Лекция 7: Технологии сетевой безопасности

Тема: Технологии безопасности и защита инфраструктуры

Технический университет Молдовы

Лектор: Максим Масютин

Введение

Здравствуйте. Сегодня мы рассмотрим технологии сетевой безопасности: техники атак, которые угрожают сетям и веб-приложениям, а также защитные инструменты и методы, которые противостоят им. Это обширная тема, и мы затронем техники сетевых атак, атаки на веб-приложения, межсетевые экраны, VPN (Virtual Private Network, виртуальная частная сеть), системы обнаружения вторжений, сегментацию сети и современные платформы обеспечения безопасности.

Позвольте обозначить контекст. На заре вычислительной техники безопасность была относительно простой. Компьютеры были изолированы, данные хранились локально, и основной задачей была физическая безопасность. Затем появились сети, и всё изменилось. Внезапно данные могли передаваться между системами, злоумышленники могли достигать целей из любой точки мира, а обеспечение безопасности стало экспоненциально более сложным.

Традиционный подход к сетевой безопасности представлял собой модель "замка и рва" — построить прочные стены вокруг сети, и всё внутри будет в безопасности. Эта модель служила нам десятилетиями, но в современных условиях она принципиально не работает. Облачные вычисления означают, что ваши данные больше не находятся за вашими стенами. Мобильная работа означает, что ваши пользователи находятся за пределами стен. А продвинутые злоумышленники регулярно преодолевают периметровую защиту.

Тем не менее сетевая безопасность остаётся критически важной. Сеть — это среда передачи данных, и именно при передаче данные наиболее уязвимы. Согласно отчёту IBM Cost of a Data Breach за 2025 год, атаки на сетевом уровне остаются одной из ведущих причин нарушений безопасности, со средней стоимостью инцидента в 4,8 миллиона долларов. Мы должны защищать наши сети, даже осознавая, что одной лишь сетевой безопасности недостаточно.

Часть 1: Основы межсетевых экранов

Начнём с межсетевых экранов — наиболее фундаментальной технологии сетевой безопасности. Межсетевой экран — это устройство сетевой безопасности, которое отслеживает входящий и исходящий сетевой трафик и решает, разрешить или заблокировать определённый трафик на основе правил безопасности.

Термин "firewall" (межсетевой экран) происходит из строительства — firewall — это стена, предназначенная для предотвращения распространения огня между секциями здания. В сетевых технологиях межсетевой экран предотвращает распространение несанкционированного трафика между сегментами сети.

Каждый межсетевой экран принимает решения на основе правил. Правило определяет условия (адрес источника, адрес назначения, порт, протокол) и действие (разрешить или запретить). Когда трафик поступает на межсетевой экран, он сравнивается с правилами. Если правило совпадает, выполняется соответствующее действие. Если ни одно правило не совпадает, применяется действие по умолчанию (обычно — запрет).

Управление правилами межсетевых экранов — непростая задача. Организации часто накапливают тысячи правил на протяжении многих лет. Правила могут быть избыточными, противоречивыми или более не нужными. Регулярный пересмотр и очистка правил необходимы как для безопасности, так и для производительности.

Эталонная модель OSI

Прежде чем рассматривать типы межсетевых экранов, полезно понять эталонную модель OSI (Open Systems Interconnection, взаимодействие открытых систем), которая определяет семь уровней сетевого взаимодействия. Каждый уровень выполняет определённую функцию, и технологии безопасности работают на различных уровнях:

Уровень 7 (прикладной) предоставляет сервисы конечному пользователю, такие как HTTP, FTP, SMTP и DNS. Уровень 6 (представления) отвечает за форматирование данных, шифрование и сжатие. Уровень 5 (сеансовый) управляет сеансами и соединениями между приложениями. Уровень 4 (транспортный) обеспечивает сквозную доставку с использованием TCP и UDP. Уровень 3 (сетевой) отвечает за маршрутизацию и IP-адресацию. Уровень 2 (канальный) управляет передачей кадров с использованием Ethernet, Wi-Fi и MAC-адресов (Media Access Control, адресов управления доступом к среде). Уровень 1 (физический) передаёт электрические или оптические сигналы по кабелям и радиоканалам.

Межсетевые экраны работают на различных уровнях, обеспечивая разные степени защиты. Фильтрация пакетов работает на уровнях 3 и 4, контроль состояний (Stateful Inspection) добавляет отслеживание соединений на уровне 4, а

межсетевые экраны уровня приложений проверяют трафик на уровне 7. Понимание того, на каком уровне работает та или иная технология, помогает объяснить, от чего она может и от чего не может защитить.

Структура IP-пакета

IP-пакет существует на уровне 3 (сетевой) модели OSI и является базовой единицей маршрутизации данных в интернете. Понимание его структуры помогает объяснить, как межсетевые экраны принимают решения о фильтрации.

Заголовок IPv4-пакета содержит: поле версии (IPv4 или IPv6), поля длины заголовка и общей длины, IP-адрес источника (32 бита), IP-адрес назначения (32 бита), время жизни (TTL, Time to Live), ограничивающее количество промежуточных узлов, через которые может пройти пакет, и поле протокола, указывающее протокол вышестоящего уровня (TCP имеет номер 6, UDP — 17, ICMP — 1).

Полезная нагрузка пакета содержит сегмент уровня 4 (транспортный), который включает заголовки TCP или UDP с номерами портов источника и назначения, за которыми следуют собственно данные приложения с уровня 7 (прикладной). Эта многоуровневая структура объясняет, почему межсетевой экран с фильтрацией пакетов на уровнях 3 и 4 может видеть IP-адреса и номера портов, но не может проверить содержимое приложения, находящееся в полезной нагрузке.

Межсетевые экраны с фильтрацией пакетов

Простейшим типом межсетевого экрана является межсетевой экран с фильтрацией пакетов. Этот тип работает на уровне 3 (сетевой) и уровне 4 (транспортный) модели OSI. Он проверяет каждый сетевой пакет по отдельности и принимает решения о разрешении/запрете на основе информации из заголовка — IP-адресов источника и назначения, портов и протокола.

Межсетевые экраны с фильтрацией пакетов работают быстро, поскольку проверяют только заголовки, а не содержимое. Они реализованы практически в каждом маршрутизаторе и операционной системе. Linux iptables и Windows Firewall являются межсетевыми экранами с фильтрацией пакетов.

Правила фильтрации пакетов обычно выражаются в виде списков управления доступом, или ACL (Access Control List). Правило может гласить: "разрешить TCP-трафик с любого источника к адресу назначения 192.168.1.10, порт 443" — это позволит HTTPS-трафику поступать на веб-сервер.

Ограничение фильтрации пакетов заключается в отсутствии контекста. Она не может понять взаимосвязь между пакетами. Она не может определить, является ли пакет частью установленного соединения или новым, потенциально вредоносным запросом. Она не может проверить содержимое трафика. Это делает

её уязвимой к различным атакам, которые фильтрация пакетов просто не может обнаружить.

Межсетевые экраны с контролем состояний пакетов

Межсетевой экран с контролем состояний пакетов (Stateful Packet Inspection, SPI), также называемый экраном с отслеживанием состояний, представляет собой межсетевой экран, который отслеживает полное состояние активных сетевых соединений и принимает решения о фильтрации на основе контекста трафика, а не только заголовков отдельных пакетов. Межсетевые экраны с контролем состояний пакетов устраняют ограничение контекста, отслеживая состояние сетевых соединений. Когда внутренний пользователь инициирует соединение с внешним сервером, межсетевой экран записывает это в свою таблицу состояний. Обратный трафик затем разрешается, поскольку он является частью установленного соединения.

Это значительно более безопасно, чем фильтрация пакетов. Без контроля состояний пришлось бы разрешать весь трафик на высоких портах (выше 1024), поскольку невозможно предсказать, какой порт сервер использует для ответа. С контролем состояний можно разрешить только трафик, являющийся частью соединений, инициированных вашими пользователями.

Межсетевые экраны с контролем состояний также понимают поведение протоколов. Они знают, что TCP-соединение начинается с пакета SYN, за которым следует SYN-ACK, а затем ACK. Пакеты, нарушающие эту последовательность, могут быть отброшены как недействительные.

Большинство корпоративных межсетевых экранов сегодня как минимум поддерживают контроль состояний. Дополнительные вычислительные затраты на современном оборудовании минимальны, а повышение безопасности существенно. Когда мы говорим о межсетевых экранах в корпоративном контексте, мы, как правило, имеем в виду экраны с контролем состояний как минимум.

Межсетевые экраны уровня приложений

Межсетевые экраны уровня приложений, иногда называемые прокси-экранами или прокси приложений, идут дальше, понимая протоколы приложений на уровне 7 (прикладной) модели OSI. Они не просто отслеживают состояние соединений — они понимают HTTP, FTP, SMTP и другие протоколы на уровне приложений.

Межсетевой экран уровня приложений выступает в роли посредника. Когда внутренний пользователь хочет получить доступ к веб-сайту, он подключается к прокси. Прокси проверяет запрос, убеждается в его соответствии протоколу HTTP, применяет политики фильтрации контента, а затем устанавливает отдельное соединение с внешним сервером от имени пользователя. Ответ следует тем же путём в обратном направлении.

Такая глубокая проверка предоставляет мощные возможности безопасности. Межсетевой экран может блокировать определённые URL-адреса, сканировать загрузки на наличие вредоносного ПО, проверять зашифрованный трафик (при наличии соответствующих сертификатов) и применять детальные политики относительно ресурсов, к которым пользователи могут получить доступ. Он может обнаруживать и блокировать атаки на уровне приложений, которые были бы пропущены при фильтрации пакетов и контроле состояний.

