правообладателя владелец ПК является «врагом». Музыкальная и киноиндустрии считали, что неограниченное копирование разрушит их бизнес, в то время как компьютерная индустрия утверждала, что DRM внутренне невозможна на универсальном компьютере, и предлагала менять бизнес-модели. Несмотря на то, что размер медиаиндустрии составлял лишь десятую часть компьютерной, у Голливуда было гораздо больше политического влияния в Конгрессе. В итоге Голливуд настоял на своем, и результатом стало появление ряда продуктов, объединенных термином «управление цифровыми правами» (DRM).

Windows Media Rights Management (WMRM)

Одной из первых массовых систем стала WMRM, встроенная в Windows Media Player (WMP), который поставлялся с каждой копией Windows. Архитектура системы строилась вокруг шифрования медиаобъекта ключом контента (content key). Затем пользователю продавалась «лицензия», которая состояла из этого ключа, зашифрованного с помощью уникального ключа пользователя, а также набора правил на «языке управления правами» (rights management language), определяющих, что пользователь может делать с контентом (например, сколько раз его можно прослушать или записать на CD).
Для работы системы плеер должен был пройти процесс «персонализации»: при первом использовании он генерировал пару открытого/закрытого ключей и

отправлял открытый ключ серверу лицензий. Это позволяло серверу шифровать ключи контента так, чтобы их мог расшифровать только данный конкретный плеер.

Другие онлайн-системы управления правами

Помимо решения от Microsoft, на рынке присутствовали и другие платформы:

- **Apple FairPlay:** Система, используемая в iTunes и iPod. Она также привязывала лицензии к учетным записям и устройствам пользователей, создавая закрытую экосистему.
- **OMA (Open Mobile Alliance) DRM:** Стандарт, разработанный для мобильных устройств. Версия 1.0 широко использовалась для защиты рингтонов и применяла механизм «forward lock» (запрет на пересылку). Версия 2.0 была более сложной и поддерживала музыку и видео, однако ее внедрению сопротивлялись операторы сотовой связи. Они не хотели, чтобы контент-провайдеры получали прямой доступ к их клиентам, минуя операторские биллинговые системы.

Peer-to-Peer (P2P) системы

Параллельно с развитием DRM расцветали пиринговые сети, которые стали основным способом распространения музыки в обход ограничений.
* Первая волна была представлена **Napster**, чья централизованная служба обмена музыкой была быстро закрыта благодаря судебным искам музыкальной индустрии.
* На смену ей пришли децентрализованные системы, такие как **Gnutella** и **Freenet**. Они заимствовали идеи из мира систем, устойчивых к цензуре, создавая сети без центрального узла, который можно было бы закрыть юридическим путем.
* Позднее доминирование захватили протоколы вроде **BitTorrent**, использующие механизмы «роя» (swarming) для эффективной загрузки файлов, что сделало борьбу с пиратством еще более сложной задачей для правообладателей.

Защита полупроводниковой интеллектуальной собственности (Semiconductor IP)

Отдельной и очень острой проблемой является защита дизайнов микросхем, лицензируемых для использования в полупроводниковых устройствах. Компании, такие как ARM, зарабатывают на проектировании процессоров и других компонентов, которые они продают фирмам, производящим заказные чипы (ASIC) или программируемые логические матрицы (FPGA).
Здесь возникают две основные угрозы:

1. **Сверхплановое производство (Overrun production):** Фабрика-производитель (foundry) может изготовить больше чипов, чем было заказано по лицензии, и продать излишки на «сером» рынке.
2. **Кража интеллектуальной собственности (IP theft):** Конкуренты могут попытаться скопировать или реверс-инжинирить сам дизайн чипа.

Для борьбы с этим применяются криптографические ключи, встраиваемые в оборудование. Однако индустрия сталкивается с дилеммой: как обеспечить жесткую защиту, сохранив при этом возможность отладки и тестирования устройств. Баланс между безопасностью и удобством разработки остается сложной инженерной задачей.

Авторское право и DRM (Скрытие информации, Политика и Контроль аксессуаров)

Скрытие информации

Интерес Голливуда к защите авторских прав совпал в середине 1990-х годов с интересом военных к незаметным средствам связи и обеспокоенностью общественности по поводу контроля над криптографией. Это привело к бурному развитию области, известной как **скрытие информации** (information hiding). Концептуальная модель, предложенная Густавом Симмонсом, предполагает наличие отправителя и получателя, которые хотят скрыть факт передачи

сообщения от надзирателя (warden). В отличие от шифрования, где секретность заключается в содержимом, здесь секретным является само наличие сообщения. Терминология в этой области включает:

- **Водяные знаки (watermarks)** — скрытые сообщения об авторских правах.
- **Отпечатки (fingerprints)** — скрытые серийные номера (например, для отслеживания утечек).
- **Покрывающий текст (cover-text)** — исходный объект (изображение, аудио).
- **Стего-текст (stego-text)** — объект со скрытым сообщением.
- **Ключ скрытия (stego-key)** — секретный параметр, используемый для внедрения и извлечения.

