Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«СИБИРСКАЯ ГОСУДАРСТВЕННАЯ ГЕОДЕЗИЧЕСКАЯ АКАДЕМИЯ»
(ФГБОУ ВПО «СГГА»)

Институт оптики и оптических технологий

Кафедра наносистем и оптотехники

КОНСПЕКТ ЛЕКЦИЙ

ПО ДИСЦИПЛИНЕ

СИСТЕМЫ ОБНАРУЖЕНИЯ ВТОРЖЕНИЙ

Новосибирск
СГГА

Введение

Общие сведения. IDS - Intrusion Detection Systems.Классификация IDS. Характеристики систем обнаружения вторжений. Архитектура IDS. Основные элементы локальной архитектуры систем обнаружения вторжений. Стандарты в области систем обнаружения вторжений.

Системы обнаружения вторжений (IDS - Intrusion Detection Systems) - один из важнейших элементов систем информационной безопасности сетей любого современного предприятия. Рост в последние годы числа проблем, связанных с компьютерной безопасностью, привёл к тому, что системы обнаружения вторжения очень быстро стали ключевым компонентом любой стратегии сетевой защиты. За последние несколько лет их популярность значительно возросла, поскольку продавцы средств защиты значительно улучшили качество и совместимость своих программ.

Системами обнаружения вторжений (СОВ) называют множество различных программных и аппаратных средств, объединяемых одним общим свойством - они занимаются анализом использования вверенных им ресурсов и, в случае обнаружения каких-либо подозрительных или просто нетипичных событий, способны предпринимать некоторые самостоятельные действия по обнаружению, идентификации и устранению их причин.

Системами обнаружения вторжений (СОВ) называют множество различных программных и аппаратных средств, объединяемых одним общим свойством - они занимаются анализом использования вверенных им ресурсов и, в случае обнаружения каких-либо подозрительных или просто нетипичных событий, способны предпринимать некоторые самостоятельные действия по обнаружению, идентификации и устранению их причин.

Но системы обнаружения вторжений лишь один из инструментов защитного арсенала и он не должен рассматриваться как замена для любого из других защитных механизмов. Защита информации наиболее эффективна, когда в интрасети поддерживается многоуровневая защита. Она складывается из следующих компонентов:

1. Политика безопасности интрасети организации;
2. Система защиты хостов в сети;
3. Сетевой аудит;
4. Защита на основе маршрутизаторов;
5. Межсетевые экраны;
6. Системы обнаружения вторжений;
7. План реагирования на выявленные атаки.

Следовательно, для полной защиты целостности сети необходима реализация всех вышеперечисленных компонентов защиты. И использование многоуровневой защиты является наиболее эффективным методом предотвращения несанкционированного использования компьютерных систем и сетевых сервисов. Таким образом, система обнаружения вторжений – это

одна из компонент обеспечения безопасности сети в многоуровневой стратегии её защиты.

Классификация IDS

Для проведения классификации IDS необходимо учесть несколько факторов (рисунок 1) .

Метод обнаружения описывает характеристики анализатора. Когда IDS использует информацию о нормальном поведении контролируемой системы, она называется поведенческой. Когда IDS работает с информацией об атаках, она называется интеллектуальной.

Рисунок 1. Характеристики систем обнаружения вторжений

Поведение после обнаружения указывает на реакцию IDS на атаки. Реакция может быть активной – IDS предпринимает корректирующие (устраняет лазейки) или действительно активные (закрывает доступ для возможных нарушителей, делая недоступными сервисы) действия. Если IDS только выдаёт предупреждения, её называют пассивной.

Расположение источников результата аудита подразделяет IDS в зависимости от вида исходной информации, которую они анализируют. Входными данными для них могут быть результаты аудита, системные регистрационные файлы или сетевые пакеты.

Частота использования отражает либо непрерывный мониторинг контролируемой системы со стороны IDS, либо соответствующие периодическим запускам IDS для проведения анализа.

Классифицировать IDS можно также по нескольким параметрам. По способам реагирования различают статические и динамические IDS. Статические средства делают «снимки» (snapshot) среды и осуществляют их анализ, разыскивая уязвимое ПО, ошибки в конфигурациях и т.д. Статические IDS проверяют версии работающих в системе приложений на наличие

известных уязвимостей и слабых паролей, проверяют содержимое специальных файлов в директориях пользователей или проверяют конфигурацию открытых сетевых сервисов. Статические IDS обнаруживают следы вторжения. Динамические IDS осуществляют мониторинг в реальном времени всех действий, происходящих в системе, просматривая файлы аудита или сетевые пакеты, передаваемые за определённый промежуток времени. Динамические IDS реализуют анализ в реальном времени и позволяют постоянно следить за безопасностью системы.

Рисунок 2. Классификация систем обнаружения вторжений

По способу сбора информации различают сетевые и системные IDS. Сетевые (NIDS)контролируют пакеты в сетевом окружении и обнаруживают попытки злоумышленника проникнуть внутрь защищаемой системы или реализовать атаку «отказ в обслуживании». Эти IDS работают с сетевыми потоками данных. Типичный пример NIDS – система, которая контролирует большое число TPC-запросов на соединение (SYN) со многими портами на выбранном компьютере, обнаруживая, таким образом, что кто-то пытается осуществить сканирование TCP-портов. Сетевая IDS может запускаться либо на отдельном компьютере, который контролирует свой собственный трафик, либо на выделенном компьютере, прозрачно просматривающим весь трафик в сети (концентратор, маршрутизатор). Сетевые IDS контролируют много компьютеров, тогда как другие IDS контролируют только один. IDS, которые устанавливаются на хосте и обнаруживают злонамеренные действия на нём называются хостовыми или системными IDS. Примерами хостовых IDS могут быть системы контроля целостности файлов (СКЦФ), которые проверяют системные файлы с целью определения, когда в них были внесены изменения. Мониторы регистрационных файлов (Log-file monitors, LFM), контролируют регистрационные файлы, создаваемые сетевыми сервисами и службами. Обманные системы, работающие с псевдосервисами, цель которых заключается

в воспроизведении хорошо известных уязвимостей для обмана злоумышленников.

По методам анализа IDS делят на две группы : IDS, которые сравнивают информацию с предустановленной базой сигнатур атак и IDS, контролирующие частоту событий или обнаружение статистических аномалий.

Анализ сигнатур был первым методом, примененным для обнаружения вторжений. Он базируется на простом понятии совпадения последовательности с образцом. Во входящем пакете просматривается байт за байтом и сравнивается с сигнатурой (подписью) – характерной строкой программы, указывающей на характеристику вредного трафика. Такая подпись может содержать ключевую фразу или команду, которая связана с нападением. Если совпадение найдено, объявляется тревога.

Второй метод анализа состоит в рассмотрении строго форматированных данных трафика сети, известных как протоколы. Каждый пакет сопровождается различными протоколами. Авторы IDS, зная это, внедрили инструменты, которые разворачивают и осматривают эти протоколы, согласно стандартам. Каждый протокол имеет несколько полей с ожидаемыми или нормальными значениями. Если что-нибудь нарушает эти стандарты, то вероятна злонамеренность. IDS просматривает каждое поле всех протоколов входящих пакетов: IP, TCP, и UDP. Если имеются нарушения протокола, например, если он содержит неожиданное значение в одном из полей, объявляется тревога

Системы анализа сигнатуры имеют несколько важных сильных сторон. Во-первых, они очень быстры, так как полный анализ пакета - относительно тяжелая задача. Правила легко написать, понять и настроить. Кроме того, имеется просто фантастическая поддержка компьютерного сообщества в быстром производстве сигнатур для новых опасностей. Эти системы превосходят все другие при отлове хакеров на первичном этапе: простые атаки имеют привычку использовать некие предварительные действия, которые легко распознать. Наконец, анализ, основанный на сигнатуре, точно и быстро сообщает, что в системе все нормально (если это действительно так), поскольку должны произойти некие особые события для объявления тревоги.

С другой стороны IDS, основывающаяся только на анализе сигнатур, имеет определенные слабости. Являясь первоначально очень быстрой, со временем скорость ее работы будет замедляться, поскольку возрастает число проверяемых сигнатур. Это – существенная проблема, поскольку число проверяемых сигнатур может расти очень быстро. Фактически, каждая новая атака или действие, придуманное атакующим, увеличивает список проверяемых сигнатур. Не помогут даже эффективные методы работы с данными и пакетами: огромное количество слегка измененных атак могут проскользнуть через такую систему

Имеется и другая сторона проблемы: так как система работает, сравнивая список имеющихся сигнатур с данными пакета, такая IDS может выявить только уже известные атаки, сигнатуры которых имеются.

Но необходимо отметить, что согласно статистике 80% атак происходит по давно известным сценариям. Наличие в системе обнаружения сигнатур известных атак даёт высокий процент обнаружения вторжений.

В случае анализа протоколов тоже имеются свои достоинства и недостатки. Из-за предпроцессов, требующих тщательной экспертизы протоколов, анализ протокола может быть довольно медленным. Кроме того, правила проверки для системы протокола трудно написать и понять. Можно даже сказать, что в этом случае приходится уповать на добросовестность производителя программы, так как правила относительно сложны и трудны для самостоятельной настройки.

На первый взгляд, IDS на основе анализа протокола работают медленнее, чем системы на основе сигнатуры, они, более «основательны» в смысле масштабности и результатов. Кроме того, эти системы ищут «генетические нарушения» и часто могут отлавливать свежайшие “эксплоиты нулевого дня”, что в принципе

Архитектура IDS

У систем обнаружения вторжений целесообразно различать локальную и глобальную архитектуру. В рамках локальной архитектуры реализуются элементарные составляющие, которые затем могут быть объединены для обслуживания корпоративных систем .

Основные элементы локальной архитектуры и связи между ними показаны на рисунке 3. Первичный сбор данных осуществляют агенты, называемые также сенсорами. Регистрационная информация может извлекаться из системных или прикладных журналов (технически несложно получать ее и напрямую от ядра ОС), либо добываться из сети с помощью соответствующих механизмов активного сетевого оборудования или путем перехвата пакетов посредством установленной в режим мониторинга сетевой карты.

На уровне агентов (сенсоров) может выполняться фильтрация данных с целью уменьшения их объема. Это требует от агентов некоторого интеллекта, но зато разгружает остальные компоненты системы.

Агенты передают информацию в центр распределения, который приводит ее к единому формату, возможно, осуществляет дальнейшую фильтрацию, сохраняет в базе данных и направляет для анализа статистическому и экспертному компонентам. Один центр распределения может обслуживать несколько сенсоров.

Содержательный активный аудит начинается со статистического и экспертного компонентов. Если в процессе статистического или экспертного анализа выявляется подозрительная активность, соответствующее сообщение направляется решателю, который определяет, является ли тревога оправданной, и выбирает способ реагирования.

Хорошая система обнаружения вторжений должна уметь внятно объяснить, почему она подняла тревогу, насколько серьезна ситуация и каковы рекомендуемые способы действия. Если выбор должен оставаться за

человеком, то пусть он сводится к нескольким элементам меню, а не к решению концептуальных проблем.

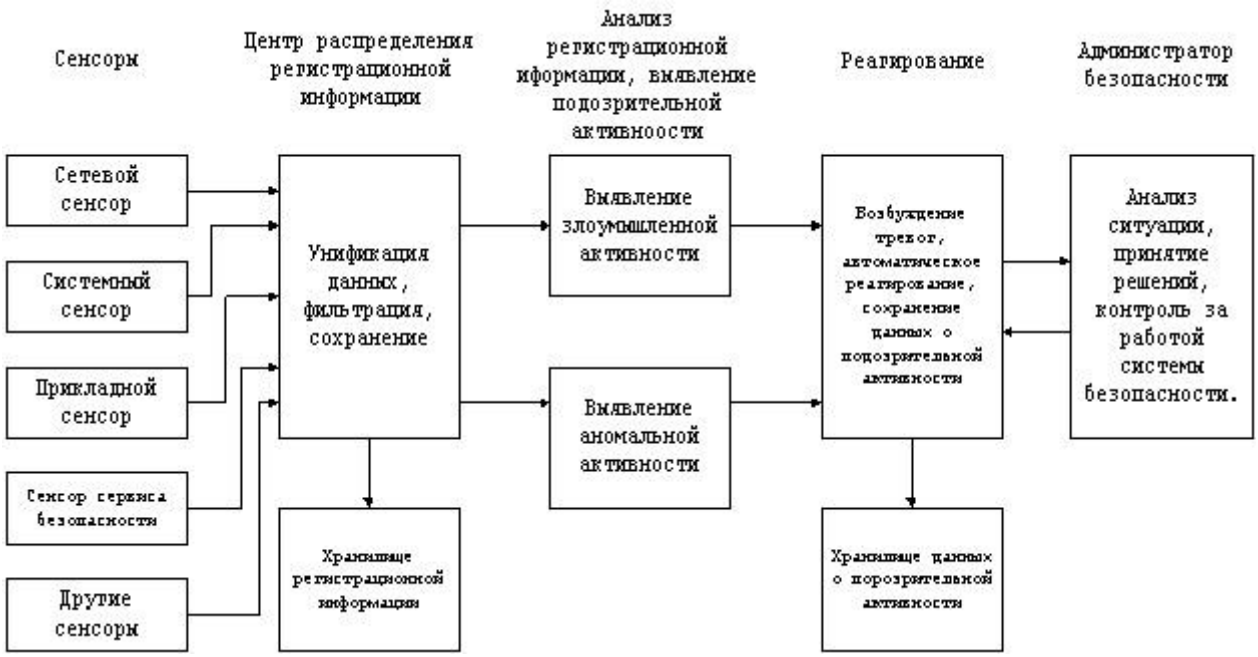


Рисунок 3. Основные элементы локальной архитектуры систем обнаружения вторжений

Глобальная архитектура подразумевает организацию одноранговых и разноранговых связей между локальными системами обнаружения вторжений (рисунок 4).