Недостатком является производительность. Глубокий анализ пакетов является вычислительно затратным. Прокси приложений должны анализировать и понимать сложные протоколы. Для высоконагруженных сетей межсетевым экранам уровня приложений требуются значительные аппаратные ресурсы.

Межсетевые экраны нового поколения (NGFW)

Межсетевые экраны нового поколения, или NGFW (Next-Generation Firewall), сочетают традиционные возможности межсетевых экранов с расширенными функциями, включая предотвращение вторжений, распознавание приложений и аналитику угроз. Термин был введён компанией Gartner, аналитической компанией в сфере информационных технологий, в 2009 году, и NGFW стали стандартом для корпоративной периметровой безопасности.

Ключевые возможности NGFW включают:

Идентификация и управление приложениями — NGFW могут идентифицировать приложения независимо от порта, протокола или шифрования. Они могут различать Facebook и Gmail, Teams и Zoom, и применять различные политики к каждому из них. Это необходимо в эпоху, когда многие приложения используют HTTP-порт 80 или HTTPS-порт 443.

Интегрированное предотвращение вторжений — вместо развёртывания отдельных устройств IPS, NGFW включают функциональность IPS. Они могут обнаруживать и блокировать попытки эксплуатации уязвимостей, сканирование на наличие уязвимостей и известные шаблоны атак.

Осведомлённость о пользователях — NGFW интегрируются со службами каталогов для привязки сетевого трафика к конкретным пользователям. Политики могут основываться на идентификации пользователя и членстве в группах, а не только на IP-адресах.

Инспекция SSL/TLS — выступая в роли посредника (при наличии соответствующих сертификатов), NGFW могут расшифровывать, проверять и повторно зашифровывать HTTPS-трафик. Это необходимо, поскольку более 90% веб-трафика в настоящее время зашифровано, а злоумышленники используют шифрование для сокрытия вредоносной активности.

Интеграция с аналитикой угроз — NGFW подписываются на потоки аналитики угроз, предоставляющие актуальную информацию об известных вредоносных IP-

адресах, доменах и хешах файлов. Трафик, соответствующий этим индикаторам, может быть заблокирован автоматически.

Ведущие производители NGFW включают Palo Alto Networks, Fortinet, Cisco и Check Point. Эти продукты стали сложными платформами, часто выступающими центральной точкой для применения политик сетевой безопасности.

Межсетевые экраны уровня веб-приложений (WAF)

Межсетевые экраны уровня веб-приложений — это специализированные межсетевые экраны, предназначенные для защиты веб-приложений. В отличие от сетевых межсетевых экранов, защищающих периметр сети, WAF (Web Application Firewall) располагаются перед веб-приложениями и проверяют HTTP/HTTPS-трафик на предмет атак, направленных на уровень приложений.

WAF защищают от уязвимостей из списка OWASP (Open Web Application Security Project, Открытый проект безопасности веб-приложений) Top 10 и аналогичных уязвимостей веб-приложений. OWASP Top 10 (редакция 2021 года) определяет наиболее критические риски безопасности веб-приложений: A01 Нарушение контроля доступа, A02 Криптографические сбои, A03 Инъекции (SQL, OC, LDAP), A04 Небезопасное проектирование, A05 Неправильная конфигурация безопасности, A06 Уязвимые и устаревшие компоненты, A07 Сбои идентификации и аутентификации, A08 Нарушения целостности программного обеспечения и данных, A09 Сбои журналирования и мониторинга безопасности, A10 Подделка запросов на стороне сервера (SSRF). WAF могут обнаруживать и блокировать SQL-инъекции, межсайтовый скриптинг (XSS), инъекции команд и другие атаки, которые традиционные межсетевые экраны не могут обнаружить.

WAF используют несколько методов обнаружения. Обнаружение на основе сигнатур сопоставляет трафик с известными шаблонами атак. Например, наличие "UNION SELECT" в параметре URL указывает на SQL-инъекцию. Обнаружение на основе аномалий изучает нормальное поведение приложения и выявляет отклонения. Модель позитивной безопасности (белый список) точно определяет, как выглядит допустимый трафик, и блокирует всё остальное.

Варианты развёртывания WAF включают аппаратные устройства, виртуальные устройства, облачные сервисы и встроенные модули. Облачные сервисы WAF, такие как Cloudflare, AWS WAF и Azure WAF, стали популярными, поскольку их легко развернуть, и они могут поглощать масштабные атаки до того, как те достигнут вашей инфраструктуры.

Настройка WAF имеет критическое значение и представляет значительную сложность. Стандартные правила WAF часто генерируют множество ложноположительных срабатываний — легитимный трафик, который выглядит подозрительным. Организации должны настраивать правила под свои конкретные приложения, что требует понимания как поведения приложения, так и техник атак.

Часть 2: Техники сетевых атак

Понимание того, как злоумышленники эксплуатируют сети, необходимо прежде, чем мы перейдём к изучению средств защиты. В этой части мы рассмотрим основные категории сетевых атак: как они работают, что делает их эффективными и какие индикаторы должны отслеживать защитники.

Отказ в обслуживании (DoS) и распределённый отказ в обслуживании (DDoS)

Атаки DoS и DDoS направлены на то, чтобы сделать сервисы недоступными путём исчерпания ресурсов. DoS исходит от одного источника; DDoS исходит от множества источников одновременно, как правило, с использованием ботнетов из тысяч или миллионов скомпрометированных устройств.

DDoS-атаки подразделяются на три категории в зависимости от целевого ресурса:

- **Объёмные атаки** заливают сеть массированным трафиком для насыщения пропускной способности. UDP-флуд и атаки с усилением являются объёмными. Измеряются в битах в секунду (бит/с).
- **Протокольные атаки** исчерпывают ресурсы сервера или сетевого оборудования, эксплуатируя механизмы протоколов на уровнях 3-4. SYN-флуд является протокольной атакой, поскольку исчерпывает таблицу состояний соединений, а не пропускную способность. Измеряются в пакетах в секунду (пакетов/с).
- **Атаки на уровне приложений** нацелены на ресурсы уровня 7, отправляя легитимно выглядящие запросы, потребляющие несоразмерное количество серверных ресурсов. HTTP-флуд и Slowloris являются атаками на уровне приложений. Измеряются в запросах в секунду (запросов/с).

Понимание категории атаки определяет соответствующую защиту: объёмные атаки требуют поглощения пропускной способности (CDN, центры очистки трафика), протокольные атаки требуют управления состояниями (SYN cookies, ограничения соединений), а атаки на уровне приложений требуют анализа запросов (WAF, ограничение частоты запросов).

Атаки SYN-флуд

SYN-флуд эксплуатирует трёхстороннее рукопожатие TCP. В нормальных условиях клиент отправляет пакет SYN, сервер отвечает SYN-ACK и выделяет память для полуоткрытого соединения, а клиент завершает рукопожатие пакетом ACK. При SYN-флуде злоумышленник отправляет тысячи пакетов SYN с поддельными адресами источника, но никогда не отправляет финальный ACK.

Таблица соединений сервера заполняется полуоткрытыми соединениями, потребляя память и препятствуя легитимным подключениям.

Средства защиты включают SYN cookies, которые позволяют не выделять ресурсы до завершения рукопожатия, и ограничение скорости приёма пакетов SYN от отдельных источников.

Атаки UDP-флуд

UDP-флуд отправляет огромные объёмы UDP-пакетов на случайные порты цели. Для каждого пакета цель должна проверить наличие прослушивающего приложения, не найти его и сгенерировать ICMP-ответ "Destination Unreachable" (порт назначения недоступен). Объём пакетов перегружает вычислительные мощности цели и пропускную способность сети.

Атаки с усилением

Атаки с усилением являются одними из наиболее опасных техник DDoS. Злоумышленник отправляет небольшие запросы к сторонним серверам, подставляя IP-адрес жертвы в качестве источника. Серверы отправляют объёмные ответы жертве, многократно умножая трафик атаки.

Распространённые протоколы усиления и их коэффициенты:

- **Усиление DNS**: до 54x (маленький запрос, объёмный ответ с DNSSEC)
- **Усиление NTP**: до 556x (команда monlist возвращает до 600 последних клиентов)
- **Усиление Memcached**: до 51 000x (наибольший известный коэффициент усиления)
- **Усиление SSDP**: до 30x (Simple Service Discovery Protocol)

DDoS-атака на GitHub в 2018 году достигла 1,35 Тбит/с с использованием усиления Memcached. К 2025 году зафиксированы атаки, превышающие 3 Тбит/с.

DDoS на уровне приложений (уровень 7)

DDoS-атаки на уровне приложений отправляют легитимно выглядящие запросы, которые непропорционально потребляют ресурсы сервера. HTTP-флуд отправляет валидные запросы GET или POST в массовом масштабе. Каждый запрос требует от сервера выполнения запросов к базе данных и генерации ответов. Поскольку трафик выглядит легитимным, его сложнее фильтровать, чем атаки с усилением.

Slowloris представляет собой специализированную атаку уровня 7, которая открывает множество соединений к веб-серверу и отправляет неполные HTTP-заголовки крайне медленно, удерживая соединения открытыми неопределённо долго. Пул соединений сервера исчерпывается без генерации значительного

объёма трафика, что затрудняет обнаружение через мониторинг пропускной способности.

Атака "человек посередине" (MitM)

В атаке "человек посередине" злоумышленник тайно перехватывает и потенциально модифицирует коммуникацию между двумя сторонами, которые считают, что общаются напрямую.

Подмена ARP (ARP Poisoning)

Протокол ARP (Address Resolution Protocol) сопоставляет IP-адреса с MAC-адресами (Media Access Control) в локальной сети. ARP не имеет механизма аутентификации, поэтому любое устройство может отправлять ложные ARP-ответы, выдавая себя за другое устройство.