Водяные знаки и управление копированием

Консорциум DVD столкнулся с проблемой «аналоговой дыры» — возможностью записать цифровой сигнал на аналоговый носитель и скопировать его без потерь. Для решения этой проблемы была предложена система управления копированием. Идея заключалась в том, чтобы помечать контент как «никогда не копировать» или «копировать один раз».
Технически это реализовывалось так: выбирается случайный «билет» X, вычисляется его хэш, который внедряется в виде невидимого водяного знака в видеопоток. Совместимые устройства считывают хэш. Если стоит пометка «копировать один раз», устройство разрешает запись, но автоматически меняет маркер на новой копии на «копировать запрещено». Эта система используется в DVD-audio, а также в более сложных схемах, таких как SPDC в Blu-ray, где каждый декодированный поток получает уникальный отпечаток.

Общие методы скрытия информации

История скрытия информации уходит корнями в глубокую древность. Геродот описывал, как послания вытатуировывались на выбритой голове раба, а после отрастания волос становились невидимыми. В эпоху Возрождения Фрэнсис Бэкон

использовал микроскопические различия в шрифтах для передачи скрытых битов.
В современную цифровую эру применяются следующие методы:

1. **Модификация наименее значимых битов (LSB).** Простейший метод, при котором данные скрываются в младших битах пикселей или аудиоотсчетов. Он крайне неустойчив: младшие биты легко обнуляются при сжатии или фильтрации.
2. **Решётка Кардано и псевдослучайные позиции.** Биты сообщения внедряются не последовательно, а в случайно выбранные позиции, определяемые ключом. Это усложняет обнаружение, но все еще уязвимо к активному надзирателю.
3. **Широкополосное кодирование (Spread Spectrum).** Используется псевдослучайная последовательность для «размазывания» сигнала по всему спектру частот. Это делает водяной знак устойчивым к потере части данных (например, при сжатии с потерями).
4. **Перцептивная фильтрация.** Сигнал внедряется только в те части изображения или звука, которые наиболее «зашумлены» или наименее заметны для человеческого восприятия.
5. **Специфичные для среды методы.** Например, сдвиг строк текста при печати на долю миллиметра или добавление неслышимых эхо-сигналов в музыку.

Атаки на схемы защиты авторских прав

Как и любые криптографические системы, схемы скрытия информации подвержены атакам.

- **Атака через геометрические искажения (Stirmark).** Исследователи создали инструмент Stirmark, который вносит микроскопические искажения (растяжение, сдвиг, поворот) в изображение, имитируя процесс печати и последующего сканирования. Эта атака успешно разрушила большинство ранних алгоритмов водяных знаков, так как они не учитывали потерю синхронизации.
- **Атака через выбранное искажение.** Если злоумышленник знает алгоритм внедрения, он может намеренно исказить контент так, чтобы разрушить скрытый маркер, но оставить файл пригодным для использования.

- **Эволюция защиты.** Компания Digimarc, чьи алгоритмы были взломаны в цифровом виде, перенесла фокус на физическую защиту: их невидимые маркеры теперь используются в банкнотах и документах, а графические редакторы (например, Photoshop) блокируют обработку изображений с их водяными знаками.

Приложения схем защиты авторских прав

Помимо защиты контента, технологии скрытия информации находят применение в других сферах:

- **Контроль копировальной техники.** Цветные копировальные аппараты (например, Xerox) внедряют невидимые желтые точки (machine identification code), содержащие серийный номер и дату, что позволяет отследить источник поддельных документов или банкнот.
- **Медицина.** Встраивание данных пациента непосредственно в цифровые рентгеновские снимки, чтобы исключить риск разделения снимка и его истории болезни.
- **Автоматизация покупок.** Радиостанции могут передавать скрытые метаданные о треке, позволяя слушателю автоматически заказать понравившийся CD.

Политика

Внедрение DRM вызвало острые политические и юридические дебаты. Законы, такие как DMCA в США и директивы ЕС, предоставили правообладателям юридическую защиту механизмов DRM. Это привело к тому, что законы об авторском праве стали использоваться не только для борьбы с пиратством, но и для блокировки фишинга, запрета на перезаправку картриджей и контроля за конкуренцией.