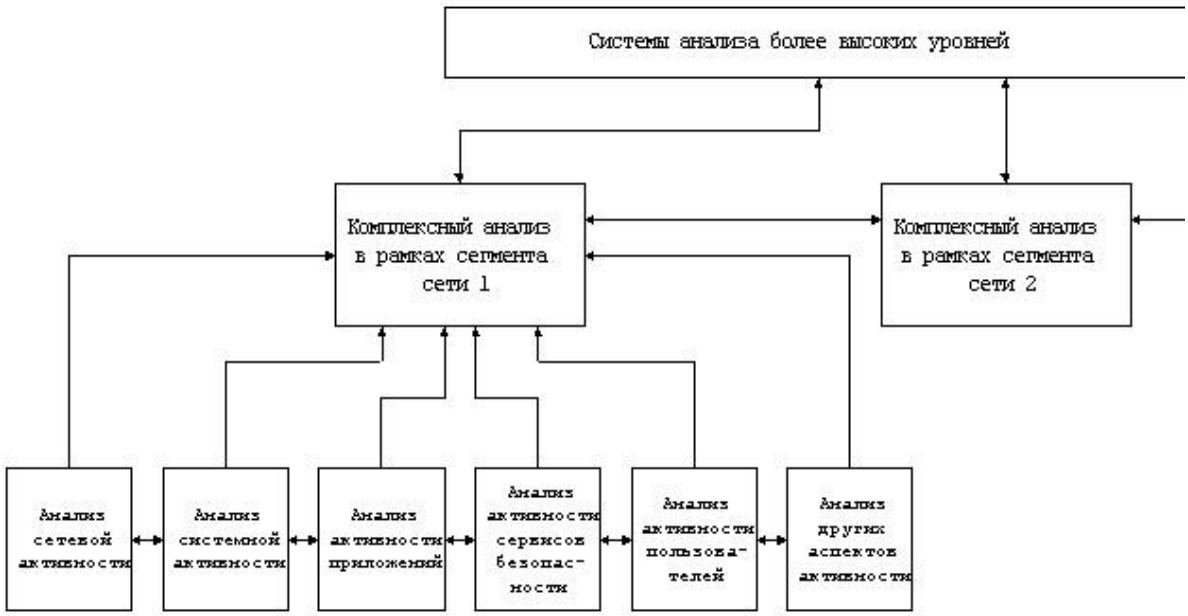


Рисунок 4. Глобальная архитектура систем обнаружения вторжений

На одном уровне иерархии располагаются компоненты, анализирующие подозрительную активность с разных точек зрения. Например, на хосте могут располагаться подсистемы анализа поведения пользователей и приложений. Их

может дополнять подсистема анализа сетевой активности. Когда один компонент обнаруживает что-то подозрительное, то во многих случаях целесообразно сообщить об этом соседям либо для принятия мер, либо для усиления внимания к определенным аспектам поведения системы.

Разноранговые связи используются для обобщения результатов анализа и получения целостной картины происходящего. Иногда у локального компонента недостаточно оснований для возбуждения тревоги, но "по совокупности" подозрительные ситуации могут быть объединены и совместно проанализированы, после чего порог подозрительности окажется превышенным. Целостная картина, возможно, позволит выявить скоординированные атаки на разные участки информационной системы и оценить ущерб в масштабе организации.

Стандарты в области систем обнаружения вторжений

Обмен данными о подозрительной активности. Многие атаки на информационные системы носят распределенный характер. При этом разные средства активного аудита видят один и тот же инцидент с разных точек зрения.

Разделение информации о подозрительной активности является главным направлением работ созданной в рамках Тематической группы по технологии Интернет (Internet Engineering Task Force, IETF) Рабочей группы по обнаружению вторжений (Intrusion Detection Working Group, IDWG).

Группе IDWG предстоит специфицировать формат IDMEF (Intrusion Detection Message Exchange Format) — формат обмена данными между компонентами IDS. Он используется для передачи предупреждающих сообщений о подозрительных событиях между системами выявления атак. Данный формат должен обеспечить совместимость между коммерческими и свободно распространяемыми IDS и возможность их совместного использования для обеспечения наивысшего уровня защищенности.

IDMEF должен поддерживать все механизмы обнаружения подозрительной активности. Он должен быть рассчитан на IPv6, содержать все необходимое для интернационализации, поддерживать фильтрацию и агрегирование сообщений компонентом реагирования, их надежную доставку (в том числе через межсетевой экран без внесения в конфигурацию последнего изменений, способных ослабить периметр безопасности).

Разумеется, формат IDMEF должен поддерживать взаимную аутентификацию общающихся сторон, неотказуемость от факта передачи, а также целостность и конфиденциальность потока сообщений.

В сообщениях формата IDMEF должны содержаться дата и время подозрительных событий и, если возможно, дата и время атаки.

Если анализатор сам принял ответные меры, в IDMEF-сообщениях должна быть информация об этом. Если анализатор может оценить последствия зафиксированной атаки, он также обязан сообщить об этом.

Формат IDMEF должен поддерживать информацию о производителе системы обнаружения вторжений, сгенерировавшей сообщение, а также расширения, специфичные для конкретной системы.

Предполагается, что будет утвержден список стандартных атак и методов их проведения. Если анализатор может идентифицировать атаку и используемый метод, он должен включить соответствующую информацию в IDMEF-сообщение. Если атака является нестандартной, ее имя может быть специфичным для производителя системы активного аудита.

Общий каркас систем обнаружения вторжений (Common Intrusion Detection Framework, CIDF) разрабатывается группой исследовательских организаций, финансируемых агентством DARPA и работающих в области выявления подозрительной активности.

В рамках CIDF разработан язык описания подозрительной активности и способ кодирования информации о подозрительных событиях. Язык приспособлен для описания, по крайней мере, трех видов сообщений:

- "сырой" информации о событиях (например, записей регистрационного журнала или сетевых пакетов);
- результатов анализа (таких как выявленные аномалии или атаки);
- рекомендованных реакций (прервать какую-либо активность или изменить конфигурацию защитных средств).

Кроме того, на языке могут быть описаны следующие сущности:

- связи между событиями (например, причинно-следственные);
- роли объектов в событиях (например, объект инициировал событие);
- свойства объектов;
- связи между объектами.

1. Основные понятия сетей TCP/IP, аппаратура локальных сетей

Основы TCP/IP

TCP/IP - это аббревиатура термина Transmission Control Protocol/Internet Protocol (Протокол управления передачей/Протокол Internet). В терминологии вычислительных сетей протокол - это заранее согласованный стандарт, который позволяет двум компьютерам обмениваться данными. Фактически TCP/IP не один протокол, а несколько. Именно поэтому вы часто слышите, как его называют набором, или комплектом протоколов, среди которых TCP и IP - два основных.

TCP - это протокол более высокого уровня, который позволяет прикладным программам, запущенным на различных главных компьютерах сети, обмениваться потоками данных. TCP делит потоки данных на цепочки, которые называются TCP-сегментами, и передает их с помощью IP. В большинстве случаев каждый TCP-сегмент пересылается в одной IP-дейтаграмме. Однако при необходимости TCP будет расщеплять сегменты на несколько IP-дейтаграмм, вмещающихся в физические кадры данных, которые

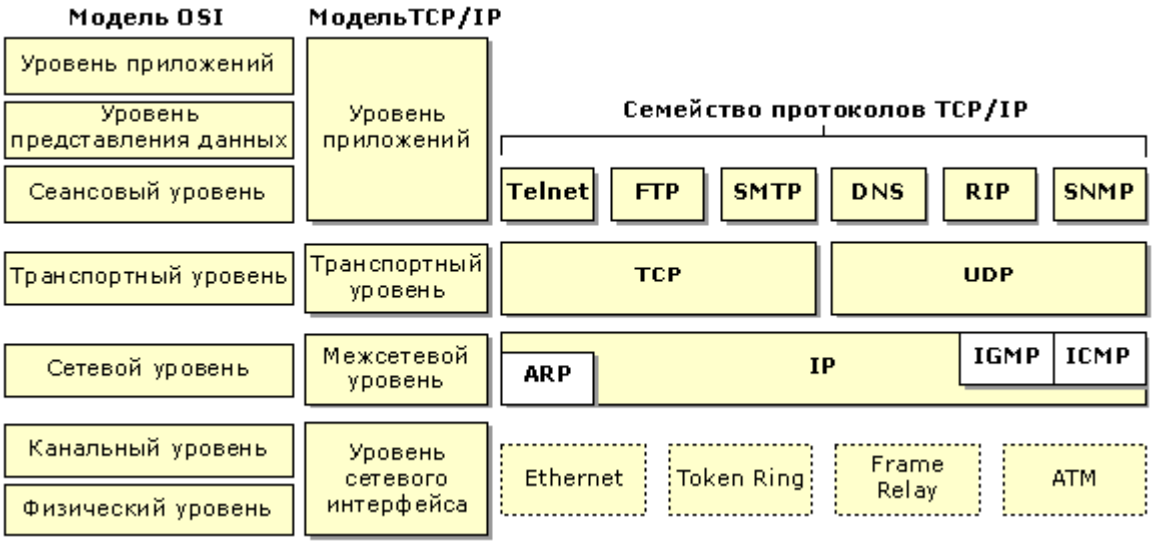
используют для передачи информации между компьютерами в сети. Поскольку IP не гарантирует, что дейтаграммы будут получены в той же самой последовательности, в которой они были посланы, TCP осуществляет повторную "сборку" TCP-сегментов на другом конце маршрута, чтобы образовать непрерывный поток данных. FTP и telnet - это два примера популярных прикладных программ TCP/IP, которые опираются на использование TCP.

Другая важная составляющая комплекта TCP/IP - User Datagram Protocol (UDP, протокол пользовательских дейтаграмм), который похож на TCP, но более примитивен. TCP - "надежный" протокол, потому что он обеспечивает проверку на наличие ошибок и обмен подтверждающими сообщениями чтобы данные достигали своего места назначения заведомо без искажений. UDP - "ненадежный" протокол, ибо не гарантирует, что дейтаграммы будут приходить в том порядке, в котором были посланы, и даже того, что они придут вообще. Если надежность - желательное условие, для его реализации потребуется программное обеспечение. Но UDP по-прежнему занимает свое место в мире TCP/IP, и испльзуется во многих программах. Прикладная программа SNMP (Simple Network Management Protocol, простой протокол управления сетями), реализуемый во многих воплощениях TCP/IP, - это один из примеров программ UDP.

Другие TCP/IP протоколы играют менее заметные, но в равной степени важные роли в работе сетей TCP/IP. Например, протокол определения адресов (Address Resolution Protocol, ARP) ппреобразует IP-адреса в физические сетевые адреса, такие, как идентификаторы Ethernet. Родственный протокол - протокол обратного преобразования адресов (Reverse Address Resolution Protocol, RARP) - выполняет обеспечивает обратное действие, преобразуя физические сетевые адреса в IP-адреса. Протокол управления сообщениями Internet (Internet Control Message Protocol, ICMP) представляет собой протокол сопровождения, который использует IP для обмена управляющей информацией и контроля над ошибками, относящимися к передаче пакетов IP. Например, если маршрутизатор не может передать IP-дейтаграмму, он использует ICMP, с тем чтобы информировать отправителя, что возникла проблема. Краткое описание некоторых других протоколов, которые "прячутся под зонтиком" TCP/IP, приведено во врезке.

Стек протоколов TCP/IP.

Стек TCP/IP - набор протоколов, разработанных для обеспечения взаимосвязи различных устройств в сети Интернет. Стек включает следующие протоколы



Протокол IP (Internet protocol) - основной протокол сетевого уровня. Определяет способ адресации на сетевом уровне. Обеспечивает маршрутизацию в сетях, представляющих собой объединение сетей, базирующихся на разных сетевых технологиях.

Протокол ARP (Address Resolution Protocol) - вспомогательный протокол стека TCP/IP, предназначенный для определения аппаратного адреса узла назначения по заданному IP-адресу.

Протокол ICMP (Internet Control Message Protocol) - вспомогательный протокол стека TCP/IP, предназначенный для обмена информацией об ошибках передачи данных протоколом IP, а также для обмена управляющей информацией на сетевом уровне. В частности, утилита PING использует этот протокол для посылки так называемого "эхо-запроса".

Протокол IGMP (Internet Group Management Protocol) - протокол, используемый для отправки данных определенной группе получателей.

Протокол TCP (Transmission Control Protocol) - протокол, обеспечивающий гарантированную доставку данных с установлением виртуального соединения между программами, которым требуется использовать сетевые услуги. Установление виртуального соединения предполагает, что получатель готов к приему данных от конкретного отправителя. Это означает, что все параметры взаимодействия согласованы, и компьютер-получатель выделил соответствующие ресурсы для обеспечения приема.

Протокол UDP (User Datagram Protocol) - протокол, обеспечивающий негарантированную доставку данных без установления виртуального соединения между программами, которым требуется использовать сетевые услуги.

Маршрутизация в IP-сетях

Маршрутизатором, или **шлюзом**, называется узел сети с несколькими IP-интерфейсами, подключенными к разным IP-сетям, осуществляющий на основе

решения задачи маршрутизации перенаправление дейтаграмм из одной сети в другую для доставки от отправителя к получателю.

Маршрутизаторы представляют собой либо специализированные вычислительные машины, либо компьютеры с несколькими IP-интерфейсами, работа которых управляется специальным программным обеспечением.

Маршрутизация служит для приема пакета от одного устройства и передачи его по сети другому устройству через другие сети. Маршрутизаторы направляют (перенаправляют) трафик во все сети, составляющие объединенную сеть.

Для маршрутизации пакета маршрутизатор должен владеть следующей информацией:

- Адрес назначения
- Соседний маршрутизатор, от которого он может узнать об удаленных сетях
- Доступные пути ко всем удаленным сетям
- Наилучший путь к каждой удаленной сети
- Методы обслуживания и проверки информации о маршрутизации

Защита маршрутизатора

Мероприятия по защите маршрутизатора проводятся с целью предотвращения атак, направленных на нарушение схему маршрутизации датаграмм или на захват маршрутизатора злоумышленником.