При атаке подмены ARP:

1. Злоумышленник отправляет ложные ARP-ответы жертве, утверждая, что MAC-адрес злоумышленника соответствует IP-адресу шлюза
2. Злоумышленник отправляет аналогичные ответы шлюзу, утверждая, что MAC-адрес злоумышленника соответствует IP-адресу жертвы
3. Весь трафик между жертвой и шлюзом теперь проходит через злоумышленника
4. Злоумышленник пересылает трафик в обоих направлениях, перехватывая или модифицируя его

Эта атака работает только в локальных сетях (в пределах одного широковещательного домена). Средства защиты включают Dynamic ARP Inspection (DAI) на управляемых коммутаторах, статические ARP-записи для критически важных устройств и мониторинг сети на предмет аномалий ARP.

Подмена DNS и отравление кэша DNS

Подмена DNS предоставляет ложные DNS-ответы для перенаправления жертв на серверы, контролируемые злоумышленником. Отравление кэша DNS повреждает кэш DNS-резолвера, заставляя его возвращать вредоносные IP-адреса для легитимных доменных имён.

При атаке отравления кэша:

1. Злоумышленник отправляет запрос целевому резолверу для определённого домена
2. До получения ответа от легитимного авторитативного сервера злоумышленник направляет резолверу поток поддельных ответов с вредоносными сопоставлениями

3. Если поддельный ответ принят, резолвер кэширует ложное сопоставление
4. Все последующие запросы к этому домену от любого пользователя резолвера перенаправляются на сервер злоумышленника

DNSSEC (DNS Security Extensions) гарантирует аутентификацию DNS-ответов с помощью цифровых подписей, предотвращая отравление кэша. Однако развёртывание DNSSEC в интернете остаётся неполным.

Атака SSL/TLS Stripping

SSL stripping понижает соединения HTTPS до HTTP. Злоумышленник, расположившийся в позиции "человека посередине" (через подмену ARP или аналогичную технику), перехватывает HTTP-запрос жертвы к веб-сайту. Когда веб-сайт перенаправляет на HTTPS, злоумышленник поддерживает HTTPS-соединение с сервером, но отправляет контент жертве по незашифрованному HTTP. Жертва видит HTTP в своём браузере, но может не заметить разницу. Злоумышленник может читать и модифицировать весь трафик в открытом виде.

HSTS (HTTP Strict Transport Security) защищает от SSL stripping, инструктируя браузеры всегда использовать HTTPS для данного домена. Однако HSTS вступает в силу только после первого посещения, если только домен не включён в предзагруженный список HSTS браузера (preload list).

Сетевая разведка

Перед проведением целенаправленных атак противники картографируют сеть для выявления целей, сервисов и уязвимостей. Разведка является первым этапом в большинстве фреймворков описания атак, включая MITRE ATT&CK.

Сканирование портов

Сканирование портов зондирует целевые системы для обнаружения работающих сервисов. Распространённые техники включают:

- **TCP SYN-сканирование (полуоткрытое сканирование)**: Отправляет пакеты SYN на порты. Ответ SYN-ACK означает, что порт открыт; RST означает закрыт. Сканер никогда не завершает рукопожатие, что делает его более скрытным, чем полное подключение. Это тип сканирования по умолчанию в Nmap, наиболее распространённом сканере портов.
- **TCP connect-сканирование**: Выполняет полное TCP-рукопожатие. Более заметно, но работает без повышенных привилегий на сканирующей системе.
- **FIN/Xmas/Null-сканирование**: Отправляет пакеты с необычными комбинациями флагов TCP. Некоторые системы по-разному реагируют на открытые и закрытые порты, раскрывая состояние порта и обходя простые правила межсетевого экрана, которые блокируют только пакеты SYN.

- **UDP-сканирование**: Отправляет UDP-пакеты на порты. Отсутствие ответа может указывать на открытый или отфильтрованный порт; ICMP "port unreachable" указывает на закрытый. UDP-сканирование медленное, поскольку многие системы ограничивают скорость ICMP-ответов.

Перечисление сервисов и сбор баннеров

После идентификации открытых портов злоумышленники определяют, какие программы и какие версии работают. Многие сервисы отправляют идентификационные баннеры при подключении. Например, SSH-сервер может ответить "SSH-2.0-OpenSSH_8.9", раскрывая точную версию. Злоумышленники проверяют выявленные версии по базам данных уязвимостей (CVE, NVD) для нахождения известных эксплойтов.

Определение операционной системы (OS Fingerprinting)

Определение операционной системы анализирует тонкие различия в реализации сетевых протоколов разными системами. Различные операционные системы используют различные значения TTL по умолчанию (Linux: 64, Windows: 128), размеры TCP-окна и поведение флагов. Такие инструменты, как Nmap, могут идентифицировать операционную систему с высокой точностью, анализируя эти специфичные для реализации характеристики.

Перехват пакетов (Packet Sniffing)

Перехват пакетов захватывает сетевой трафик для анализа. В средах с общим доступом к среде передачи (беспроводные сети, сети на основе концентраторов) любое устройство может перехватить весь трафик. В коммутируемых сетях злоумышленники используют подмену ARP или получают доступ к SPAN-порту для перенаправления трафика через свою систему.

Незашифрованные протоколы раскрывают конфиденциальные данные для перехватчиков: HTTP раскрывает веб-контент и учётные данные, FTP раскрывает содержимое файлов и пароли, Telnet раскрывает все данные сеанса, включая учётные данные, а SMTP раскрывает содержимое электронной почты. Именно поэтому шифрование (TLS для веб-трафика, SSH вместо Telnet, SFTP вместо FTP) необходимо для всех сетевых коммуникаций.

Атаки на беспроводные сети

Беспроводные сети создают дополнительные поверхности атаки помимо угроз проводных сетей.

Атака "злой двойник": Злоумышленник создаёт мошенническую точку доступа с тем же SSID, что и у легитимной сети, часто с более мощным сигналом. Жертвы

подключаются к мошеннической точке доступа, направляя весь свой трафик через злоумышленника. Эта атака особенно эффективна в общественных Wi-Fi средах, таких как аэропорты, гостиницы и кафе.

Атака деаутентификации: Злоумышленник отправляет поддельные кадры деаутентификации для отключения клиентов от легитимных точек доступа. Протокол 802.11 допускает неаутентифицированные управляющие кадры, что делает эту атаку тривиальной для выполнения. Отключённые клиенты могут затем подключиться к "злому двойнику", либо атака служит как отказ в обслуживании. Поправка 802.11w (Management Frame Protection) смягчает эту проблему, требуя аутентификации управляющих кадров.

Атаки на WPA2: Хотя WPA2 (Wi-Fi Protected Access 2) в целом безопасен, уязвимость KRACK (Key Reinstallation Attack), обнаруженная в 2017 году, продемонстрировала, что четырёхстороннее рукопожатие можно манипулировать для переустановки уже использованных ключей, что позволяет расшифровывать трафик. Кроме того, перехваченные рукопожатия WPA2 могут быть подвергнуты офлайн-атакам по словарю при наличии слабых паролей.

Улучшения WPA3: WPA3 (Wi-Fi Protected Access 3), утверждённый в 2018 году, устраняет слабости WPA2. Он использует SAE (Simultaneous Authentication of Equals) вместо рукопожатия PSK, обеспечивая устойчивость к офлайн-атакам по словарю и прямую секретность даже при слабых паролях. WPA3 также требует защиты управляющих кадров (Protected Management Frames), предотвращая атаки деаутентификации.

Часть 3: Атаки на веб-приложения

Веб-приложения являются одной из наиболее часто атакуемых поверхностей. Согласно отчёту Verizon 2025 DBIR, атаки на веб-приложения участвуют примерно в 26% всех нарушений безопасности. Понимание этих атак необходимо для настройки WAF, проектирования безопасных приложений и проведения оценки безопасности. OWASP Top 10, который мы перечислили в Части 1, предоставляет стандартную классификацию рисков веб-приложений. Здесь мы рассмотрим наиболее критические техники атак подробно.

SQL-инъекция

SQL-инъекция возникает, когда вводимые злоумышленником данные включаются в SQL-запросы без надлежащей очистки, позволяя злоумышленнику манипулировать операциями базы данных. Соответствует OWASP A03 (Инъекции).

Внутриполосная SQL-инъекция

Внутриполосная инъекция возвращает результаты непосредственно в ответе приложения.

Инъекция на основе UNION использует SQL-оператор UNION для объединения результатов запроса злоумышленника с исходным запросом. Например, если запрос поиска товара выглядит так:

```
SELECT name, price FROM products WHERE id = [user_input]
```

Злоумышленник может подставить: `1 UNION SELECT username, password FROM users`

Приложение возвращает данные о товаре вместе с учётными данными пользователей из таблицы users.

Инъекция на основе ошибок извлекает данные через сообщения об ошибках базы данных. Формируя ввод, вызывающий специфические ошибки SQL, злоумышленники могут извлекать данные порциями из текста ошибки, возвращаемого в браузер.

Слепая SQL-инъекция

Когда приложение не отображает результаты запроса или подробные сообщения об ошибках, слепые техники извлекают данные косвенно.

Слепая инъекция на основе булевых выражений задаёт вопросы типа "истина/ложь". Злоумышленник отправляет запросы, которые условно изменяют вывод страницы. Например, ввод `1 AND SUBSTRING(password,1,1)='a'` отобразит нормальную страницу, если первый символ пароля равен 'a', и другую страницу в противном случае. Повторяя этот процесс для каждой позиции символа, злоумышленник извлекает всё значение.

Слепая инъекция на основе времени использует функции задержки базы данных. Злоумышленник отправляет запросы, которые вводят измеримую задержку, когда условие истинно. Например, запрос, вызывающий 5-секундную задержку при совпадении первого символа, позволяет извлекать данные, измеряя время ответа.