Лобби интеллектуальной собственности

Современное «IP-лобби» зародилось не в Голливуде, а в фармацевтической индустрии. В 1970-х годах компания Pfizer и другие производители лекарств лоббировали продление действия патентов в развивающихся странах (Бразилия, Индия).
Позже к ним присоединились музыкальная и киноиндустрии, а также производители предметов роскоши. Эта коалиция добилась принятия международных соглашений (TRIPS в 1994 году и договоров ВОИС в 1996 году),

которые фактически навязали жесткие стандарты защиты ИС всему миру под угрозой торговых санкций.
Однако со временем общественность осознала угрозы, которые эти законы несут для свободы слова, библиотек и права на частную жизнь. Лобби потеряло часть своего влияния. Например, попытки во Франции принять закон, возлагающий ответственность за пиратство на интернет-провайдеров, вызвали массовое возмущение.
Кроме того, возникла проблема **сохранения цифрового наследия**. Если физические книги хранятся веками, то цифровые файлы требуют постоянных затрат на миграцию форматов. Поскольку законы запрещают обход DRM даже для архивирования, библиотеки сталкиваются с юридическими барьерами при сохранении культурного наследия.

Кто выигрывает?

В 2005 году экономист Хэл Вэриан предсказал, что усиление DRM принесет больше пользы производителям платформ (Apple, Microsoft), чем создателям контента (музыкальным лейблам).
Экономическая теория гласит, что в цепочке создания стоимости выигрывает та сторона, которая обладает большей концентрацией рынка и создает «эффект блокировки» (lock-in). Музыкальная индустрия оказалась распыленной, тогда как Apple смогла создать закрытую экосистему (iTunes + iPod), получив основную прибыль.
Голливуд, пытаясь избежать участи музыкальных лейблов, пытается сам стать «владельцем платформы», внедряя такие механизмы, как SPDC в Blu-ray, чтобы контролировать не просто контент, а сами устройства воспроизведения.

Контроль аксессуаров

Одним из самых спорных последствий DRM стало использование криптографии для **контроля аксессуаров** (accessory control). Производители начали встраивать чипы аутентификации в расходные материалы, чтобы блокировать использование сторонних или перезаправленных картриджей.

- **Принтеры.** Если принтер обнаруживает неоригинальный картридж, он может незаметно снизить качество печати до 300 dpi или полностью отказаться от работы. Также применяется региональное кодирование: чернила, купленные в США, не будут работать в принтере, купленном в Великобритании.

- **Игровые консоли и телефоны.** Компании (Sony, Motorola) используют криптографические чипы в батарейках и модулях памяти, чтобы предотвратить использование неоригинальных комплектующих.
- **Автомобили.** Автопроизводители (например, Volkswagen) рассматривают внедрение систем, при которых автомобиль будет аутентифицировать запчасти, не позволяя использовать дешевые аналоги.

Юридический и экономический аспект.
Попытки регулировать это на уровне экологических законов (например, директива WEEE в ЕС) столкнулись с лоббизмом производителей и не принесли результата. В США суды (например, дело Lexmark против Static Control) рассматривали вопросы того, не нарушает ли использование DMCA для блокировки конкурентов антимонопольное законодательство.
Экономист Хэл Вэриан отмечает, что если рынок конкурентен, то контроль аксессуаров просто перекладывает затраты на потребителя. Но если рынок монополизирован, такие практики становятся инструментом нечестной конкуренции и ущемления прав потребителей.

Итог

Техническая защита цифрового контента и управление правами — это сложнейшая задача на стыке технологий, экономики и политики.

- **Техническая сложность.** Универсальный компьютер по своей природе способен копировать любые биты. Создание DRM-систем, которые одновременно надежны, незаметны и устойчивы к атакам (таким как геометрические искажения или обратная разработка), требует колоссальных усилий.
- **Политическая сложность.** Законы об авторском праве все чаще используются не для защиты творцов, а для создания искусственных барьеров, контроля над потребителями и подавления конкуренции.
- **Экономический сдвиг.** Власть в цепочке создания стоимости смещается от создателей контента к владельцам платформ и устройств.

Для инженера по безопасности уроки из области DRM очевидны: сложность — главный враг. Системы, которые пытаются контролировать каждый аспект поведения пользователя, неизбежно порождают уязвимости и вызывают отторжение. Будущее защиты информации лежит не в тотальном контроле, а в поиске баланса между безопасностью, удобством и экономическими интересами всех участников.