- Использовать аутентификацию сообщений протоколов маршрутизации с помощью алгоритма MD5.
- Осуществлять фильтрацию маршрутов, объявляемых сетями-клиентами, провайдером или другими автономными системами. Фильтрация выполняется в соответствии с маршрутной политикой организации; маршруты, не соответствующие политике, игнорируются.
- Использовать на маршрутизаторе, а также на коммутаторах статическую ARP-таблицу узлов сети организации.
- Отключить на маршрутизаторе все ненужные сервисы (особенно так называемые «диагностические» или «малые» сервисы TCP: echo, chargen, daytime, discard, и UDP: echo, chargen, discard).
- Ограничить доступ к маршрутизатору консолью или выделенной рабочей станцией администратора, использовать парольную защиту; не использовать telnet для доступа к маршрутизатору в сети, которая может быть прослушана.
- Использовать последние версии и обновления программного обеспечения, следить за бюллетенями по безопасности, выпускаемыми производителем.

Аппаратура локальных сетей

Аппаратура локальных сетей обеспечивает реальную связь между абонентами. Выбор аппаратуры имеет важнейшее значение на этапе

проектирования сети, так как стоимость аппаратуры составляет наиболее существенную часть от стоимости сети в целом, а замена аппаратуры связана не только с дополнительными расходами, но зачастую и с трудоемкими работами. К аппаратуре локальных сетей относятся:

- кабели для передачи информации;
- разъемы для присоединения кабелей;
- согласующие терминаторы;
- сетевые адаптеры ;
- репитеры ;
- трансиверы ;
- концентраторы ;
- мосты ;
- маршрутизаторы ;
- шлюзы.

О первых трех компонентах сетевой аппаратуры уже говорилось в предыдущих главах. А сейчас следует остановиться на функциях остальных компонентов.

Сетевые адаптеры (они же контроллеры, карты, платы, интерфейсы, NIC – Network Interface Card) – это основная часть аппаратуры локальной сети. Назначение сетевого адаптера – сопряжение компьютера (или другого абонента) с сетью, то есть обеспечение обмена информацией между компьютером и каналом связи в соответствии с принятыми правилами обмена. Именно они реализуют функции двух нижних уровнеймодели OSI. Как правило, сетевые адаптеры выполняются в виде платы (рис. 5.5), вставляемой в слоты расширения системной магистрали (шины) компьютера (чаще всего PCI, ISA или PC-Card). Плата сетевого адаптера обычно имеет также один или несколько внешних разъемов для подключения к ней кабеля сети.

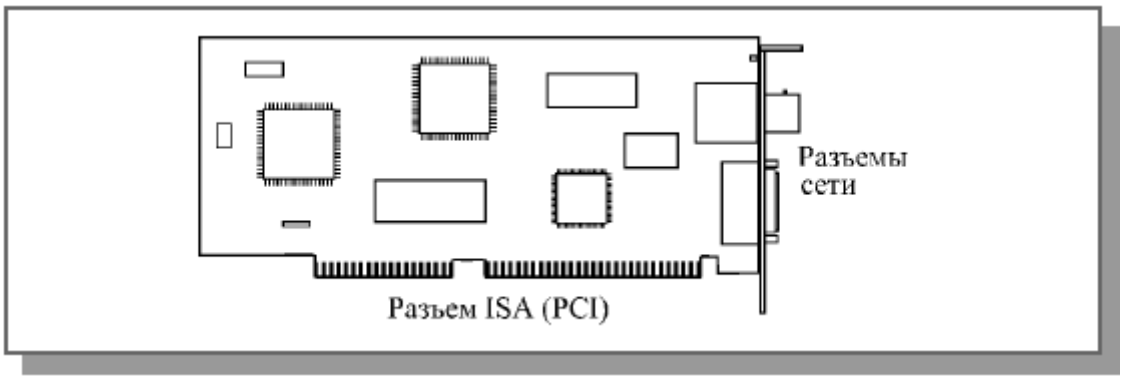


Рис. 5.5. Плата сетевого адаптера

Например, сетевые адаптерыEthernet могут выпускаться со следующими наборами разъемов:

- TPO – разъем RJ-45 (для кабеля на витых парах по стандарту 10BASE-T).
- TPC – разъемы RJ-45 (для кабеля на витых парах 10BASE-T) и BNC (для коаксиального кабеля 10BASE2).
- TP – разъем RJ-45 (10BASE-T) и трансиверный разъем AUI.

- Combo – разъемы RJ-45 (10BASE-T), BNC (10BASE2), AUI.
- Coax – разъемы BNC, AUI.
- FL – разъем ST (для оптоволоконного кабеля 10BASE-FL).

Функции сетевого адаптера делятся на магистральные и сетевые. К магистральным относятся те функции, которые осуществляют взаимодействие адаптера с магистралью (системной шиной) компьютера (то есть опознание своего магистрального адреса, пересылка данных в компьютер и из компьютера, выработка сигнала прерывания процессора и т.д.). Сетевые функции обеспечивают общение адаптера с сетью.

К основным сетевым функциям адаптеров относятся:

- гальваническая развязка компьютера и кабеля локальной сети (для этого обычно используется передача сигналов через импульсные трансформаторы);
- преобразование логических сигналов в сетевые (электрические или световые) и обратно;
- кодирование и декодирование сетевых сигналов, то есть прямое и обратное преобразование сетевых кодов передачи информации (например, манчестерский код);
- опознание принимаемых пакетов (выбор из всех приходящих пакетов тех, которые адресованы данному абоненту или всем абонентам сети одновременно);
- преобразование параллельного кода в последовательный при передаче и обратное преобразование при приеме;
- буферизация передаваемой и принимаемой информации в буферной памяти адаптера ;
- организация доступа к сети в соответствии с принятым методом управления обменом;
- подсчет контрольной суммы пакетов при передаче и приеме.

Типичный алгоритм взаимодействия компьютера с сетевым адаптером выглядит следующим образом.

Если компьютер хочет передать пакет, то он сначала формирует этот пакет в своей памяти, затем пересылает его в буферную памятьсетевого адаптера и дает команду адаптеру на передачу. Адаптер анализирует текущее состояние сети и при первой же возможности выдает пакет в сеть (выполняет управление доступом к сети). При этом он производит преобразование информации из буферной памяти в последовательный вид для побитной передачи по сети, подсчитывает контрольную сумму, кодирует биты пакета в сетевой код и через узел гальванической развязки выдает пакет в кабель сети. Буферная память в данном случае позволяет освободить компьютер от контроля состояния сети, а также обеспечить требуемый для сети темп выдачи информации.

Если по сети приходит пакет, то сетевой адаптер через узел гальванической развязки принимает биты пакета, производит их декодирование из сетевого кода и сравнивает сетевой адрес приемника из пакета со своим собственным адресом. Адрессетевого адаптера, как правило, устанавливается производителем адаптера. Если адрес совпадает, то сетевой адаптер записывает

пришедший пакет в свою буферную память и сообщает компьютеру (обычно – сигналом аппаратного прерывания) о том, что пришел пакет и его надо читать. Одновременно с записью пакета производится подсчет контрольной суммы, что позволяет к концу приема сделать вывод, имеются ли ошибки в этом пакете. Буферная память в данном случае опять же позволяет освободить компьютер от контроля сети, а также обеспечить высокую степень готовности сетевого адаптера к приему пакетов.

Чаще всего сетевые функции выполняются специальными микросхемами высокой степени интеграции, что дает возможность снизить стоимостьадаптера и уменьшить площадь его платы.

Некоторые адаптеры позволяют реализовать функцию удаленной загрузки, то есть поддерживать работу в сети бездисковых компьютеров, загружающих свою операционную систему прямо из сети. Для этого в состав таких адаптеров включается постоянная память с соответствующей программой загрузки. Правда, не все сетевые программные средства поддерживают данный режим работы.

Сетевой адаптер выполняет функции первого и второго уровнеймодели OSI (рис. 5.6).

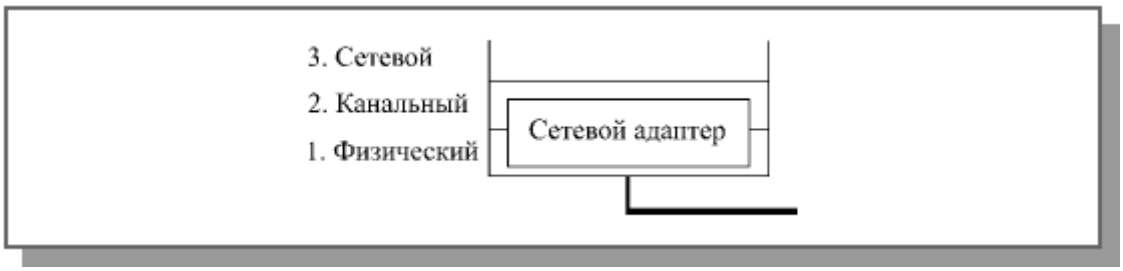


Рис. 5.6. Функции сетевого адаптера в модели OSI

Все остальные аппаратные средства локальных сетей (кроме адаптеров) имеют вспомогательный характер, и без них часто можно обойтись. Это сетевые промежуточные устройства.

Трансиверы или приемопередатчики (от английского TRANsmitter + reCEIVER) служат для передачи информации между адаптером и кабелем сети или между двумя сегментами (частями) сети. Трансиверы усиливают сигналы, преобразуют их уровни или преобразуют сигналы в другую форму (например, из электрической в световую и обратно). Трансиверами также часто называют встроенные в адаптер приемопередатчики.

Репитеры или повторители (repeater) выполняют более простую функцию, чем трансиверы. Они не преобразуют ни уровни сигналов, ни их физическую природу, а только восстанавливают ослабленные сигналы (их амплитуду и форму), приводя их к исходному виду. Цель такой ретрансляции сигналов состоит исключительно в увеличении длины сети (рис. 5.7).

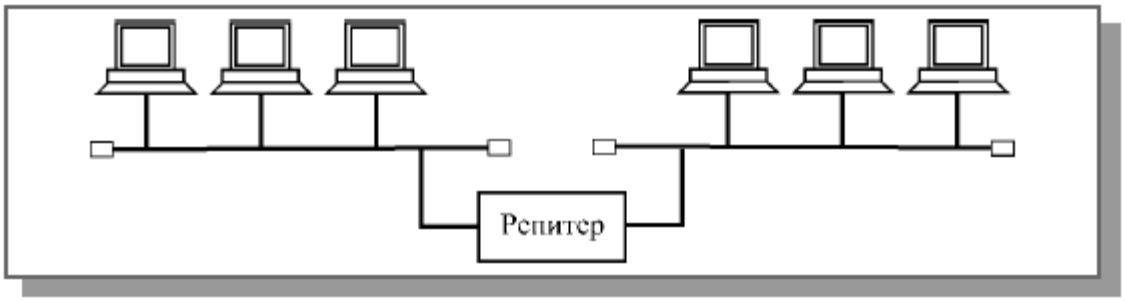


Рис. 5.7. Соединение репитером двух сегментов сети

Однако часто репитеры выполняют и некоторые другие, вспомогательные функции, например, гальваническую развязку соединяемых сегментов и оконечное согласование. Репитеры так же как трансиверы не производят никакой информационной обработки проходящих через них сигналов.

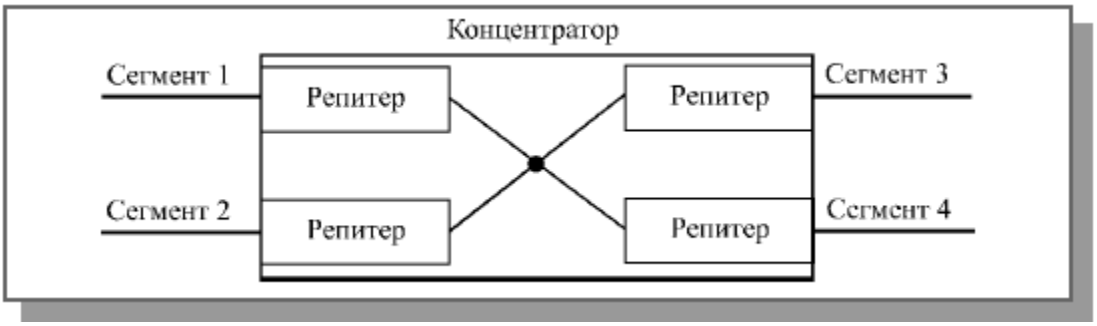


Рис. 5.8. Структура репитерного концентратора

Концентраторы (хабы, hub), как следует из их названия, служат для объединения в сеть нескольких сегментов. Концентраторы (или репитерные концентраторы) представляют собой несколько собранных в едином конструктиве репитеров, они выполняют те же функции, что и репитеры (рис. 5.8).

Преимущество подобных концентраторов по сравнению с отдельными репитерами в том, что все точки подключения собраны в одном месте, это упрощает реконфигурацию сети, контроль и поиск неисправностей. К тому же все репитеры в данном случае питаются от единого качественного источника питания.

Концентраторы иногда вмешиваются в обмен, помогая устранять некоторые явные ошибки обмена. В любом случае они работают на первом уровне модели OSI, так как имеют дело только с физическими сигналами, с битами пакета и не анализируют содержимое пакета, рассматривая пакет как единое целое (рис. 5.9). На первом же уровне работают и трансиверы, и репитеры.

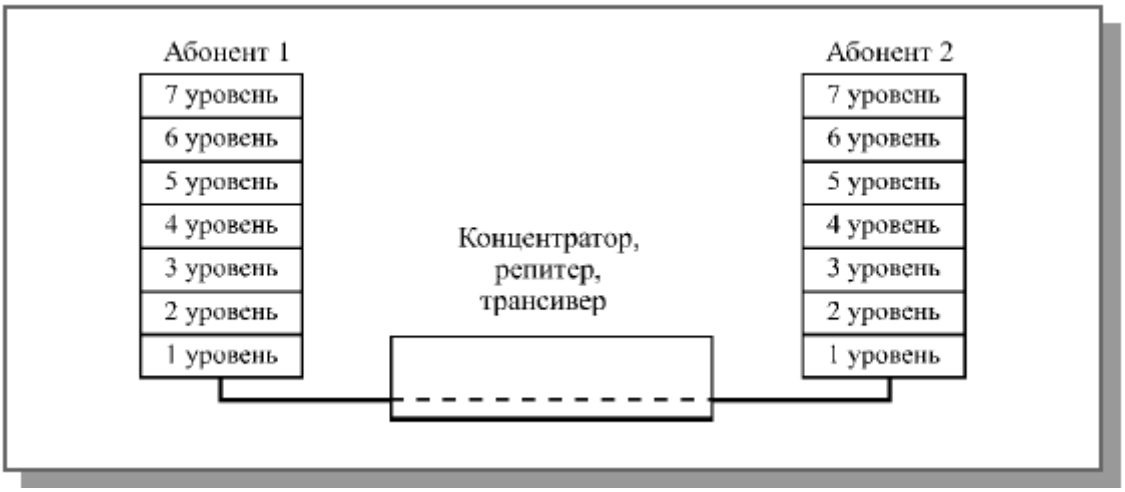


Рис. 5.9. Функции концентраторов, репитеров и трансиверов в модели OSI

Выпускаются также совсем простые концентраторы, которые соединяют сегменты сети без восстановления формы сигналов. Они не увеличивают длину сети.