SQL-инъекция второго порядка

При инъекции второго порядка вредоносный ввод безопасно сохраняется в базе данных и срабатывает позднее, когда используется в другом запросе. Например, имя пользователя, содержащее SQL-синтаксис, может быть безопасно сохранено при регистрации с использованием параметризованных запросов, но вызвать инъекцию, когда административный отчёт, не использующий параметризацию, извлекает и использует это имя пользователя.

Защита от SQL-инъекций

Основным средством защиты являются **параметризованные запросы** (подготовленные выражения), которые разделяют SQL-код и данные. База данных обрабатывает параметры как литеральные значения, а не как исполняемый код, независимо от их содержимого. Дополнительные средства защиты включают валидацию ввода, хранимые процедуры с параметризованными вызовами, учётные записи базы данных с минимальными привилегиями, ограничивающие возможности скомпрометированного запроса, и WAF как дополнительный уровень обнаружения.

Межсайтовый скриптинг (XSS)

Атаки XSS внедряют вредоносные скрипты в веб-страницы, просматриваемые другими пользователями. Браузер жертвы выполняет скрипт злоумышленника, поскольку он кажется исходящим от доверенного веб-сайта. XSS соответствует части OWASP A03 (Инъекции).

Хранимый (постоянный) XSS

Вредоносный скрипт постоянно хранится на целевом сервере, обычно в поле базы данных, таком как комментарий, сообщение на форуме или поле профиля. Когда другие пользователи просматривают затронутую страницу, скрипт выполняется в их браузерах.

Пример: Злоумышленник публикует комментарий, содержащий тег скрипта, который отправляет сессионный cookie просматривающего пользователя на сервер, контролируемый злоумышленником. Каждый пользователь, просматривающий страницу с комментарием, лишается своей сессии.

Хранимый XSS является наиболее опасным типом, поскольку он затрагивает всех пользователей, просматривающих скомпрометированный контент, без необходимости каких-либо действий помимо посещения страницы.

Отражённый XSS

Вредоносный скрипт включается в запрос (обычно в параметр URL) и отражается в ответе без сохранения. Злоумышленник должен обманом заставить жертву перейти по сформированной ссылке.

Пример: Страница поиска, которая отображает "Результаты для: [запрос]" без кодирования. Злоумышленник создаёт ссылку, содержащую скрипт в параметре запроса, и распространяет её через электронную почту или социальные сети. Когда жертва переходит по ссылке, скрипт выполняется в её браузере в контексте доверенного сайта.

DOM-основанный XSS

DOM-основанный XSS происходит полностью в клиентском JavaScript без отправки вредоносных данных на сервер. Уязвимый JavaScript считывает данные, контролируемые злоумышленником (из фрагмента URL, параметра запроса или другого клиентского источника), и вставляет их в DOM страницы без надлежащей очистки.

Защита от XSS

Кодирование вывода (кодирование HTML-сущностей) предотвращает интерпретацию браузером пользовательских данных как исполняемого кода. Символы `<`, `>`, `"`, `'` и `&` заменяются их HTML-эквивалентами. **Content Security Policy (CSP)** ограничивает, какие скрипты могут выполняться на странице, блокируя встроенные скрипты и скрипты из неавторизованных источников. Флаг **HTTPOnly** для cookie предотвращает доступ JavaScript к сессионным cookie, ограничивая последствия успешных XSS-атак. Валидация ввода создаёт дополнительный уровень защиты, но не должна быть единственным средством защиты.

Подделка межсайтовых запросов (CSRF)

CSRF эксплуатирует автоматическое включение браузером cookie в каждый запрос к домену. Злоумышленник создаёт веб-страницу, которая отправляет запрос к целевому сайту, где жертва аутентифицирована. Поскольку браузер автоматически прикрепляет сессионный cookie жертвы, целевой сайт обрабатывает запрос так, как если бы его инициировала жертва.

Пример: Банковский сайт обрабатывает переводы средств через POST-запрос. Злоумышленник создаёт страницу со скрытой формой, которая автоматически отправляет запрос на перевод при загрузке. Если клиент банка, вошедший в систему, посещает страницу злоумышленника, перевод выполняется с аутентифицированной сессией жертвы.

Защита от CSRF: Токены анти-CSRF представляют собой уникальные непредсказуемые значения, встраиваемые в формы и проверяемые при отправке. Поскольку страница злоумышленника находится на другом источнике (origin), она не может прочитать токен со страниц целевого сайта. Атрибут cookie **SameSite** (со значениями Strict или Lax) инструктирует браузер не отправлять cookie при межсайтовых запросах. Проверка заголовков Origin или Referer обеспечивает дополнительную серверную проверку.

Подделка запросов на стороне сервера (SSRF)

SSRF (Server-Side Request Forgery, подделка запросов на стороне сервера) обманывает сервер, заставляя его выполнять HTTP-запросы к непредусмотренным адресатам. Если веб-приложение загружает ресурсы по URL, предоставленным пользователями (например, импорт аватара по URL, генерация PDF из веб-страницы или вебхуки), злоумышленник может подставить внутренние URL для доступа к сервисам, недоступным напрямую из интернета. SSRF соответствует OWASP A10.

Наиболее примечательной целью SSRF в облачных средах является сервис метаданных экземпляра. В AWS (Amazon Web Services) URL `http://169.254.169.254/latest/meta-data/` возвращает метаданные экземпляра, включая учётные данные IAM-роли. Утечка данных Capital One в 2019 году использовала уязвимость SSRF в неправильно настроенном WAF для доступа к метаданным AWS, кражи временных учётных данных и эксфильтрации более 100 миллионов записей клиентов.

Защита от SSRF: Валидируйте и очищайте предоставленные пользователем URL перед выполнением запросов. Блокируйте запросы к диапазонам частных IP-адресов (10.x, 172.16-31.x, 192.168.x, 169.254.x, 127.x) и к конечным точкам облачных метаданных. Используйте белые списки разрешённых доменов назначения, когда это возможно. В AWS развёртывайте IMDSv2 (Instance Metadata Service version 2, служба метаданных экземпляра версии 2), который требует токен сеанса, получаемый через PUT-запрос, что SSRF-атаки обычно не могут выполнить.

Внедрение команд ОС

Внедрение команд происходит, когда пользовательский ввод передаётся командам оболочки операционной системы без очистки. Если веб-приложение конструирует системные команды с использованием пользовательского ввода, злоумышленники могут добавлять дополнительные команды с помощью метасимволов оболочки (`;` , `|` , `&&` , обратные кавычки, `$()`).

Пример: Страница сетевой диагностики, которая выполняет ping на указанный пользователем хост, запуская команду `ping [user_input]` через оболочку. Злоумышленник подставляет `8.8.8.8; cat /etc/passwd` , заставляя сервер выполнить как ping, так и команду чтения файла.

Средства защиты включают полный отказ от использования команд оболочки с пользовательским вводом (использование нативных библиотек языка программирования), валидацию ввода по строгим белым спискам разрешённых символов и запуск приложений с минимальными привилегиями операционной системы для ограничения последствий успешного внедрения.

Обход каталогов (Path Traversal)

Атаки обхода каталогов получают доступ к файлам за пределами предусмотренного каталога путём манипуляции параметрами пути к файлу с помощью последовательностей `../`. Если веб-приложение обслуживает файлы на основе пользовательского ввода, например `/download?file=report.pdf`, злоумышленник может запросить `/download?file=../../../etc/passwd` для чтения системных файлов.

Средства защиты включают каноникализацию путей к файлам перед валидацией, проверку того, что разрешённый путь остаётся в пределах предусмотренного каталога, использование chroot-изоляции или контейнерной изоляции для ограничения доступа к файловой системе и отказ от прямого использования пользовательского ввода в файловых операциях.

Атаки, специфичные для API

Современные веб-архитектуры в значительной степени полагаются на API, которые подвержены паттернам атак, описанным в OWASP API Security Top 10 (2023):

Некорректная авторизация на уровне объектов (BOLA, Broken Object-Level Authorization): API, использующие предсказуемые идентификаторы объектов (последовательные числовые ID) без проверки авторизации запрашивающего для каждого конкретного объекта. Злоумышленник изменяет ID в вызове API со своей записи на запись другого пользователя и получает доступ к несанкционированным данным. BOLA является наиболее распространённой уязвимостью API и соответствует OWASP API1.

Некорректная аутентификация (API2): API со слабыми механизмами аутентификации, отсутствующей валидацией токенов или чрезмерным временем жизни токенов. Злоумышленники эксплуатируют такие недостатки, как приём неподписанных JWT (JSON Web Token), отсутствие проверки срока действия токена или использование слабых API-ключей.

Избыточное раскрытие данных (API3): API, которые возвращают больше полей данных, чем нужно клиентскому приложению, полагаясь на клиентский код для фильтрации отображения. Злоумышленники перехватывают необработанный ответ API и получают доступ к конфиденциальным полям (адреса электронной почты, внутренние идентификаторы, финансовые данные), которые пользовательский интерфейс скрывает.

Массовое присвоение (API6): API, которые автоматически привязывают все предоставленные клиентом JSON-поля к внутренним объектам данных без фильтрации. Злоумышленник добавляет непредусмотренные поля в запрос, такие как `"role": "admin"` или `"account_balance": 999999`, которые API обрабатывает из-за неизбирательной привязки данных.

Часть 4: Технологии VPN

Виртуальные частные сети расширяют частные сети через публичные сети, позволяя пользователям отправлять и получать данные так, как если бы они были напрямую подключены к частной сети. VPN обеспечивают конфиденциальность посредством шифрования и аутентификацию с помощью различных механизмов.