Коммутаторы (свичи, коммутирующие концентраторы, switch), как и концентраторы, служат для соединения сегментов в сеть. Они также выполняют более сложные функции, производя сортировку поступающих на них пакетов.

Коммутаторы передают из одного сегмента сети в другой не все поступающие на них пакеты, а только те, которые адресованы компьютерам из другого сегмента. Пакеты, передаваемые между абонентами одного сегмента, через коммутатор не проходят. При этом сам пакет коммутатором не принимается, а только пересылается. Интенсивность обмена в сети снижается вследствие разделения нагрузки, поскольку каждый сегмент работает не только со своими пакетами, но и с пакетами, пришедшими из других сегментов.

Коммутатор работает на втором уровне модели OSI (подуровень MAC), так как анализирует MAC-адреса внутри пакета (рис. 5.10). Естественно, он выполняет и функции первого уровня.

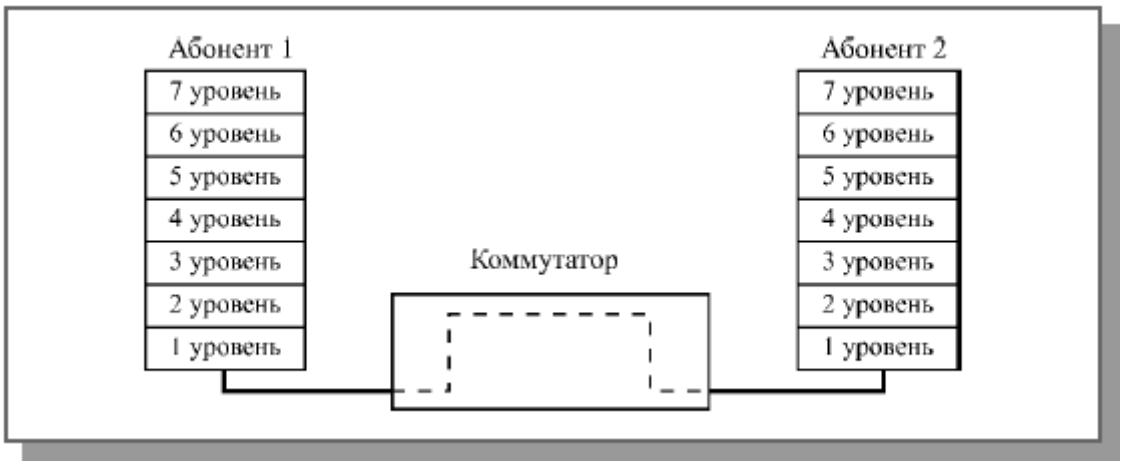


Рис. 5.10. Функции коммутаторов в модели OSI

В последнее время объем выпуска коммутаторов сильно вырос, цена на них упала, поэтому коммутаторы постепенно вытесняют концентраторы.

Мосты (bridge), маршрутизаторы (router) и шлюзы (gateway) служат для объединения в одну сеть нескольких разнородных сетей с разными протоколами обмена нижнего уровня, в частности, с разными форматами пакетов, методами кодирования, скоростью передачи и т.д. В результате их применения сложная и неоднородная сеть, содержащая в себе различные сегменты, с точки зрения пользователя выглядит самой обычной сетью. Обеспечивается прозрачность сети для протоколов высокого уровня. Все они гораздо дороже, чем концентраторы, так как от них требуется довольно сложная обработка информации. Реализуются они обычно на базе компьютеров, подключенных к сети с помощью сетевых адаптеров. По сути, они представляют собой специализированные абоненты (узлы) сети.

Мосты – наиболее простые устройства, служащие для объединения сетей с разными стандартами обмена, например, Ethernet и Arcnet, или нескольких сегментов (частей) одной и той же сети, например, Ethernet (рис. 5.11). В последнем случае мост, как и коммутатор, только разделяет нагрузку сегментов, повышая тем самым производительность сети в целом. В отличие от коммутаторов мосты принимают поступающие пакеты целиком и в случае необходимости производят их простейшую обработку. Мосты, как и коммутаторы, работают на втором уровне модели OSI (рис. 5.10), но в отличие от них могут захватывать также и верхний подуровень LLC второго уровня (для связи разнородных сетей). В последнее время мосты быстро вытесняются коммутаторами, которые становятся более функциональными.

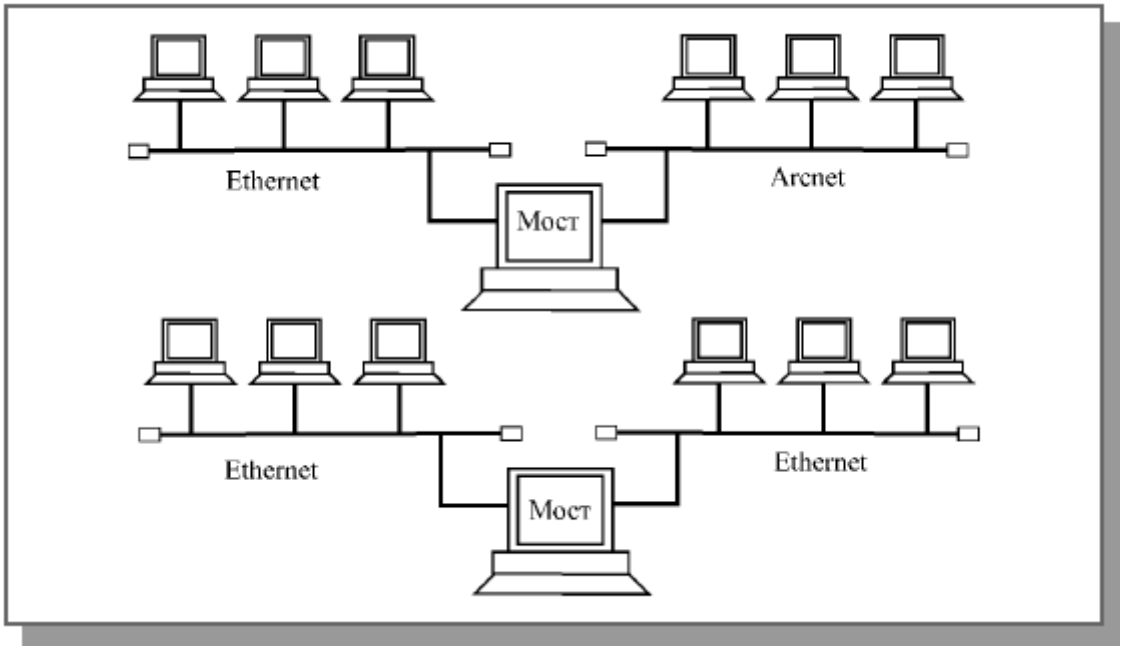


Рис. 5.11. Включение моста

Маршрутизаторы осуществляют выбор оптимального маршрута для каждого пакета с целью избежания чрезмерной нагрузки отдельных участков

сети и обхода поврежденных участков. Они применяются, как правило, в сложных разветвленных сетях, имеющих несколько маршрутов между отдельными абонентами. Маршрутизаторы не преобразуют протоколы нижних уровней, поэтому они соединяют только сегменты одноименных сетей.

Маршрутизаторы работают на третьем уровне модели OSI, так как они анализируют не только MAC-адреса пакета, но и IP-адреса, то есть более глубоко проникают в инкапсулированный пакет (рис. 5.12).

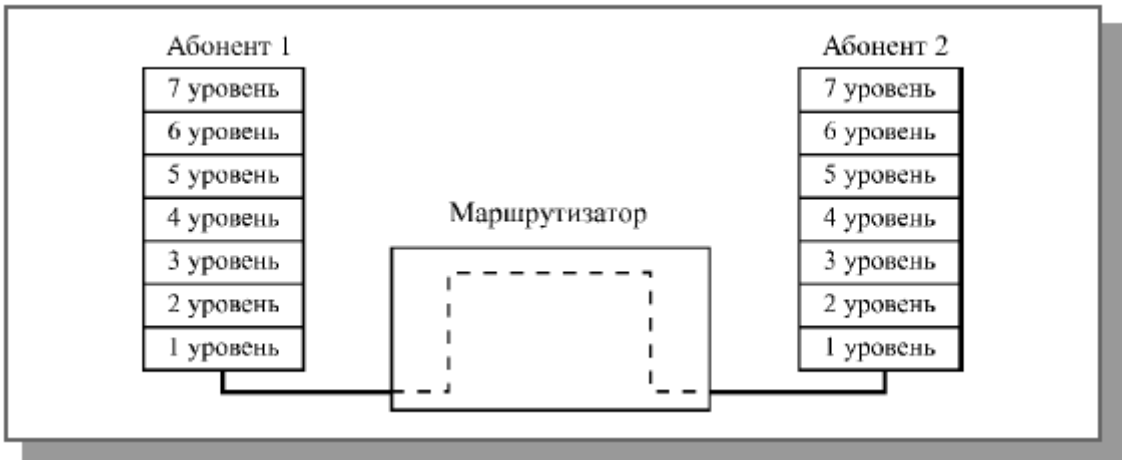


Рис. 5.12. Функции маршрутизатора в модели OSI

Существуют также гибридныемаршрутизаторы (brouter), представляющие собой гибрид моста и маршрутизатора. Они выделяют пакеты, которым нужна маршрутизация и обрабатывают их как маршрутизатор, а для остальных пакетов служат обычным мостом.

Шлюзы – это устройства для соединения сетей с сильно отличающимися протоколами, например, для соединения локальных сетей с большими компьютерами или с глобальными сетями. Это самые дорогие и редко применяемые сетевые устройства. Шлюзы реализуют связь между абонентами на верхних уровнях модели OSI (с четвертого по седьмой). Соответственно, они должны выполнять и все функции нижестоящих уровней.

Подробнее промежуточные сетевые устройства будут рассмотрены в разделах, посвященных конкретным стандартным локальным сетям.

2. ПОНЯТИЕ СЕТЕВОЙ АТАКИ. ВИДЫ СЕТЕВЫХ АТАК

Сетевая атака - действие, целью которого является захват контроля (повышение прав) над удалённой/локальной вычислительной системой, либо её дестабилизация, либо отказ в обслуживании, а также получение данных пользователей пользующихся этой удалённой/локальной вычислительной системой.

На данный момент выделяют следующие атаки: mailbombing, переполнение буфера, использование специализированных программ (вирусов, снифферов, троянских коней, почтовых червей, rootkit-ов и т.д.), сетевая

разведка, IP-спуфинг, man-in-the-middle, инъекция (SQL-инъекция, PHP-инъекция, межсайтовый скриптинг или XSS-атака, XPath-инъекция), отказ в обслуживании (DoS- и DDoS- атаки), phishing-атаки. Рассмотрим каждую из них.

Mailbombing.

Суть данной атаки заключается в том, что на почтовый ящик посылается огромное количество писем на почтовый ящик пользователя. Эта атака может вызвать отказ работы почтового ящика или даже целого почтового сервера. Данная атака может проводиться любым хотя бы немного подготовленным противником. Простым примером программы, с помощью которой можно осуществить подобную атаку- The Unabomber. Достаточно знать адрес сервера, позволяющего анонимно отправлять почтовые сообщения, и адрес пользователя, которому эти сообщения предназначены. Количество писем, которое можно отослать для этой программы равно 12 разрядному числу. Рассмотрим способы защиты от данной атаки.

1. Давать адрес электронной почты только проверенным источникам.
2. В качестве преграды для mailbombing-а может выступать и Web-сайт провайдера, иногда настраиваемый таким образом, чтобы он автоматически определял почтовые атаки. В большинстве случаев они распознаются сервером посредством сравнения исходных IP-адресов входящих сообщений. Если количество сообщений из одного источника превышает некие разумные пределы, то все они автоматически поступают в Recycle Bin на сервере. Конечно же, ничто не мешает злоумышленнику фальсифицировать собственный IP-адрес.

Обычно к таким атакам опытные злоумышленники прибегают крайне редко.

Переполнение буфера(buffer overflows).

Атака на переполнение буфера основывается на поиске программных или системных уязвимостей, способных вызвать нарушение границ памяти и аварийно завершить приложение или выполнить произвольный бинарный код от имени пользователя, под которым работала уязвимая программа. Если программа работает под учетной записью администратора, то данная атака может позволить получить полный контроль над компьютером, на котором исполняется данная программа.

Реализации атаки требует решения двух подзадач:

1. Подготовка кода, который будет выполняться в контексте привилегированной программы.
2. Изменение последовательности выполнения программы с передачей управления подготовленному коду.

Классификация атак по переполнению буфера представлена в таблице:

Подготовка кода Цель переполнения	Внедрение кода	Внедрение параметров	Не требуется
Искажение адреса возврата из функции	Атака “срыв стека”	Атака “срыв стека” с параметризацией	Атака “срыв стека” с передачей управления
Искажение указателей функций	Атака на указатели функций	Атака на указатели функций с параметризацией	Атака на указатели функций с передачей управления
Искажение таблиц переходов	Атака на таблицы переходов	Атака на таблицы переходов с параметризацией	Атака на таблицы переходов с передачей управления
Искажение указателей данных	Атака с искажением указателей данных	Атака с искажением указателей данных с параметризацией	Атака с искажением указателей данных с оригинальным кодом

Исходя из подзадач, реализацию которых требует атака, выделяют следующие способы борьбы с атаками подобного типа:

1. Корректировка исходных кодов программы для устранения уязвимостей. Переполнение буфера происходит, прежде всего, из-за неправильного алгоритма работы программы, который не предусматривает проверок выхода за границы буферов. Также возможно применение специальных утилит автоматического поиска уязвимостей в исходном коде программы. Указанные методы и средства позволяют создавать более защищенные программы, но не решают проблему в принципе, а лишь минимизируют число уязвимостей по переполнению буфера. Данный подход ориентирован непосредственно на разработчиков программного обеспечения и не является инструментом конечного пользователя или системного администратора.

2. Использование неисполнимых буферов. Суть метода заключается в запрещении исполнения кода в сегментах данных и стека, т.е. параметры сегментов данных и стека содержат только атрибуты записи и чтения, но не

исполнения. Однако ограничение на исполнение данных приводит к проблеме несовместимости. Исполняемый стек необходим для работы многим программам, так как на его основе генерируется код компиляторами, реализуются системные функции операционных систем, реализуется автоматическая генерация кода. Защита с использованием неисполнимых буферов предотвратит только атаки с внедрением кода, но не поможет при других видах атак.

3. Применение проверок выхода за границы. В основе данного метода лежит выполнение проверок выхода за границы переменной при каждом обращении к ней. Это предотвращает все возможные атаки по переполнению буфера, так как полностью исключает само переполнение. Однако, у этого решения есть существенный недостаток - значительное (до 30 раз) снижение производительности программы.

4. Применение проверок целостности. Решение, основанное на данном методе, получено благодаря проекту Synthetix. Цель Synthetix - специализация кода для увеличения производительности операционных систем. При этом вводится понятие так называемого квази-постоянства (Quasi-invariant), т.е. состояния среды, которое неизменно в определенных рамках. Такое квази-постоянство позволяет устранить ряд избыточного кода проверки выполнения различных условий. В рамках проекта реализован набор утилит, в том числе обеспечивающих контроль и защиту квази-постоянных состояний среды. К их числу относятся StackGuard и PointGuard.

Использование специализированных программ.

Рабочие станции конечных пользователей очень уязвимы для вирусов и троянских коней. Вирусами называются вредоносные программы, которые внедряются в другие программы для выполнения определенной нежелательной функции на рабочей станции конечного пользователя. В качестве примера можно привести вирус, который прописывается в файле command.com (главном интерпретаторе систем Windows) и стирает другие файлы, а также заражает все другие найденные им версии command.com.

"Троянский конь" - это не программная вставка, а настоящая программа, которая выглядит как полезное приложение, а на деле выполняет вредную роль. Примером типичного "троянского коня" является программа, которая выглядит, как простая игра для рабочей станции пользователя. Однако пока пользователь играет в игру, программа отправляет свою копию по электронной почте каждому абоненту, занесенному в адресную книгу этого пользователя. Все абоненты получают по почте игру, вызывая ее дальнейшее распространение.

Сниффер пакетов представляет собой прикладную программу, которая использует сетевую карту, работающую в режиме promiscuous mode (в этом режиме все пакеты, полученные по физическим каналам, сетевой адаптер отправляет приложению для обработки). При этом сниффер перехватывает все сетевые пакеты, которые передаются через определенный домен. В настоящее время снифферы работают в сетях на вполне законном основании. Они используются для диагностики неисправностей и анализа трафика. Однако

ввиду того, что некоторые сетевые приложения передают данные в текстовом формате (telnet, FTP, SMTP, POP3 и т.д.), с помощью сниффера можно узнать полезную, а иногда и конфиденциальную информацию (например, имена пользователей и пароли).

Перехват имен и паролей создает большую опасность, так как пользователи часто применяют один и тот же логин и пароль для множества приложений и систем. Многие пользователи вообще имеют один пароль для доступа ко всем ресурсам и приложениям. Если приложение работает в режиме клиент/сервер, а аутентификационные данные передаются по сети в читаемом текстовом формате, эту информацию с большой вероятностью можно использовать для доступа к другим корпоративным или внешним ресурсам.

Rootkit - программа или набор программ для скрытия следов присутствия злоумышленника или вредоносной программы в системе. Большинство из реализаций современных rootkit могут прятать от пользователя файлы, папки и ключи реестра, скрывать запущенные программы, системные службы, драйверы и сетевые соединения. Т.е. злоумышленник имеет возможность создавать файлы и ключи реестра, запускать программы, работать с сетью и эта активность не будет обнаружена администратором. Кроме того, rootkits могут скрывать сетевую активность путем модификации стека протоколов TCP/IP. Так, например rootkit Hacker Defender перехватывает вызовы Winsock и может обрабатывать сетевой трафик до того как он будет передан приложению. Т.е. если в системе установлен Web сервер, и соответственно открыт 80й порт, rootkit может использовать его для взаимодействия с взломщиком, в то время как другие пользователи будут без проблем работать по протоколу HTTP.

Выделяют следующие способы борьбы с этими видами атак:

1. Использование антивирусных средств и регулярное обновление их сигнатур. Может решить проблему с троянскими программами, вирусами, почтовыми червями, но не решит проблему снифферов и rootkit-ов.
2. Шифрование передаваемых данных. Проблема не решает полностью проблему снифферов, однако, противник перехватывает данные, которые нельзя свободно прочитать. Для их расшифровки требуется время.
3. Использование антиснифферов (Например, AntiSniff или PromiScan).
4. Использование межсетевых экранов.
5. Использование антируткитов.

Сетевая разведка.

Сетевой разведкой называется сбор информации о сети с помощью общедоступных данных и приложений. При подготовке атаки против какой-либо сети злоумышленник, как правило, пытается получить о ней как можно больше информации. Сетевая разведка проводится в форме запросов DNS, эхо-тестирования (ping sweep) и сканирования портов. Запросы DNS помогают понять, кто владеет тем или иным доменом и какие адреса этому домену присвоены. Эхо-тестирование (ping sweep) адресов, раскрытых с помощью DNS, позволяет увидеть, какие хосты реально работают в данной среде.

Получив список хостов, злоумышленник использует средства сканирования портов, чтобы составить полный список услуг, поддерживаемых этими хостами. И, наконец, злоумышленник анализирует характеристики приложений, работающих на хостах. В результате добывается информация, которую можно использовать для взлома.

Способы борьбы с данной атакой:

1. Отключение эхо ICMP и эхо-ответ на периферийных маршрутизаторах. Это, однако, приведет к потере данных необходимых для диагностики сетевых сбоев.
2. Использование систем обнаружения вторжений (IDS).

IP-спуфинг.

IP-спуфинг происходит, когда злоумышленник, находящийся внутри корпорации или вне ее выдает себя за санкционированного пользователя. Это можно сделать двумя способами. Во-первых, злоумышленник может воспользоваться IP-адресом, находящимся в пределах диапазона санкционированных IP-адресов, или авторизованным внешним адресом, которому разрешается доступ к определенным сетевым ресурсам. Атаки IP-спуфинга часто являются отправной точкой для прочих атак. Классический пример - атака DoS, которая начинается с чужого адреса, скрывающего истинную личность злоумышленника.

Обычно IP-спуфинг ограничивается вставкой ложной информации или вредоносных команд в обычный поток данных, передаваемых между клиентским и серверным приложением или по каналу связи между одноранговыми устройствами. Для двусторонней связи злоумышленник должен изменить все таблицы маршрутизации, чтобы направить трафик на ложный IP-адрес. Некоторые злоумышленники, однако, даже не пытаются получить ответ от приложений. Если главная задача состоит в получении от системы важного файла, ответы приложений не имеют значения.

Если же злоумышленнику удается поменять таблицы маршрутизации и направить трафик наложный IP-адрес, злоумышленник получит все пакеты и сможет отвечать на них так, будто он является санкционированным пользователем.

Угрозу спуфинга можно ослабить (но не устранить) с помощью следующих мер:

1. Контроль доступа. Самый простой способ предотвращения IP-спуфинга состоит в правильной настройке управления доступом. Чтобы снизить эффективность IP-спуфигна, настройте контроль доступа на отсечение любого трафика, поступающего из внешней сети с исходным адресом, который должен располагаться внутри вашей сети. Если санкционированными являются и некоторые адреса внешней сети, данный метод становится неэффективным.
2. Фильтрация RFC 2827. Вы можете пресечь попытки спуфинга чужих сетей пользователями вашей сети (и стать добропорядочным "сетевым гражданином"). Для этого необходимо отбраковывать любой исходящий трафик, исходный адрес которого не является одним из IP-адресов вашей

организации. Этот тип фильтрации, известный под названием "RFC 2827", может выполнять и провайдер. В результате отбраковывается весь трафик, который не имеет исходного адреса, ожидаемого на определенном интерфейсе.

3. Использование криптографической аутентификации.

Атакатипа man-in-the-middle.

Для атаки типа Man-in-the-Middle злоумышленнику нужен доступ к пакетам, передаваемым по сети. Такой доступ ко всем пакетам, передаваемым от провайдера в любую другую сеть, может, к примеру, получить сотрудник этого провайдера. Для атак этого типа часто используются снифферы пакетов, транспортные протоколы и протоколы маршрутизации. Атаки проводятся с целью кражи информации, перехвата текущей сессии и получения доступа к частным сетевым ресурсам, для анализа трафика и получения информации о сети и ее пользователях, для проведения атак типа DoS, искажения передаваемых данных и ввода несанкционированной информации в сетевые сессии.

Способы борьбы с данной атакой:

1. Использование шифрования данных

Инъекция.

SQL-инъекция.

SQL-инъекция – это атака, в ходе которой изменяются параметры SQL-запросов к базе данных. В результате запрос приобретает совершенно иной смысл, и в случае недостаточной фильтрации входных данных способен не только произвести вывод конфиденциальной информации, но и изменить/удалить данные.

Способы защиты от данной атаки (используются исключительно администраторами ресурсов):

1. Для целых и дробных величин, перед их использованием в запросе достаточно привести величину к нужному типу.

$id=(int)$id; $total=(float)$total;

Вместо этого можно вставить систему слежения за тестированием на SQL инъекцию.

```
if((string)$id<>(string)(int)$id) {
die('ops');
}
```

2. Для строковых параметров, которые не используются в like, regexp и тд, экранируем кавычки.

$str=addslashes($str);

или, лучше,

mysql_escape_string($str)

3. В строках, которые предполагается использовать внутри like, regexp и тд, необходимо так же заэкранировать специальные символы, применяющиеся в этих операторах, если это необходимо. В противном случае, можно задокументировать использование этих символов.

PHP-инъекция.

PHP-инъекция - один из способов взлома веб-сайтов, работающих на PHP. Он заключается в том, чтобы внедрить специально сформированный злонамеренный сценарий в код веб-приложения на серверной стороне сайта, что приводит к выполнению произвольных команд.

Способы борьбы с данной атакой (используются исключительно администраторами ресурсов):

1. Проверять, не содержит ли переменная $name посторонние символы:

```
<?
...
$name = $_GET['name'];
if (strpbrk($name, '.?/:')) die('Blocked');
include $name. '.php';
...
?>
```

2. Проверять, что $name присвоено одно из допустимых значений:

```
<?
...
$name = $_GET['name'];
$arr = array('main', 'about', 'links', 'forum');
if (!in_array($name,$arr)) $name = $arr[0];
include $name . '.php';
...
?>
```

Межсайтовыйскриптингили XSS-атака.

XSS атака – это атака на уязвимость, которая существует на сервере, позволяющая внедрить в генерируемую сервером HTML-страницу некий произвольный код, в котором может быть вообще все что угодно и передавать этот код в качестве значения переменной, фильтрация по которой не работает, то есть сервер не проверяет данную переменную на наличие в ней запрещенных знаков –, <, >, ’, ”. Значение данной переменной передается от генерируемой HTML-страницы на сервер в скрипт, ее вызвавший путем отправки запроса.

А далее начинается самое интересное для Злоумышленника. PHP-скрипт в ответ на данный запрос генерирует HTML-страницу, в которой отображаются значения требующихся злоумышленнику переменных, и отправляет данную страницу на браузер злоумышленника.

То есть, говоря проще, XSS атака – это атака с помощью уязвимостей на сервере на компьютеры клиентов.

XSS атака чаще всего используется для кражи Cookies (или куки, как их произносят по-русски). В них хранится информация о сессии пребывания пользователя на сайтах, что и бывает нужным злоумышленникам для перехвата управления личными данными пользователя на сайте в пределах, пока сессия не будет закрыта сервером, на котором размещен сайт. Помимо этого в Cookies хранится зашифрованный пароль, под которым пользователь входит на данный

сайт, и при наличии необходимых утилит и желания злоумышленникам не доставляет особого труда расшифровать данный пароль.

Теперь опишем другие возможности XSS атак (конечно при условии их успешного проведения).

1. Возможно при открытии страницы вызвать открытие большого количества ненужных пользователю окон.
2. Возможна вообще переадресация на другой сайт (например, на сайт конкурента).
3. Существует возможность загрузки на компьютер пользователя скрипта с произвольным кодом (даже вредоносного) путем внедрения ссылки на исполняемый скрипт со стороннего сервера.
4. Зачастую происходит кража личной информации с компьютера пользователя, помимо Cookies в качестве объекта кражи выступает информация о посещенных сайтах, о версии браузера и операционной системе, установленной на компьютере пользователя, да к тому же еще и плюсуется IP-адрес компьютера пользователя.
5. XSS атака может быть проведена не только через сайт, но и через уязвимости в используемом программном обеспечении (в частности, через браузеры). Поэтому рекомендуется обновлять используемое программное обеспечение.
6. Также возможно проведение XSS атак через использование SQL-кода.

Как мы видим из всего вышесказанного, возможностей у XSS атак достаточно много. Злоумышленник может овладеть вашей личной информацией вплоть до получения паролей доступа к сайтам, а это очень неприятно. К тому же XSS атака наносит вред исключительно клиентским машинам, оставляя сервер в полностью рабочем состоянии, и у администрации различных серверов порой мало стимулов устанавливать защиту от этого вида атак.