VPN могут использовать любой из диапазонов частных IP-адресов, определённых стандартом RFC 1918, для своих туннельных сетей: 10.0.0.0/8 (от 10.0.0.0 до 10.255.255.255, приблизительно 16 миллионов адресов), 172.16.0.0/12 (от 172.16.0.0 до 172.31.255.255, приблизительно 1 миллион адресов) и 192.168.0.0/16 (от 192.168.0.0 до 192.168.255.255, приблизительно 65 000 адресов). Некоторые VPN-провайдеры, такие как Tailscale, разработчик VPN-решений на основе WireGuard, также используют диапазон CGNAT (Carrier-Grade NAT) по RFC 6598: 100.64.0.0/10 (от 100.64.0.0 до 100.127.255.255). Другие зарезервированные диапазоны включают 169.254.0.0/16 для локальной автоконфигурации (link-local) и 127.0.0.0/8 для обратной петли (loopback).

VPN служат для нескольких целей. VPN удалённого доступа позволяют отдельным пользователям подключаться к корпоративным сетям из любого места с доступом в интернет. VPN типа "сеть-сеть" (site-to-site) соединяют целые сети, позволяя филиалам взаимодействовать так, как если бы они находились в одной локальной сети. Потребительские VPN обеспечивают конфиденциальность, направляя трафик через VPN-серверы, скрывая IP-адрес пользователя и зашифровывая трафик от наблюдателей.

IPSec VPN

IPSec (Internet Protocol Security) — это набор протоколов для защиты IP-коммуникаций путём аутентификации и шифрования каждого пакета. IPSec работает на уровне 3 (сетевой) модели OSI, что делает его прозрачным для приложений.

IPSec имеет два режима. Транспортный режим шифрует только полезную нагрузку каждого пакета, оставляя IP-заголовок видимым. Этот режим используется для связи между хостами. Туннельный режим шифрует весь исходный пакет и инкапсулирует его в новый пакет. Этот режим используется для VPN типа "сеть-сеть" и удалённого доступа.

Ключевые протоколы IPSec — это Authentication Header (AH), обеспечивающий аутентификацию и целостность, и Encapsulating Security Payload (ESP), обеспечивающий конфиденциальность, аутентификацию и целостность. На практике почти всегда используется ESP, поскольку конфиденциальность обычно необходима.

IPSec использует протокол Internet Key Exchange (IKE) для согласования ассоциаций безопасности и обмена ключами. IKEv2 является текущей версией, предлагающей улучшения в производительности, надёжности и поддержке мобильных устройств по сравнению с IKEv1.

IPSec VPN широко развёрнуты для соединения "сеть-сеть". Они хорошо подходят для постоянных соединений между фиксированными точками. Однако IPSec может быть сложен в настройке и диагностике и не всегда беспрепятственно проходит через устройства NAT и межсетевые экраны.

VPN на основе SSL/TLS

VPN на основе SSL/TLS используют те же протоколы шифрования, которые защищают веб-просмотр, для создания VPN-туннелей. Поскольку они используют стандартный HTTPS (порт 443), они легко проходят через межсетевые экраны и устройства NAT — любая сеть, допускающая веб-просмотр, допустит и VPN на основе SSL/TLS.

Существуют два типа VPN на основе SSL/TLS. Бесклиентский SSL VPN предоставляет доступ через веб-браузер без установки клиентской программы. Пользователи подключаются к веб-порталу, предоставляющему доступ к внутренним веб-приложениям, файловым хранилищам и другим ресурсам. Это удобно для временного доступа с недоверенных устройств.

Полноценный SSL/TLS VPN требует клиентской программы, но обеспечивает сетевой доступ, аналогичный IPSec VPN. Клиент создаёт виртуальный сетевой адаптер, и трафик туннелируется через зашифрованное соединение. Это даёт доступ к любому сетевому ресурсу, а не только к веб-приложениям.

Основные продукты SSL/TLS VPN включают Cisco AnyConnect, Palo Alto GlobalProtect, Fortinet FortiClient и решение с открытым исходным кодом OpenVPN. Эти решения в значительной степени заменили IPSec для удалённого доступа благодаря своей гибкости и простоте использования.

OpenVPN

OpenVPN — это VPN-решение с открытым исходным кодом, впервые выпущенное в 2001 году James Yonan (Джеймсом Йонаном), разработчиком ПО. Оно использует библиотеку OpenSSL для шифрования и построено на протоколах TLS (Transport Layer Security), что делает его тесно связанным с категорией VPN на основе SSL/TLS, рассмотренной выше.

OpenVPN поддерживает транспортные протоколы TCP и UDP. TCP гарантирует надёжную доставку ценой некоторой потери производительности, тогда как UDP обеспечивает лучшую производительность для большинства сценариев использования VPN. Возможность переключения на TCP через порт 443 означает,

что OpenVPN может работать в сетях, ограничивающих нестандартный трафик, поскольку он выглядит идентично обычному HTTPS.

OpenVPN может работать в двух режимах: режим TUN (виртуальный сетевой интерфейс для маршрутизируемого трафика), создающий виртуальный интерфейс уровня 3 (сетевой), и режим TAP (виртуальный сетевой интерфейс для мостового трафика), создающий виртуальный интерфейс уровня 2 (канальный). Режим TUN более распространён и эффективен для большинства сценариев; режим TAP используется, когда требуется мостовое соединение на уровне Ethernet, например, для работы устаревших протоколов, которым необходима связность на уровне 2.

OpenVPN доступен для Linux, Windows, macOS, Android и iOS. Он предлагает гибкие варианты аутентификации, включая сертификаты, предварительно распределённые ключи и интеграцию с LDAP или Active Directory. OpenVPN Access Server предоставляет коммерческое решение с веб-интерфейсом управления.

Важным событием стало появление DCO (Data Channel Offload, разгрузка канала данных) — модуля ядра, который переносит обработку плоскости данных из пользовательского пространства в ядро. Без DCO OpenVPN работает полностью в пользовательском пространстве, требуя копирования данных между ядром и пользовательским пространством для каждого пакета. DCO устраняет эти накладные расходы, обеспечивая увеличение пропускной способности в 2-3 раза. DCO был представлен в OpenVPN 2.6, выпущенном в январе 2023 года, и доступен для Linux (ovpn-dco) и Windows (ovpn-dco-win).

WireGuard

WireGuard — это современный протокол VPN, получивший значительное распространение после включения в ядро Linux. В частности, WireGuard был включён в ядро Linux версии 5.6, выпущенной 29 марта 2020 года. До этого выпуска WireGuard был доступен как внешний модуль ядра (out-of-tree), который необходимо было компилировать отдельно. Включение в основное ядро означало, что любая система Linux с ядром 5.6 или новее имеет встроенную поддержку WireGuard, что значительно упрощает развёртывание. WireGuard стремится быть проще, быстрее и безопаснее, чем IPSec и OpenVPN.

Простота WireGuard примечательна — вся кодовая база составляет около 4000 строк по сравнению с приблизительно 100 000 для OpenVPN и сотнями тысяч для реализаций IPSec. Такой малый объём упрощает аудит и снижает вероятность наличия уязвимостей.

WireGuard использует современные консервативные криптографические решения: Curve25519 для обмена ключами, ChaCha20 для шифрования, Poly1305 для аутентификации и BLAKE2s для хеширования. Эти алгоритмы были выбраны за их безопасность и производительность.

ChaCha20, потоковый шифр, разработанный Daniel J. Bernstein (Дэниелом Бернштейном), криптографом, был выбран вместо AES (Advanced Encryption Standard) по важной причине. AES достигает наилучшей производительности на процессорах с аппаратными инструкциями AES-NI (AES New Instructions), доступными на процессорах Intel x86-64 начиная с архитектуры Westmere (2010) и на процессорах AMD начиная с Bulldozer (2011). На процессорах с AES-NI алгоритм AES-256-GCM приблизительно в 2-5 раз быстрее, чем ChaCha20-Poly1305. Однако на устройствах без аппаратного ускорения AES-NI, таких как многие мобильные устройства на базе ARM, старые процессоры и устройства IoT, ChaCha20-Poly1305 приблизительно в 3 раза быстрее, чем AES. Поскольку WireGuard стремится к стабильной производительности на всех платформах без необходимости специализированного оборудования, ChaCha20 оказался лучшим выбором. И ChaCha20, и AES-256 обеспечивают эквивалентный уровень безопасности в 256 бит.

По производительности WireGuard значительно превосходит как IPSec, так и OpenVPN в большинстве тестов. Он обеспечивает более высокую пропускную способность при меньшей нагрузке на процессор, что делает его идеальным для мобильных устройств, где важен ресурс батареи.

Компромиссы WireGuard включают ограниченный набор функций (отсутствие встроенной аутентификации пользователей помимо ключей), менее зрелую экосистему и некоторые корпоративные функции, находящиеся на стадии разработки. Тем не менее распространение быстро растёт, и многие VPN-провайдеры и предприятия добавляют поддержку WireGuard.

OpenVPN с DCO в сравнении с WireGuard

Сравнение OpenVPN с DCO и WireGuard наглядно демонстрирует компромиссы между зрелостью и простотой. OpenVPN с DCO переносит свою плоскость данных в ядро, обеспечивая увеличение пропускной способности в 2-3 раза по сравнению со стандартным OpenVPN в пользовательском пространстве, при сохранении всей гибкости OpenVPN: переключение на TCP для ограничительных сетей, режим TAP для мостового соединения на уровне 2 (канальный) и расширенные варианты аутентификации. Тем не менее, даже с DCO, WireGuard по-прежнему превосходит OpenVPN в большинстве тестов благодаря своей изначально резидентной в ядре архитектуре и минимальной кодовой базе (приблизительно 4000 строк против приблизительно 100 000 для OpenVPN).