Различают XSS атаки двух видов: активные и пассивные. При первом виде атаки вредоносный скрипт хранится на сервере и начинает свою деятельность при загрузке страницы сайта в браузере клиента. При втором виде атак скрипт не хранится на сервере и вредоносное действие начинает выполняться только в случае какого-либо действия пользователя, например, при нажатии на сформированную ссылку.

Способы борьбы с данным видом атак (используются исключительно администраторами ресурсов):

1. Запретить включение напрямую параметров $_GET, $_POST, $_COOKIE в генерируемую HTML-страницу.
2. Запретить загрузку произвольных файлов на сервер во избежание загрузки вредоносных скриптов.
3. Все загруженные файлы хранить в базе данных, а не в файловой системе.

XPath-инъекция.

XPath-инъекция - вид уязвимостей, который заключается во внедрении XPath-выражений в оригинальный запрос к базе данных XML. XPath (XML Path Language) – это язык, который предназначен для произвольного обращения к частям XML документа. XML (eXtensible Markup Language) – это всем известный язык разметки, с помощью которого создаются XML документы, имеющие древовидную структуру. [6]

Способы борьбы с этой атакой (используются исключительно администраторами ресурсов):

1. Проверка корректности. Независимо от приложения, среды или языка необходимо следовать следующим практическим правилам:

- Предполагайте, что все вводимые данные сомнительны.
- Проверяйте не только тип вводимых данных, но также их формат, длину, диапазон и содержимое (например, такое простое регулярное выражение как if (/^"*^';&<>()/) должно находить большинство подозрительных специальных символов).
- Проверяйте данные как на стороне клиента, так и на стороне сервера, поскольку проверку на стороне клиента чрезвычайно легко перехитрить.
- Следуйте последовательной [missing word] стратегии защищенности приложения, основываясь на передовом опыте разработки защищенных приложений
- Тестируйте приложение на известные угрозы перед его выпуском.

2. Проверка данных на Web-сервере. Для защиты против XPath-внедрения и других форм внедрения кода необходимо проверять все данные, передаваемые от Web-сервера к службам системы хранения данных. Этот подход может быть очень хорош для некоторых приложений, которые, возможно, используют основанные на REST или SOAP XML-сервисы, но в других случаях он может быть не возможен. Как всегда наилучшим подходом является разумное проектирование, начиная с первоначального дизайна и до реализации приложения.

Отказ в обслуживании (DoS- и DDoS- атаки).

DoS, без всякого сомнения, является наиболее известной формой атак. Кроме того, против атак такого типа труднее всего создать стопроцентную защиту. Даже среди злоумышленников атаки DoS считаются тривиальными, а их применение вызывает презрительные усмешки, потому что для организации DoS требуется минимум знаний и умений. Тем не менее, именно простота реализации и огромный причиняемый вред привлекают к DoS пристальное внимание администраторов, отвечающих за сетевую безопасность.

Атаки DoS отличаются от атак других типов. Они не нацелены на получение доступа к вашей сети или на получение из этой сети какой-либо информации. Атака DoS делает вашу сеть недоступной для обычного использования за счет превышения допустимых пределов функционирования сети, операционной системы или приложения.

В случае использования некоторых серверных приложений (таких как Web-сервер или FTP-сервер) атаки DoS могут заключаться в том, чтобы занять

все соединения, доступные для этих приложений и держать их в занятом состоянии, не допуская обслуживания обычных пользователей. В ходе атак DoS могут использоваться обычные Интернет-протоколы, такие как TCP и ICMP (Internet Control Message Protocol). Большинство атак DoS опирается не на программные ошибки или бреши в системе безопасности, а на общие слабости системной архитектуры. Некоторые атаки сводят к нулю производительность сети, переполняя ее нежелательными и ненужными пакетами или сообщая ложную информацию о текущем состоянии сетевых ресурсов. Этот тип атак трудно предотвратить, так как для этого требуется координация действий с провайдером. Если трафик, предназначенный для переполнения вашей сети, не остановить у провайдера, то на входе в сеть вы это сделать уже не сможете, потому что вся полоса пропускания будет занята. Когда атака этого типа проводится одновременно через множество устройств, мы говорим о распределенной атаке DoS (DDoS - distributed DoS).

Угроза атак типа DoS может снижаться тремя способами:

1. Функции анти-спуфинга. Правильная конфигурация функций анти-спуфинга на маршрутизаторах и межсетевых экранах поможет снизить риск DoS. Эти функции, как минимум, должны включать фильтрацию RFC 2827.
2. Функции анти-DoS. Правильная конфигурация функций анти-DoS на маршрутизаторах и межсетевых экранах может ограничить эффективность атак. Эти функции часто ограничивают число полуоткрытых каналов в любой момент времени.
3. Ограничение объема трафика (traffic rate limiting). Организация может попросить провайдера ограничить объем трафика. Этот тип фильтрации позволяет ограничить объем некритического трафика, проходящего по вашей сети.

Phishing-атаки.

Phishing (фишинг) - процесс обмана или социальная разработка клиентов организаций для последующего воровства их идентификационных данных и передачи их конфиденциальной информации для преступного использования. Преступники для своего нападения используют spam или компьютеры-боты. При этом размер компании-жертвы не имеет значения; качество личной информации полученной преступниками в результате нападения, имеет значение само по себе.

Приведем пример фишинг-атаки:

Пользователь получает электронную почту otsupport@mybank.com <mailto:support@mybank.com> (адрес - подменен) со строкой сообщения "модификация защиты", в котором ее просят перейти по адресу www.mybank-validate.info <http://www.mybank-validate.info> (имя домена принадлежит нападавшему, а не банку) и ввести его банковский PIN-код.

Способы защиты от данной атаки:

1. Использовать только проверенные ресурсы и пути доступа к ним.
2. Использовать антивирусные средства и регулярно обновлять их сигнатуры.

3. ПОНЯТИЕ МЕЖСЕТЕВОГО ЭКРАНА

Межсетевой экран (другие названия брандмауэр, фаервол) это защитный барьер между вашим компьютером и сетью к которой он подключен. Когда вы заходите в интернет, то ваш компьютер становиться видимыми для внешнего мира. Вы видимы через нечто, что называется порт. Порт это идентифицируемый определенным номером системный ресурс, выделяемый приложению, которое выполняется на некотором сетевом хосте (компьютер или другое сетевое устройство), для связи с приложениями, которые выполняются на других сетевых хостах (в том числе с другими приложениями на этом же хосте). Есть много тысяч таких портов и, как было сказано выше, у каждого есть свой уникальный номер.

Наиболее часто используемыми портами во всемирной сети являются:

1. 80 — порт для загрузки web-страниц;
2. 110 — используется по умолчанию для загрузки электронной почты;
3. 25 — используется по умолчанию для отправки электронной почты.

Суть брандмауэра в том, чтобы **закрыть порты**, которые вы не используете. В противном случае через них злоумышленник или вредоносная программа (вирус, троян) могут проникнуть на ваш ПК. Если вы подключены к сети интернет, то вы просто обязаны иметь межсетевой экран.

Если после установки операционной системы вы не активируете брандмауэр или если он не будет активирован по умолчанию, то ваша система может быть атакована спустя несколько минут после выхода в интернет. Через открытые порты могут проникнуть вирусы, троянские черви и шпионские программы, которые принесут вам немало неприятностей в будущем, а вы об этом даже не будете догадываться. Однажны, авторитетное за границей издание PC Format запустило эксперимент и в результате операционная система на их абсолютно незащищенном компьютере была приведена в полную непригодность спустя два с половиной часа серфинга по интернету. Межсетевой экран мог бы остановить часть атак, которым подвергся компьютер.

Не стоит также считать, что наличие только фаервола поможет уберечься от всех бед. Не стоит забывать об установке антивируса с последующим регулярным обновлением баз, а также о регулярном скачивании и установке обновлений безопасности для вашей Windows.

Большинство межсетевых экранов, включая встроенные в Windows, будет оповещать вас о подозрительном входящем трафике. Но хороший брандмауэр должен оповещать и о подозрительном исходящем трафике. Например встроенный брандмауэр Windows XP не умеет это делать, по этому его лучше заменить на программу от стороннего производителя. Наличие подозрительного исходящего трафика поможет вам понять, что ваш компьютер уже заражен троянским или шпионским ПО.

Брандмауэры можно поделить на две простые категории: аппаратные и программные. Аппаратным фаерволом может быть маршрутизатор, который находиться между вашим ПК и сетью Интернет. В таком случае к нему можно подключить несколько компьютеров и все они будут защищены брандмауэром, который является частью маршрутизатора. Программный межсетевой экран — это специализированное ПО, которое пользователь устанавливает себе на компьютер.

Даже если у вас уже есть маршрутизатор со встроенным межсетевым экраном, вы можете также установить программный фаервол на каждый компьютер в отдельности. Тогда злоумышленнику будет значительно тяжелее проникнуть в вашу систему.

4. ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ КОМПЬЮТЕРНЫХ СЕТЕЙ

Рассмотрим некоторые базовые понятия, относящиеся к изучаемой предметной области.

Теория защиты информации – система основных идей, относящихся к защите информации, дающая целостное представление о сущности проблемы защиты, закономерностях ее развития и существенных связях с другими отраслями знания, формирующаяся на основе опыта практического решения задач защиты и определяющая основные ориентиры в направлении совершенствования практики защиты информации.

Защита информации – это комплекс мероприятий, проводимых с целью предотвращения утечки, хищения, утраты, несанкционированного уничтожения, искажения, модификации (подделки), несанкционированного копирования, блокирования информации.

Следует заметить, что наряду с термином «защита информации» применительно к компьютерным сетям широко используется,как правило, в близком значении, термин «компьютерная безопасность».

Компьютерная безопасность – одна из основных задач, решаемых любой компьютерной сетью. Проблему безопасности можно рассматривать с разных сторон – злонамеренная порча данных, конфиденциальность информации, несанкционированный доступ, хищения и т. п.

Надежность компьютерной сети – характеристика способности ее аппаратного, программного и программно-аппаратного обеспечения выполнить при определенных условиях требуемые функции в течение определенного периода времени. Повышение надежности основано на принципе предотвращения неисправностей путем снижения интенсивности отказов и сбоев за счет применения электронных схем и компонентов с высокой и сверхвысокой степенью интеграции, снижения уровня помех, облегченных режимов работы схем, обеспечения тепловых режимов их работы, а также за счет совершенствования методов сборки аппаратуры.

Главной целью повышения надежности систем является обеспечение целостности хранимых в них данных.

Отказоустойчивость – это такое свойство вычислительной системы, которое обеспечивает ей как логической машине возможность продолжения действий, заданных программой, после возникновения неисправностей. Введение отказоустойчивости требует избыточного аппаратного и программного обеспечения.

Секретность (**конфиденциальность**) информации – свойство информации быть известной только допущенным и прошедшим авторизацию субъектам системы (пользователям, программам, процессам и др.); статус, предоставленный информации и определяющий требуемую степень ее защиты.

Субъект – активный компонент системы, который может инициировать поток информации или изменить состояние системы.

Объект – пассивный компонент системы, хранящий, перерабатывающий, передающий или принимающий информацию (например, страницы, файлы, папки, директории, компьютерные программы, устройства и т. д.).

Доступ – специальный тип взаимодействия между объектом и субъектом, в результате которого создается поток информации от одного к другому.

Санкционированный доступ (СД) к информации – это доступ к информации, не нарушающий установленные правила разграничения доступа.

Несанкционированный доступ (НСД) к информации характеризуется нарушением установленных правил разграничения доступа. Несанкционированный доступ является наиболее распространенным видом компьютерных нарушений. НСД использует любую ошибку в системе защиты и возможен при нерациональном выборе средств защиты, их некорректной установке и настройке.

Пассивное вторжение (**перехват информации**) характеризуется тем, что нарушитель только наблюдает за прохождением информации по каналу связи, не вторгаясь ни в информационный поток, ни в содержание передаваемой информации.

Активное вторжение характеризуется стремлением нарушителя подменить информацию, передаваемую в сообщении. Он может выборочно модифицировать или добавлять правильное или ложное сообщение, удалять, задерживать или изменять порядок Методы обеспечения надежности компьютерных сетей следования сообщений, а также аннулировать или задерживать все сообщения, передаваемые по каналу.

Удаленная атака – информационное разрушающее воздействие на распределенную компьютерную сеть, программно осуществленное по каналам связи.

Интруз – физическое лицо или процесс, которые реализуют неразрешенный, или несанкционированный, доступ к информации, т. е. **атаку** на систему.

Авторизация – предоставление субъектам доступ к объектам системы. Доступ к объекту означает доступ к содержащейся в нем информации.

Аутентификация – проверка идентификации пользователя, устройства или другого компонента в системе (обычно для принятия решения о разрешении доступа к ресурсам системы). Частным вариантом аутентификации является установление принадлежности сообщения конкретному автору.

Целостность – состояние данных или компьютерной системы, в которой данные и программы используются установленным способом, обеспечивающим устойчивую работу системы и единство данных.

Безопасная (защищенная) система – система со средствами защиты, которые успешно и эффективно противостоят угроза безопасности (возможным действиям, которые прямо или косвенно могут нанести ущерб системе).

Введение в проблему безопасности компьютерных сетей. В компьютерных сетях особую опасность представляют собой атаки на локальную сеть через подключение к сети Интернет для того, чтобы получить доступ к конфиденциальной информации, что связано с недостатками встроенной системы защиты информации в протоколах TCP/IP.

До сих пор не существует единой и общепринятой классификации угроз безопасности компьютерных сетей.

Сетевые атаки столь же разнообразны, как и системы, против которых они направлены. Некоторые атаки отличаются большой сложностью. Другие может осуществить обычный оператор, даже не предполагающий, какие последствия может иметь его деятельность. Для оценки типов атак необходимо знать некоторые ограничения, изначально присущие протоколу TPC/IP.

Ввиду того, что изначально средства защиты для протокола IP не разрабатывались, все его реализации стали дополняться разнообразными сетевыми процедурами, услугами и продуктами, снижающими риски, присущие этому протоколу. Далее будут кратко рассмотрены основные типы атак, обычно применяемых против сетей IP, и перечислены способы борьбы с ними.

Сниффер пакетов (sniffer – в данном случае фильтрация) – прикладная программа, которая использует сетевую карту, работающую в режиме (promiscuous («не делающий различия») mode), в котором все пакеты, полученные по физическим каналам, сетевой адаптер отправляет приложению для обработки.

Сниффер перехватывает все сетевые пакеты, которые передаются через атакуемый домен. В настоящее время снифферы работают в сетях на вполне законном основании. Они используются для диагностики неисправностей и анализа трафика. Однако ввиду того, что некоторые сетевые приложения передают данные в текстовом формате (Telnet, FTP, SMTP, POP3 и т. д.), с помощью сниффера можно узнать полезную, а иногда и конфиденциальную информацию (например, имена пользователей и пароли).

Перехват имен и паролей создает большую опасность, так как пользователи часто применяют один и тот же логин и пароль для множества приложений и систем. Многие пользователи вообще имеют один пароль для доступа ко всем ресурсам и приложениям. Если приложение работает в режиме

клиент – сервер, а аутентификационные данные передаются по сети в читаемом текстовом формате, эту информацию с большой вероятностью можно использовать для доступа к другим корпоративным или внешним ресурсам. Таким образом, человек, конечный пользователь, оказывается самым слабым звеном системы информационной безопасности, и хакеры, зная это, умело применяют методы социальной инженерии.

Социальная инженерия – это использование хакером психологических приемов «работы» с пользователем. В самом худшем случае хакер, перехватив пароль, получает доступ к пользовательскому ресурсу на системном уровне и с его помощью создает нового пользователя, которого можно в любой момент использовать для доступа в сеть и к ее ресурсам.

Смягчить угрозу сниффинга пакетов можно с помощью следующих средств.

1. Аутентификация. Сильные средства аутентификации являются первым способом защиты от сниффинга пакетов. Под «сильным» понимается такой метод аутентификации, который трудно обойти. Примером такой аутентификации являются однократные пароли(OTP – One-Time Passwords).

OTP – это технология двухфакторной аутентификации.

Типичным примером двухфакторной аутентификации является работа обычного банкомата, который опознает клиента, во-первых, по пластиковой карточке и, во-вторых, по вводимому

ПИН-коду. Для аутентификации в системе OTP также требуется ПИН-код и личная карточка. Снифферы, перехватывающие другую информацию (например, сообщения электронной почты), не теряют своей эффективности.

2. Коммутируемая инфраструктура. Еще одним способом борьбы со сниффингом пакетов в сетевой среде является создание коммутируемой инфраструктуры. Если, к примеру, во всей организации используется коммутируемый Ethernet, хакеры могут получить доступ только к трафику, поступающему на тот порт, к которому они подключены. Коммутируемая инфраструктура не ликвидирует угрозу сниффинга, но заметно снижает ее остроту.

3. Антиснифферы. Третий способ борьбы со сниффингом заключается в установке аппаратных или программных средств, распознающих снифферы, работающие в вашей сети. Эти средства не могут полностью ликвидировать угрозу, но, как и многие другие средства сетевой безопасности, включаются в общую систему защиты. Так называемые «антиснифферы» измеряют время реагирования хостов и определяют, не приходится ли хостам обрабатывать «лишний» трафик. Подобного рода средства не могут полностью ликвидировать угрозу сниффинга, но крайне необходимы при построении комплексной системы защиты. Одно из таких средств, поставляемых компанией LOpht Heavy Industries, называется AntiSniff.

3. Криптография. Самый эффективный способ борьбы со сниффингом пакетов не предотвращает перехват и не распознает работу снифферов, но делает эту работу бесполезной. Если канал связи является криптографически

защищенным, это значит, что хакер перехватывает не сообщение, а зашифрованный текст.

Например, криптография Cisco на сетевом уровне базируется на протоколе IPSec. IPSec представляет собой стандартный метод защищенной связи между устройствами с помощью протокола IP.

К прочим криптографическим протоколам сетевого управления относятсяпротоколыSSH (Secure Shell) иSSL (Secure Socket Layer).

IP-спуфинг – это вид атаки, при которой хакер, находящийся внутри организации или за ее пределами, выдает себя за санкционированного пользователя.

Это можно сделать двумя способами. Во-первых, хакер может воспользоваться IP-адресом, находящимся в пределах диапазона санкционированных IP-адресов, или авторизованным внешним адресом, которому разрешается доступ к определенным сетевым ресурсам. Атаки IP-спуфинга часто являются отправной точкой для прочих атак. Классический пример – атака DoS, которая начинается с чужого адреса, скрывающего истинную личность хакера.

Если же хакеру удается поменять таблицы маршрутизации и направить трафик наложный IP-адрес, хакер получит все пакеты и сможет отвечать на них так, будто он является санкционированным пользователем.

Полностью устранить угрозу спуфинга практически невозможно, но ее можно ослабить с помощью следующих мер.

1. Контроль доступа. Самый простой способ предотвращения IP-спуфинга состоит в правильной настройке управления доступом.

Чтобы снизить эффективность IP-спуфинга, необходимо настроить контроль доступа на отсечение любого трафика, поступающего из внешней сети с исходным адресом, который должен располагаться внутри защищаемой сети. Заметим, что это помогает бороться с IP- спуфингом, когда санкционированными являются только внутренние адреса. Если санкционированными являются и некоторые адреса внешней сети, данный метод становится неэффективным.

2. Фильтрация RFC 2827. Можно пресечь попытки спуфинга чужих сетей пользователями некоторой сети. Для этого необходимо отбраковывать любой исходящий трафик, исходный адрес которого не является одним из IP-адресов данной организации. В результате отбраковывается весь трафик, который не имеет исходного адреса, ожидаемого на определенном интерфейсе.

3. Криптография. Наиболее эффективный метод борьбы с IP-спуфингом тот же, что и в случае со сниффингом пакетов: необходимо сделать атаку абсолютно неэффективной. IP-спуфинг может функционировать только при условии, что аутентификация происходит на базе IP-адресов.

Отказ в обслуживании(Denial of Service, DoS). Атака DoS делает сеть недоступной для обычного использования за счет превышения допустимых пределов функционирования сети, операционной системы или приложения.

Атаки DoS, без всякого сомнения, являются наиболее известной формой хакерских атак и одной из самых молодых технологий. Против атак такого типа труднее всего создать стопроцентную защиту. Атаки DoS считаются тривиальными, а от хакера для своей организации они требуют минимум знаний и умений: все необходимое программное обеспечение вместе с описаниями самой технологии совершенно свободно доступно в Интернете.

Именно простота реализации и огромный причиняемый вред привлекают к DoS пристальное внимание администраторов, отвечающих за сетевую безопасность.

О DoS-атаках широко заговорили после того, как в декабре 1999 года при помощи этой технологии были успешно атакованы web-узлы таких известных корпораций, как Amazon, Yahoo, CNN, eBay и E-Trade.

В случае использования некоторых серверных приложений (таких, например, как Web-сервер или FTP-сервер) атаки DoS могут заключаться в том, чтобы занять все соединения, доступные для этих приложений, и держать их в занятом состоянии, не допуская обслуживания обычных пользователей. В ходе атак DoS могут использоваться обычные Интернет-протоколы, такие как TCP и ICMP (Internet Control Message Protocol).Большинство атак DoS опирается не на программные ошибки или бреши в системе безопасности, а на общие слабости системной архитектуры. Некоторые атаки сводят к нулю производительность сети, переполняя ее нежелательными и ненужными пакетами или сообщая ложную информацию о текущем состоянии сетевых ресурсов. Этот тип атак трудно предотвратить, так как для этого требуется координация действий с провайдером. Когда атака этого типа проводится одновременно через множество устройств, речь идет о распределенной атаке DDoS (Distributed DoS).

Наиболее известными разновидностями атак DoS являются: TCP SYN Flood, Ping of Death, Tribe Flood Network (TFN) и Tribe Flood Network 2000 (TFN2K), Trinco, Stacheldracht, Trinity, Smurf, ICMP flood, UDP flood, TCP flood.

Рассмотрим некоторые из них более подробно.

Smurf – ping-запросы ICMP (Internet Control Message Protocol) по адресу направленной широковещательной рассылки. Используемый в пакетах этого запроса фальшивый адрес источника в результате оказывается мишенью атаки. Системы, получившие направленный широковещательный ping-запрос, отвечают на него и «затапливают» сеть, в которой находится сервер-мишень.

ICMP flood – атака, аналогичная Smurf, только без усиления, создаваемого запросами по направленному широковещательному адресу.

UDP flood – отправка на адрес системы-мишени множества пакетов UDP, что приводит к «связыванию» сетевых ресурсов.

TCP flood – отправка на адрес системы-мишени множества TCP-пакетов, что также приводит к «связыванию» сетевых ресурсов.

TCP SYN flood – при проведении такого рода атаки выдается большое количество запросов на инициализацию TCP-соединений с узлом-мишенью,

которому, в результате, приходится расходовать все свои ресурсы на то, чтобы отслеживать эти частично открытые соединения.

Угроза атак типа DoS может снижаться тремя способами.

1. Функции антиспуфинга. Правильная конфигурация функций антиспуфинга на маршрутизаторах и межсетевых экранах помогает снизить риск DoS атак. Эти функции, как минимум, должны включать фильтрацию RFC 2827.

2. Функции антиDoS. Правильная конфигурация функций антиDoS на маршрутизаторах и межсетевых экранах может ограничить эффективность атак. Эти функции часто ограничивают число полуоткрытых каналов в любой момент времени.

3. Ограничение объема трафика (traffic rate limiting). Организация может попросить провайдера ограничить объем трафика.

Этот тип фильтрации позволяет ограничить объем некритического трафика, проходящего по сети. Обычным примером является ограничение объемов трафика ICMP, который используется только для диагностических целей. Атаки DDoS часто используют ICMP.

Парольные атаки – попытка подбора пароля легального пользователя для входа в сеть.

Хакеры могут проводить парольные атаки с помощью целого ряда методов, таких как простой перебор (brute force attack), троянский конь, IP-спуфинг и сниффинг пакетов. Хотя логин и пароль часто можно получить при помощи IP-спуфинга и сниффинга пакетов, хакеры часто пытаются подобрать пароль и логин, используя для этого многочисленные попытки доступа. Такой

подход носит название простого перебора. Часто для такой атаки используется специальная программа, которая пытается получить доступ к ресурсу общего пользования (например к серверу).

Еще одна проблема возникает в случаях, когда пользователи применяют один и тот же пароль для доступа ко многим системам: корпоративной, персональной и системам Интернет.

Парольных атак можно избежать, если не пользоваться паролями в текстовой форме. Одноразовые пароли и/или криптографическая аутентификация могут практически свести на нет угрозу таких атак. К сожалению, не все приложения, хосты и устройства поддерживают указанные выше методы аутентификации.

С точки зрения администратора, существует несколько методов использовании средства LophtCrack, которое часто применяют хакеры для подбора паролей в среде Windows NT. Это средство быстро показывает, легко ли подобрать пароль, выбранный пользователем.

Атаки типа Man-in-the-Middle – непосредственный доступ к пакетам, передаваемым по сети.

Такой доступ ко всем пакетам, передаваемым от провайдера в любую другую сеть, может, к примеру, получить сотрудник этого провайдера. Для атак

этого типа часто используются снифферы пакетов, транспортные протоколы и протоколы маршрутизации.

Атаки проводятся с целью кражи информации, перехвата текущей сессии и получения доступа к частным сетевым ресурсам, для анализа трафика и получения информации о сети и ее пользователях, для проведения атак типа DoS, искажения передаваемых данных и ввода несанкционированной информации в сетевые сессии.

Эффективно бороться с атаками типа Man-in-the-Middle можно только с помощью криптографии.

Атаки на уровне приложений могут проводиться несколькими способами. Самый распространенный из них состоит в использовании хорошо известных слабостей серверного программного обеспечения (sendmail, HTTP, FTP). Используя эти слабости, хакеры могут получить доступ к компьютеру от имени пользователя,

работающего с приложением (обычно это бывает не простой пользователь, а привилегированный администратор с правами системного доступа). Сведения об атаках на уровне приложений широко публикуются, чтобы дать возможность администраторам исправить проблему с помощью коррекционных модулей (патчей).

К сожалению, многие хакеры также имеют доступ к этим сведениям, что позволяет им учиться.

Главная проблема с атаками на уровне приложений состоит в том, что хакеры часто пользуются портами, которым разрешен проход через межсетевой экран. К примеру, хакер, эксплуатирующий известную «слабость» Web-сервера, часто использует в ходе атаки TCP порт 80. Поскольку Web-сервер предоставляет пользователям Web-страницы, межсетевой экран должен предоставлять доступ к этому порту. С точки зрения межсетевого экрана, атака рассматривается как стандартный трафик для порта 80.

Полностью исключить атаки на уровне приложений невозможно. Хакеры постоянно открывают и публикуют в Интернете все новые уязвимые места прикладных программ. Самое главное здесь – хорошее системное администрирование. Вот некоторые меры, которые можно предпринять, чтобы снизить уязвимость для атак этого типа:

− чтение лог-файлов операционных систем и сетевых логфайлов и/или их анализ с помощью специальных аналитических приложений;

− подписка на услуги по рассылке данных о слабых местах прикладных программ;

− использование последних версий операционных систем и приложений и самых последних коррекционных модулей (патчей);

− кроме системного администрирования необходимо использование систем распознавания атак (IDS); существуют две взаимно дополняющие друг друга технологии IDS: первая – сетевая система IDS (NIDS), которая отслеживает все пакеты, проходящие через определенный домен; когда система NIDS видит пакет или серию пакетов, совпадающих с сигнатурой известной

или вероятной атаки, она генерирует сигнал тревоги и/или прекращает сессию; вторая – хост-система IDS (HIDS), защищающая хост с помощью программных агентов; эта система борется только с атаками против одного хоста;

− в своей работе системы IDS пользуются сигнатурами атак, которые представляют собой профили конкретных атак или типов атак. Сигнатуры определяют условия, при которых трафик считается хакерским.