Выбор между ними зависит от требований. OpenVPN предпочтителен, когда первостепенное значение имеет гибкость: кроссплатформенная совместимость, переключение на TCP через межсетевые экраны, мостовое соединение на уровне 2 или интеграция с корпоративными системами аутентификации. WireGuard предпочтителен, когда важна максимальная производительность, особенно на мобильных устройствах и в средах с ограниченными ресурсами. Многие организации развёртывают оба решения, используя WireGuard для соединений

"сеть-сеть", где критична производительность, и OpenVPN для удалённого доступа, где более важны гибкость и совместимость.

Часть 5: Системы обнаружения и предотвращения вторжений

Системы обнаружения вторжений, или IDS (Intrusion Detection System), и системы предотвращения вторжений, или IPS (Intrusion Prevention System), отслеживают сетевой трафик на предмет вредоносной активности. Разница между ними заключается в реагировании: IDS обнаруживает и оповещает, тогда как IPS обнаруживает, оповещает и блокирует.

Методы обнаружения

Обнаружение на основе сигнатур сравнивает трафик с базой данных известных сигнатур атак. Когда трафик соответствует сигнатуре, генерируется оповещение (IDS) или трафик блокируется (IPS). Обнаружение на основе сигнатур эффективно против известных атак, но бессильно против новых, неизвестных атак.

Сигнатуры должны обновляться часто, чтобы оставаться эффективными. Производители выпускают новые сигнатуры ежедневно или даже чаще. Организации должны находить баланс между необходимостью актуальных сигнатур и риском того, что ошибочная сигнатура вызовет ложноположительные срабатывания или проблемы с производительностью.

Обнаружение на основе аномалий устанавливает базовый уровень нормального сетевого поведения и оповещает, когда трафик отклоняется от этого базового уровня. Это позволяет обнаруживать новые атаки, не соответствующие ни одной сигнатуре. Однако обнаружение аномалий требует настройки под конкретную среду и генерирует больше ложноположительных срабатываний, чем обнаружение на основе сигнатур.

Анализ протоколов проверяет соответствие трафика спецификациям протоколов. Например, HTTP-трафик должен соответствовать правилам протокола HTTP. Трафик, нарушающий эти правила, может быть вредоносным — злоумышленники часто формируют некорректный трафик для эксплуатации уязвимостей парсеров.

Поведенческий анализ рассматривает закономерности активности во времени. Единичная неудачная попытка входа может быть нормальной; тысячи неудачных попыток входа с одного источника — это атака методом перебора. Поведенческий анализ может обнаруживать медленные, скрытные атаки, которые были бы невидимы для сигнатурного и протокольного анализа.

Развёртывание IDS/IPS

Сетевые IDS/IPS, также обозначаемые как NIDS/NIPS (Network IDS/Network IPS), отслеживают трафик в узловых точках сети. Сенсор устанавливается там, где он может видеть трафик — часто подключённый к SPAN-порту коммутатора или сетевому отводу. Сенсор анализирует трафик и отправляет оповещения на консоль управления.

Для блокировки трафика IPS должна быть развёрнута в режиме inline — трафик должен проходить через IPS. Это вносит задержку и создаёт единую точку отказа. Если IPS выходит из строя, сетевое соединение может быть потеряно. Конфигурации высокой доступности с переключением на резерв необходимы для рабочих развёртываний.

Хостовые IDS/IPS, или HIDS/HIPS (Host-based IDS/Host-based IPS), работают на отдельных конечных точках и отслеживают локальную активность — изменения файлов, выполнение процессов, сетевые соединения с данного хоста. HIDS даёт видимость, недоступную сетевым IDS, например, содержимое зашифрованного трафика и локальную файловую активность.

Ведущие продукты IDS/IPS включают Snort (открытый исходный код), Suricata (открытый исходный код), Cisco Firepower и Palo Alto Threat Prevention. Многие NGFW включают интегрированную функциональность IPS, что уменьшает потребность в отдельных устройствах IPS.

Проблемы IDS/IPS

Ложноположительные срабатывания являются постоянной проблемой. Команды безопасности могут получать тысячи оповещений в день, большинство из которых являются ложноположительными. Усталость от оповещений приводит к тому, что аналитики начинают их игнорировать, потенциально пропуская реальные атаки. Настройка правил IDS/IPS под конкретную среду необходима, но требует значительного времени.

Зашифрованный трафик становится всё более серьёзной проблемой. Когда трафик зашифрован с помощью TLS, IDS/IPS не может увидеть его содержимое. Злоумышленники знают об этом и регулярно используют шифрование для уклонения от обнаружения. Инспекция SSL/TLS может помочь, но вызывает опасения относительно конфиденциальности и добавляет сложность.

Техники уклонения позволяют злоумышленникам формировать трафик, обходящий обнаружение. Фрагментация разбивает атаки на несколько пакетов. Кодирование преобразует строки атак для избежания совпадения с сигнатурами. Уклонение на уровне протокола использует различия в интерпретации трафика между IDS и целевыми системами. Поддержание IDS/IPS в актуальном состоянии и правильная настройка необходимы для противодействия уклонению.

Часть 6: Сегментация сети

Сегментация сети разделяет сеть на несколько сегментов или подсетей, каждая из которых выступает как отдельная небольшая сеть. Это повышает безопасность, ограничивая радиус поражения при нарушениях безопасности и обеспечивая управление доступом между сегментами.

В плоской сети без сегментации злоумышленник, скомпрометировавший любую систему, потенциально может получить доступ ко всем остальным системам. При сегментации компрометация рабочей станции в финансовом отделе не даёт автоматического доступа к серверам в центре обработки данных.

Подходы к сегментации

Сегментация на основе VLAN (Virtual Local Area Network, виртуальная локальная сеть) использует виртуальные локальные сети для создания отдельных широковещательных доменов. Устройства в разных VLAN не могут взаимодействовать напрямую — трафик должен проходить через маршрутизатор или межсетевой экран, который может применять политики доступа. VLAN легко реализуются и поддерживаются всеми управляемыми коммутаторами.

Ограничение сегментации на основе VLAN заключается в относительно крупной гранулярности. Можно разделить департаменты или функции, но не отдельные рабочие нагрузки. VLAN также плохо охватывают несколько центров обработки данных, что является проблемой в распределённых средах.

Сегментация с помощью межсетевых экранов размещает межсетевые экраны между сегментами сети. Весь трафик между сегментами проходит через межсетевой экран, который применяет политики доступа. Это обеспечивает детальный контроль, но добавляет задержку, стоимость и сложность.

Микросегментация идёт дальше, применяя средства безопасности на уровне отдельных рабочих нагрузок. Вместо сегментации сетей микросегментация сегментирует приложения. Каждый сервер или контейнер имеет собственную политику безопасности, и трафик контролируется даже между системами в одном сегменте.

Программно-определяемая микросегментация, предлагаемая такими производителями, как VMware NSX, Cisco Tetration (теперь часть Cisco Secure Workload) и Illumio, разработчик решений микросегментации, реализует микросегментацию без необходимости изменений физической сети. Политики определяются централизованно и применяются программными агентами на каждой рабочей нагрузке.

Архитектура DMZ

Демилитаризованная зона, или DMZ (Demilitarized Zone), — это сегмент сети, расположенный между доверенной внутренней сетью и недоверенной внешней сетью (как правило, интернетом). Системы, которые должны быть доступны из интернета, размещаются в DMZ.

Классическая архитектура DMZ использует два межсетевых экрана — внешний межсетевой экран, обращённый к интернету, и внутренний межсетевой экран, обращённый к внутренней сети. DMZ располагается между ними. Трафик из интернета может достигать систем в DMZ, но не внутренних систем. Внутренние системы могут обращаться как к DMZ, так и к интернету, но системы DMZ имеют ограниченный доступ к внутренним ресурсам.

Системы в DMZ обычно включают веб-серверы, почтовые шлюзы, DNS-серверы и VPN-концентраторы — всё, что требует внешнего подключения. Изолируя эти системы, вы ограничиваете ущерб в случае их компрометации. Злоумышленник, захвативший веб-сервер в DMZ, по-прежнему должен преодолеть внутренний межсетевой экран для доступа к критически важным системам.

Современные вариации включают многоуровневые DMZ с различными уровнями доверия, DMZ для конкретных приложений и облачные эквиваленты DMZ. Принцип остаётся прежним: предоставлять недоверенным сетям как можно меньше доступа и изолировать то, что должно быть доступно.

Сеть с нулевым доверием

Сеть с нулевым доверием применяет принцип "никогда не доверяй, всегда проверяй" к проектированию сети. Традиционная сетевая безопасность предполагает, что внутренний трафик является доверенным — если вы находитесь за межсетевым экраном, вы можете свободно взаимодействовать. Нулевое доверие предполагает, что никакой трафик не является доверенным, независимо от местоположения.

В сети с нулевым доверием каждое соединение аутентифицируется и авторизуется. Рабочая станция, подключающаяся к серверу, должна подтвердить свою идентификацию, даже если обе находятся в одном сегменте сети. Политики основываются на идентификации и контексте, а не на местоположении в сети.

Сеть с нулевым доверием обычно включает:

Прокси с проверкой идентификации, которые аутентифицируют пользователей и устройства перед предоставлением доступа к ресурсам. BeyondCorp от Google является каноническим примером — сотрудники получают доступ к внутренним приложениям через прокси, который проверяет их идентификацию и состояние устройства, независимо от местоположения в сети.

Программно-определяемые периметры, или SDP (Software-Defined Perimeter), которые скрывают сетевые ресурсы до аутентификации пользователей. До аутентификации ресурсы невидимы — они не отвечают на попытки подключения. Это сокращает поверхность атаки, значительно затрудняя разведку.

Микросегментацию, которая применяет политики на уровне рабочих нагрузок. Даже если злоумышленник скомпрометирует одну систему, средства контроля нулевого доверия предотвращают горизонтальное перемещение к другим системам.

Часть 7: Управление информацией и событиями безопасности (SIEM)

Системы SIEM собирают, агрегируют и анализируют данные безопасности со всего предприятия. Они обеспечивают централизованную видимость событий безопасности и позволяют обнаруживать угрозы, которые были бы невидимы при анализе любого отдельного источника данных.

Функции SIEM

Сбор и агрегация журналов — это основа. Системы SIEM собирают журналы от межсетевых экранов, IDS/IPS, серверов, конечных точек, приложений и других источников. Эти журналы нормализуются в единый формат и хранятся для анализа и обеспечения требований хранения.

Корреляция в реальном времени анализирует события по мере их возникновения, ища закономерности, указывающие на атаки. Например, неудачные попытки входа в одну систему, за которыми следует успешный вход в другую, могут указывать на горизонтальное перемещение. По отдельности события нормальны; закономерность раскрывает атаку.

Оповещение и уведомление информируют команды безопасности о значимых событиях. Оповещения могут основываться на отдельных событиях, правилах корреляции или обнаружении аномалий. Настройка оповещений критически важна — слишком много оповещений вызывает усталость, слишком мало — приводит к пропуску реальных угроз.

Информационные панели и отчётность обеспечивают видимость состояния безопасности. Информационные панели отображают метрики и тенденции в реальном времени. Отчёты документируют соответствие требованиям и поддерживают расследования. Руководству нужны иные представления, чем аналитикам, и системы SIEM обеспечивают и то, и другое.

Поддержка криминалистических расследований позволяет аналитикам осуществлять поиск по историческим данным при расследовании инцидентов.

Когда атака обнаружена, аналитикам необходимо понять, что произошло, когда и что было затронуто. SIEM предоставляет данные и поисковые возможности для такого анализа.

Проблемы SIEM

Объём данных огромен и постоянно растёт. Организации генерируют терабайты журналов безопасности ежедневно. Хранение, обработка и поиск по этим данным требуют значительной инфраструктуры.

Усталость от оповещений остаётся серьёзной проблемой. Правила корреляции SIEM генерируют множество ложноположительных срабатываний. Команды безопасности могут получать сотни или тысячи оповещений в день, большинство из которых являются безвредными. Реальные атаки могут быть потеряны в этом шуме.

Квалифицированные аналитики — дефицитный ресурс. Эффективная эксплуатация SIEM требует экспертных знаний в области безопасности, среды организации и самой платформы SIEM. Найти и удержать квалифицированных аналитиков сложно и дорого.

Ведущие поставщики SIEM включают Splunk, Microsoft Sentinel, IBM QRadar и Elastic Security. Облачные предложения SIEM значительно выросли, снижая нагрузку на инфраструктуру, но порождая вопросы облачной безопасности.

Часть 8: Расширенное обнаружение и реагирование (XDR)

XDR (Extended Detection and Response, расширенное обнаружение и реагирование) расширяет концепцию обнаружения и реагирования за пределы конечных точек, охватывая электронную почту, сеть, облако и другие источники данных. XDR стремится преодолеть разрозненность между инструментами безопасности и обеспечить унифицированное обнаружение и реагирование по всей среде.

Традиционные операции безопасности включают множество инструментов, не взаимодействующих между собой. Команда EDR видит угрозы на конечных точках, команда по электронной почте видит почтовые угрозы, сетевая команда видит сетевые угрозы. Злоумышленники используют эти пробелы, проводя многоэтапные атаки, которые ни один инструмент не может увидеть полностью.

XDR коррелирует данные из разных источников для обнаружения сложных атак. Письмо с подозрительным вложением, за которым следуют аномалии поведения на конечной точке, а затем необычный сетевой трафик — это история, которую ни

один из этих сигналов не рассказывает в отдельности. XDR соединяет эти точки воедино.

Платформы XDR обычно включают:

Встроенные интеграции с распространёнными инструментами безопасности, позволяющие собирать данные без индивидуальной разработки.

Унифицированную модель данных, которая нормализует данные из разных источников, обеспечивая корреляцию и анализ.

Автоматизированное расследование, которое обогащает оповещения контекстом, снижая нагрузку на аналитиков.

Оркестрацию реагирования, которая позволяет выполнять действия через множество инструментов с единой консоли.

XDR может быть "нативным" или "открытым". Нативный XDR предоставляется одним производителем, чьи продукты разработаны для совместной работы — например, CrowdStrike Falcon предлагает XDR на базе продуктов CrowdStrike для конечных точек, облака и идентификации. Открытый XDR интегрирует продукты разных производителей, обеспечивая гибкость, но требуя больших усилий по интеграции.

Часть 9: Оркестрация, автоматизация и реагирование (SOAR)

Платформы SOAR (Security Orchestration, Automation, and Response, оркестрация, автоматизация и реагирование в сфере безопасности) автоматизируют рабочие процессы операций безопасности, связывая инструменты безопасности и оркестрируя действия реагирования через автоматизированные сценарии. По мере роста объёма атак и сохранения нехватки кадров в командах безопасности автоматизация становится необходимостью. Если SIEM — это пожарный датчик, который обнаруживает дым и подаёт сигнал тревоги, то SOAR — это спринклерная система, которая автоматически активируется в нужном помещении для тушения огня, не дожидаясь, пока кто-то нажмёт рычаг.

SOAR подключается к различным инструментам безопасности через API, принимает оповещения от таких источников, как межсетевые экраны, инструменты EASM (External Attack Surface Management, управление внешней поверхностью атаки) и ПО для конечных точек, а затем использует сценарии (автоматизированные блок-схемы) для оркестрации задач расследования и реагирования. Например, если инструмент EASM, такой как UpGuard, разработчик решений управления поверхностью атаки, обнаружит критическую уязвимость, платформа SOAR может автоматически создать заявку, уведомить команду безопасности и инициировать рабочий процесс устранения.

Возможности SOAR

Автоматизация сценариев (playbook) кодифицирует процедуры реагирования в исполняемые рабочие процессы. При срабатывании оповещения сценарий автоматически выполняется — собирая контекст, выполняя обогащение, принимая решения и выполняя действия. Аналитики проверяют результаты и обрабатывают исключения.

Например, сценарий для фишингового оповещения может автоматически извлечь URL-адреса и вложения из письма, проверить их по аналитике угроз, просканировать вложения в песочнице, найти других получателей и подготовить сводку для проверки аналитиком. То, что могло бы занять у аналитика 30 минут, сценарий выполняет за секунды.

Управление инцидентами предоставляет структурированное рабочее пространство для расследования и отслеживания инцидентов. Инциденты включают соответствующие доказательства, заметки аналитиков и историю действий. Метрики отслеживают среднее время обнаружения, среднее время реагирования и другие ключевые показатели эффективности.

Интеграция с аналитикой угроз обогащает оповещения контекстом из внешних источников. Известен ли этот IP-адрес как вредоносный? Связан ли этот хеш файла с семейством вредоносного ПО? Аналитика угроз автоматически отвечает на эти вопросы.

Интеграция с инструментами безопасности позволяет SOAR выполнять действия по всему стеку безопасности. Заблокировать IP-адрес на межсетевом экране, изолировать конечную точку, отключить учётную запись пользователя — всё это с платформы SOAR.

Преимущества и проблемы SOAR

Преимущества SOAR включают: более быстрое реагирование (автоматизированные сценарии выполняются немедленно), последовательное реагирование (сценарии не забывают шаги и не допускают ошибок), масштабируемость (автоматизация справляется с объёмами, недоступными людям) и документирование (каждое действие регистрируется для обеспечения соответствия и анализа).

Проблемы включают сложность внедрения (создание эффективных сценариев требует экспертных знаний), усилия по интеграции (подключение ко всем соответствующим инструментам требует времени) и поддержание актуальности сценариев по мере изменения среды. При неправильной настройке автоматизация может нарушить бизнес-процессы. SOAR является мощным инструментом, но требует инвестиций для реализации своих преимуществ.

Ведущие корпоративные платформы SOAR включают Palo Alto Cortex XSOAR, предлагающий обширный маркетплейс интеграций; Splunk SOAR (ранее

Phantom), создающий сценарии, которые действуют на основе данных SIEM-анализа Splunk; Rapid7, разработчик решений кибербезопасности, InsightConnect, облачную платформу с визуальным конструктором рабочих процессов; и ManageEngine Log360, интегрирующий функции SIEM и SOAR в одном продукте. IBM Resilient, Google Chronicle SOAR (ранее Siemplify) и ServiceNow Security Operations также широко развёрнуты.

Более доступные варианты существуют для небольших команд: Tines, разработчик решений автоматизации безопасности, предлагает бесплатный уровень с автоматизацией на основе событий без кода, которая интегрируется с любым инструментом, имеющим API; DTonomy, разработчик решений автоматизации безопасности, предоставляет встроенные интеграции со стартовой ценой $49 в месяц; и Shuffle — платформа SOAR с открытым исходным кодом, которую можно развернуть на собственных серверах. Многие поставщики SIEM добавили возможности SOAR, а платформы XDR часто включают функции автоматизации.

Часть 10: Сетевая безопасность в облаке

Облачные среды требуют адаптации концепций сетевой безопасности к новым архитектурам. Традиционная периметровая безопасность не применима напрямую, когда ваша инфраструктура находится в чужом центре обработки данных.

Сетевая безопасность облачных провайдеров

Каждый крупный облачный провайдер предлагает собственные сервисы сетевой безопасности. AWS предоставляет Security Groups (межсетевой экран уровня экземпляра), Network ACLs (межсетевой экран уровня подсети), AWS WAF и AWS Network Firewall. Azure предоставляет Network Security Groups, Azure Firewall, Azure WAF и Azure Front Door. GCP предоставляет Firewall Rules, Cloud Armor и Cloud NAT.