Сетевая разведка – сбор информации о сети с помощью общедоступных данных и приложений.

При подготовке атаки против какой-либо сети хакер, как правило, пытается получить о ней как можно больше информации.

Сетевая разведка проводится в форме запросов DNS, эхотестирования (ping sweep) и сканирования портов. Запросы DNS помогают понять, кто владеет тем или иным доменом и какие адреса этому домену присвоены. Эхо-тестирование _____ адресов, раскрытых с помощью DNS, позволяет увидеть, какие хосты реально работают в данной среде.

Полностью избавиться от сетевой разведки невозможно.

Системы IDS на уровне сети и хостов обычно хорошо справляются с задачей уведомления администратора о ведущейся сетевой разведке, что позволяет лучше подготовиться к предстоящей атаке и оповестить провайдера, в сети которого установлена система, проявляющая чрезмерное любопытство.

Злоупотребление доверием – злонамеренное использование отношений доверия, существующих в сети.

Классическим примером такого злоупотребления является ситуация в периферийной части корпоративной сети. В этом сегменте часто располагаются серверы DNS, SMTP и HTTP. Поскольку все они принадлежат к одному и тому же сегменту, взлом одного из них приводит к взлому и всех остальных, так как эти серверы доверяют другим системам своей сети. Другим примером является система, установленная с внешней стороны межсетевого экрана, имеющая отношения доверия с системой, установленной с его внутренней стороны. В случае взлома внешней системы, хакер может использовать отношения доверия для проникновения в систему, защищенную межсетевым экраном.

Риск злоупотребления доверием можно снизить за счет более жесткого контроля уровней доверия в пределах своей сети.

Системы, расположенные с внешней стороны межсетевого экрана, никогда не должны пользоваться абсолютным доверием со стороны защищенных экраном систем. Отношения доверия должны ограничиваться определенными протоколами и, по возможности, аутентифицироваться не только по IP-адресам, но и по другим параметрам.

Переадресация портов представляет собой разновидность злоупотребления доверием, когда взломанный хост используется для передачи через межсетевой экран трафика, который в противном случае был бы обязательно отбракован.

Представим себе межсетевой экран с тремя интерфейсами, к каждому из которых подключен определенный хост. Внешний

хост может подключаться к хосту общего доступа (DMZ), но не к хосту, установленному с внутренней стороны межсетевого экрана. Хост общего доступа может подключаться и к внутреннему, и к внешнему хосту. Если хакер захватит хост общего доступа, он сможет установить на нем программное средство, перенаправляющее трафик с внешнего хоста прямо на внутренний хост. Примером приложения, которое может предоставить такой доступ, является netcat.

Основным способом борьбы с переадресацией портов является использование надежных моделей доверия.

Несанкционированный доступ не может считаться отдельным типом атаки. Большинство сетевых атак проводится ради получения несанкционированного доступа. Источник таких атак может находиться как внутри сети, так и снаружи.

Способы борьбы с несанкционированным доступом достаточно просты. Главным здесь является сокращение или полная ликвидация возможностей хакера получения доступа к системе с помощью несанкционированного протокола. В качестве примера можно рассмотреть недопущение хакерского доступа к порту telnet на сервере, который предоставляет Web-услуги внешним пользователям.

Рабочие станции конечных пользователей очень уязвимы для вирусов (компьютерных) и «троянских коней».

Вирусами называются вредоносные программы, которые внедряются в другие программы для выполнения определенной

нежелательной функции на рабочей станции конечного пользователя, способные к самомодификации (мутации).

В качестве примера можно привести вирус, который прописывается в файле command.com (главном интерпретаторе систем Windows) и стирает другие файлы, а также заражает все другие найденные им версии command.com.

Троянский конь – это не программная вставка, а настоящая программа, которая выглядит как полезное приложение, а на деле причиняет вред.

Примером типичного троянского коня является программа, которая выглядит, как простая игра для рабочей станции пользователя. Однако пока пользователь играет в игру, программа отправляет свою копию по электронной почте каждому абоненту, занесенному в адресную книгу этого пользователя. Все абоненты получают по почте игру, вызывая ее дальнейшее распространение.

Борьба с вирусами и троянскими конями ведется с помощью эффективного антивирусного программного обеспечения, работающего на пользовательском уровне и, возможно, на уровне сети.

По мере появления новых вирусов и «троянских коней» организация должна устанавливать новые версии антивирусных средств и приложений.

Почтовая бомбардировка, или **бомбардировка электронной почтой** (mailbombing, мэйлбомбинг) – один из самых старых и примитивных видов интернет-атак. Суть мэйлбомбинга заключается в засорении почтового ящика

«мусорной» корреспонденцией или даже выведении из строя почтового сервера интернетпровайдера.

Для этого применяются специальные программы – **мэйлбомберы**. Они попросту засыпают указанный в качестве мишени почтовый ящик огромным количеством писем, указывая при этом фальшивые данные отправителя – вплоть до IP-адреса. Все, что нужно агрессору, использующему такую программу, указать e-mail объекта атаки, число сообщений, написать текст письма, указать фальшивые данные отправителя, если программа этого не делает сама, и нажать кнопку Отправить.

Принципы криптографической защиты информации

Ранее было отмечено, что последний и практически непреодолимый «рубеж» защиты от несанкционированного доступа в компьютерных системах и в том числе в компьютерных сетях образует шифрование.

Шифрование данных представляет собой разновидность программных средств защиты информации и имеет особое значение на практике как единственная надежная защита информации, передаваемой по протяженным последовательным линиям, от утечки.

Понятие «шифрование» часто употребляется в связи с понятием «криптография».

Криптография изучает методы преобразования информации, обеспечивающие ее конфиденциальность и аутентичность.

Аутентичность информации состоит в подлинности авторства и целостности.

В проблематике современной криптографии можно выделить следующие три типа основных задач:

− обеспечение конфиденциальности;

− создание условий для анонимности (неотслеживаемости);

− обеспечение аутентификации информации и источника сообщения.

Первый тип задач относится к защите информации от несанкционированного доступа по секретному ключу. Доступ к информации (информационным ресурсам) имеют только обладатели ключа. Второй и третий типы задач обязаны своей постановкой массовому применению электронных способов обработки и передачи информации (банковская сфера, электронная коммерция, каналы межличностной коммуникации и др.).

Криптографическое преобразование состоит из двух этапов:

прямого и обратного. Прямое преобразование называют зашифрованием (в соответствии со стандартом ISO 7492-2, шифрованием, encrypt), обратное – расшифрованием (дешифрованием, decrypt).

С точки зрения криптографии **шифр**, или **криптографическая система** – совокупность обратимых преобразований множества открытых данных на множество зашифрованных данных, задаваемых ключом и алгоритмом криптографического преобразования.

Следует различать понятия ключ и пароль.

Пароль является секретной последовательностью букв алфавита, однако используется не для шифрования (как ключ), а для аутентификации субъектов.

В симметричных криптосистемах для зашифрования и для расшифрования используется один и тот же ключ.

В ассиметричных криптосистемах используются два ключа – открытый (публичный) и закрытый (секретный, тайный), которые математически связаны друг с другом.

Электронной цифровой подписью называется присоединяемое к тексту его криптографическое преобразование, которое позволяет при получении текста другим пользователем проверить авторство и целостность сообщения.

Криптостойкостью называется характеристика шифра, определяющая его стойкость к расшифрованию без знания ключа, т. е. к криптоатаке.

Удачную криптоатаку называют **взломом.**

Эффективность использования пароля для защиты информации

Наиболее часто применяемыми методами идентификации и аутентификации пользователей являются методы, основанные на использовании паролей.

Пароль представляет собой некоторую последовательность символов, сохраняемую в секрете и предъявляемую при обращении к компьютерной системе.

Для ввода пароля, как правило, используется штатная клавиатура компьютерных систем (КС); при этом в процессе ввода пароль не должен отображаться на экране монитора, а чтобы пользователь мог ориентироваться в количестве введенных символов, на экран выдаются специальные символы.

При составлении и хранении пароля пользователи должны придерживаться следующих рекомендаций:

− пароль должен запоминаться субъектом доступа;

− запись пароля (на бумажном или электронном носителе) значительно повышает вероятность его компрометации (нарушения конфиденциальности);

− легко запоминаемый пароль должен быть в то же время сложным для отгадывания; не рекомендуется использовать для этой цели имена, фамилии, даты рождения и т. п.;

− желательным является наличие в пароле парадоксального сочетания букв, слов и т. п., полученного, например, путем набора русских букв пароля на латинском регистре.

Вероятность подбора пароля уменьшается также при увеличении его длины и времени задержки между разрешенными попытками повторного ввода неправильно введенного пароля. Ожидаемое время раскрытия пароля TP можно вычислить по следующей приближенной формуле:

$$TP=2AS \cdot t,$$

где A – число символов в алфавите, из которых составляется пароль (например, 26 символов латинского алфавита);

S – длина пароля; t = E / R – время, необходимое на попытку введения пароля;

E – число символов в сообщении, передаваемом в систему при попытке получить к ней доступ (включая пароль и служебные символы);

R – скорость передачи символов пароля (симв./мин).

В приведенной формуле считается, что злоумышленник имеет возможность непрерывно осуществлять подбор пароля. Например, еслиA= 26, t = 2 с и S = 6 символов, то ожидаемое время раскрытия TPпароля приблизительно равно одному году. Если в данном примере после каждой неудачной попытки ввода пароля предусмотреть временную задержку в 10 с, то ожидаемое время раскрытия пароля увеличится в 5 раз.

Следует также отметить, что на безопасное время раскрытия пароля оказывает существенное влияние длина пароля S (в степенной зависимости). Так, если для трехсимвольного пароля, выбранного из 26-символьного алфавита, время TPсоставит 3 мес., то для 4-символьного – 65 лет.

Выбор необходимой длины пароля S можно производить исходя из заданной вероятности Ртого, что данный пароль может быть раскрыт посторонним лицом за время M. Если необходимо построить систему, где незаконный пользователь имел бы вероятность отгадывания правильного пароля не большую, чем заданная вероятность P, то следует выбрать такое значение S, которое удовлетворяло бы формуле Андерсена:

$$AS \geq (4{,}32 \cdot 104 \cdot R \cdot M) / (E \cdot P), \quad (11.14)$$

где M – период времени, в течение которого предпринимаются попытки раскрытия пароля (в месяцах при ежедневном 24-часовом тестировании).

Пример 8. Допустим, требуется, используя стандартны его отгадывания не превысила 0,001 после трехмесячного систематического тестирования. Если за одну попытку доступа посылается 20 символов (E = 20), а скорость их передачи R = 600 симв./мин, то по формуле Андерсена получаем:

$(4{,}32 \cdot 104 \cdot R \cdot M) / (E \cdot P) = (4{,}32 \cdot 104 \cdot 3 \cdot 103 \cdot 600) / 20 = 3{,}888 \cdot 109.$

ДляS = 6: $26S = 3{,}089 \cdot 108$, т. е. $< 3{,}888 \cdot 109$.

ДляS = 7: $26S = 8{,}03 \cdot 109$, т. е. $> 3{,}888 \cdot 109$.

Таким образом, при данных обстоятельствах следует выбрать длину пароля S = 7.

При существенном увеличении длины пароля он может быть разбит на две части: запоминаемую пользователем и вводимую вручную, а также размещенную в зашифрованном виде на специальном носителе (например, дискете, магнитной карте и т. д.) и считываемую специальным устройством.

Повышение стойкости системы защиты на этапе аутентификации можно достигнуть и увеличением числа символов алфавита, используемого при вводе пароля. Для этого при наборе символов пароля можно использовать несколько

регистров клавиатуры, соответствующих, например, строчным и прописным латинским символам, а также строчным и прописным символам кириллицы.

На степень информационной безопасности при использовании простого парольного метода проверки подлинности пользователей большое влияние оказывают ограничения на минимальное и максимальное время действия каждого пароля. Чем чаще меняется пароль, тем более высокий уровень безопасности обеспечивается.

Администратор службы безопасности должен постоянно контролировать своевременность смены паролей пользователей.

Таким образом, для повышения надежности аутентификации пользователей следует, по возможности, использовать нетривиальные (уникальные) пароли и, кроме того, обеспечивать более частую их смену.

С этой точки зрения являются достаточно эффективными методы, основанные на использовании динамически изменяющихся паролей. При смене пароля осуществляется его функциональное преобразование, зависящее от динамически изменяющихся параметров, например, суточного времени в часах, номера дня недели, месячной даты и т. д. Такая смена пароля производится либо периодически (ежедневно, каждые три дня или каждую неделю), либо

при очередном обращении пользователя.

6. IDSsnort. Средства анализа информационной безопасности компьютерных сетей

Обнаружения факта проведения и причин возникновения сетевой атаки подручными средствами. Общие сведения об IDS snort. Синтаксис и механизм работы правил snort. Дополнительные инструменты, облегчающие эффективность snort.

Значение IDS для решения задач поиска злоумышленников в собственной ЛВС.

Рассмотрение успешной сетевой атаки ЛВС через сеть Интернет на примере конкретной локальной сети, не оборудованной IDS. Выявление факта проведения атаки, а также причин ее успешного проведения при помощи стандартных средств. Меры по предупреждению подобных атак как с помощью IDS, так и ее отсутствии. Преимущества IDS перед другими средствами защиты ЛВС (с точки зрения обнаружения и предотвращения рассмотренной разновидности сетевых атак)

Режимы работы snort. Конфигурирование системы регистрации событий румента snort. Предпроцессоры snort: назначение, конфигурация, использование. Краткий обзор и классификация типов и правил инструмента snort. Синтаксические правила Snort. Примеры написания и работы на конкретных примерах.

Основные цели сетевых злоумышленников, использующиеся для незаконного доступа к информации, хранящейся на сторонних ПК, локальной сети предприятия. Процесс обнаружения подобных злоумышленников и сбор доказательств их незаконной деятельности с помощью инструментов Etheral и

Tcpdump. Перехват почтовых сообщений злоумышленника при помощи инструмента Dsniff.Инструмент IE history