Эти сервисы глубоко интегрированы с облачной платформой, просты в развёртывании и часто включены в базовую стоимость. Однако они могут уступать по возможностям специализированным продуктам безопасности и создают зависимость от поставщика.

Облачные сервисы безопасности

Брокеры безопасности облачного доступа, или CASB (Cloud Access Security Broker), обеспечивают видимость и контроль для облачных приложений. Они могут обнаруживать использование облачных сервисов, применять политики, обнаруживать угрозы и защищать данные. По мере того как организации

используют сотни облачных приложений, CASB обеспечивают необходимое управление.

Платформы защиты облачных рабочих нагрузок, или CWPP (Cloud Workload Protection Platform), обеспечивают безопасность облачных рабочих нагрузок, включая виртуальные машины, контейнеры и бессерверные функции. Они предоставляют управление уязвимостями, защиту во время выполнения и мониторинг соответствия.

Пограничный сервис безопасного доступа, или SASE (Secure Access Service Edge), объединяет сетевую безопасность и WAN (Wide Area Network, глобальная сеть) в облачный сервис. Пользователи подключаются к ближайшей точке присутствия SASE, которая обеспечивает безопасный доступ как к облачным, так и к локальным ресурсам. SASE сочетает SD-WAN (Software-Defined Wide Area Network, программно-определяемая глобальная сеть), межсетевой экран, безопасный веб-шлюз, CASB и сетевой доступ с нулевым доверием в единый сервис.

Заключение

Сетевая безопасность остаётся необходимой, даже несмотря на размывание периметра. Сегодня мы рассмотрели обширную область — от техник атак на сети и веб-приложения через защитные технологии до сложных платформ XDR и SOAR. Ключевые темы для запоминания:

Знайте своего противника — понимание техник атак, таких как DDoS с усилением, атаки "человек посередине" через подмену ARP, SQL-инъекции и XSS, необходимо для настройки эффективной защиты, оценки рисков и реагирования на инциденты.

Эшелонированная защита — ни одна отдельная технология не гарантирует полную защиту. Сочетайте межсетевые экраны, IDS/IPS, сегментацию и мониторинг для комплексной безопасности.

Видимость необходима — невозможно защитить то, чего не видно. SIEM, XDR и другие платформы предоставляют видимость, необходимую для обнаружения угроз и реагирования на них.

Автоматизация необходима — объём атак превышает возможности ручного реагирования. SOAR и другие средства автоматизации помогают командам безопасности масштабироваться.

Облако меняет всё и ничего — технологии различаются, но принципы остаются неизменными. Применяйте принципы сетевой безопасности к облачным средам с использованием облачных и сторонних инструментов.

На следующей лекции мы сосредоточимся на защите конечных точек и систем — обеспечении безопасности устройств, на которых работают пользователи и

хранятся данные.

Вопросы для обсуждения

1. По мере размывания традиционного сетевого периметра, как организациям следует переосмыслить свою стратегию сетевой безопасности?
2. Каковы компромиссы между всесторонним мониторингом сети и конфиденциальностью пользователей?
3. Как организации могут эффективно управлять объёмом оповещений безопасности, генерируемых современными инструментами сетевой безопасности?

Благодарю за внимание, увидимся в следующий раз.

Контрольные вопросы

1. Назовите все семь уровней модели OSI и объясните, на каких уровнях работают межсетевые экраны с фильтрацией пакетов.
2. Сравните и сопоставьте межсетевые экраны с фильтрацией пакетов, с контролем состояний пакетов (Stateful Packet Inspection, SPI) и уровня приложений.
3. Какие возможности определяют межсетевой экран нового поколения (NGFW)?
4. Опишите три типа DDoS-атак (объёмные, протокольные, на уровне приложений) и объясните, как атаки с усилением достигают многократного увеличения трафика.
5. Что такое подмена ARP и как она позволяет осуществлять атаки типа "человек посередине" в локальной сети?
6. Опишите три типа XSS (хранимый, отражённый, DOM-основанный) и назовите основное средство защиты от XSS-атак.
7. Что такое SQL-инъекция и каково основное средство защиты от неё?
8. Что такое SSRF и почему она опасна в облачных средах?
9. Перечислите пять категорий уязвимостей OWASP Top 10 и объясните, как WAF защищают от них.
10. Сравните OpenVPN с Data Channel Offload (DCO) и WireGuard с точки зрения производительности, гибкости и архитектуры.
11. Почему WireGuard использует ChaCha20 вместо AES, и в каких случаях AES будет быстрее?
12. Объясните разницу между IDS и IPS и опишите три метода обнаружения.

13. Что такое микросегментация и чем она отличается от сегментации на основе VLAN?
14. Опишите назначение DMZ (Demilitarized Zone, демилитаризованная зона) и её типичную архитектуру.
15. Каковы основные функции платформы управления информацией и событиями безопасности (SIEM)?
16. Чем расширенное обнаружение и реагирование (XDR) отличается от традиционных решений SIEM и обнаружения и реагирования на конечных точках (EDR)?
17. Объясните концепцию оркестрации, автоматизации и реагирования (SOAR) и приведите пример сценария безопасности.

Ключевые термины

- **AES-NI**: AES New Instructions, аппаратные инструкции процессора для ускорения шифрования AES
- **BOLA**: Некорректная авторизация на уровне объектов (Broken Object-Level Authorization), наиболее распространённая уязвимость API, при которой контроль доступа не проверяет разрешения для каждого объекта
- **CASB**: Брокер безопасности облачного доступа (Cloud Access Security Broker)
- **ChaCha20**: Потоковый шифр, разработанный Дэниелом Бернштейном, используемый в WireGuard
- **CSRF**: Подделка межсайтовых запросов (Cross-Site Request Forgery), атака, заставляющая браузер отправлять аутентифицированные запросы к целевому сайту
- **DCO**: Data Channel Offload, модуль ядра для ускорения пропускной способности OpenVPN
- **DDoS**: Распределённый отказ в обслуживании (Distributed Denial of Service), атака из множества источников одновременно, перегружающая цель
- **DMZ**: Демилитаризованная зона (Demilitarized Zone)
- **IDS**: Система обнаружения вторжений (Intrusion Detection System)
- **IPS**: Система предотвращения вторжений (Intrusion Prevention System)
- **IPSec**: Internet Protocol Security
- **Атака "злой двойник"**: Мошенническая беспроводная точка доступа, имитирующая легитимную сеть
- **Атака "человек посередине" (MitM)**: Атака, при которой злоумышленник тайно перехватывает и может изменять коммуникацию между двумя

сторонами

- **Атака с усилением**: Техника DDoS, при которой небольшие запросы к сторонним серверам генерируют объёмные ответы, направленные на жертву
- **Внедрение команд ОС**: Атака, выполняющая произвольные команды операционной системы через уязвимое приложение
- **Межсайтовый скриптинг (XSS)**: Атака, внедряющая вредоносные скрипты в веб-страницы, просматриваемые другими пользователями
- **Микросегментация**: Метод обеспечения безопасности, разделяющий сеть на изолированные сегменты для применения детального управления доступом
- **Модель OSI**: Модель взаимодействия открытых систем, семиуровневая архитектура сетевого взаимодействия
- **NGFW**: Межсетевой экран нового поколения (Next-Generation Firewall)
- **Обход каталогов**: Атака, получающая доступ к файлам за пределами предусмотренного каталога с помощью относительных последовательностей пути
- **OpenVPN**: VPN-решение с открытым исходным кодом на основе TLS-шифрования, поддерживающее режимы TUN и TAP
- **Отравление кэша DNS**: Повреждение кэша DNS-резолвера для перенаправления пользователей на вредоносные серверы
- **OWASP Top 10**: Список десяти наиболее критических рисков безопасности веб-приложений, публикуемый Open Web Application Security Project
- **Подмена ARP**: Отправка ложных ARP-сообщений в локальной сети для связывания MAC-адреса злоумышленника с легитимным IP-адресом
- **RFC 1918**: Стандарт, определяющий диапазоны частных IPv4-адресов (10.x, 172.16-31.x, 192.168.x)
- **SASE**: Пограничный сервис безопасного доступа (Secure Access Service Edge)
- **Сегментация сети**: Практика разделения сети на меньшие подсети для повышения безопасности и производительности
- **Сканирование портов**: Зондирование сетевых портов целевой системы для обнаружения работающих сервисов и потенциальных уязвимостей
- **SIEM**: Управление информацией и событиями безопасности (Security Information and Event Management)
- **Slowloris**: DDoS-атака уровня 7, удерживающая соединения открытыми неполными HTTP-запросами
- **SOAR**: Оркестрация, автоматизация и реагирование (Security Orchestration, Automation, and Response)

- **SQL-инъекция**: Атака, внедряющая вредоносные SQL-выражения в запросы приложения через неочищенный ввод
- **SSRF**: Подделка запросов на стороне сервера (Server-Side Request Forgery), атака, заставляющая сервер выполнять запросы к непредусмотренным адресатам
- **SYN-флуд**: DoS-атака, исчерпывающая ресурсы сервера отправкой TCP SYN-пакетов без завершения рукопожатия
- **VLAN**: Виртуальная локальная сеть — логическое разделение сети на канальном уровне (Virtual Local Area Network)
- **VPN**: Виртуальная частная сеть (Virtual Private Network)
- **WAF**: Межсетевой экран уровня веб-приложений (Web Application Firewall)
- **WireGuard**: Современный протокол VPN, использующий шифрование ChaCha20, включённый в ядро Linux 5.6
- **XDR**: Расширенное обнаружение и реагирование (Extended Detection and Response)