

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

С.М. Платунова

Построение корпоративной сети
с применением коммутационного
оборудования и настройкой безопасности

Учебное пособие



Санкт-Петербург

2012

Платунова С.М. Построение корпоративной сети с применением коммутационного оборудования и настройкой безопасности. Учебное пособие по дисциплине «Корпоративные сети». – СПб: НИУ ИТМО, 2012. – 85 с.

В учебном пособии описаны общие принципы построения вычислительных сетей, стандарты построения сетей, протоколы стека TCP/IP, протоколы маршрутизации, особенности маршрутизации сетей класса SONET, технология VPN, пример настройки оборудования фирмы ZyXEL серии ZyWALL USG.

Пособие адресовано специалистам с высшим и средним профессиональным образованием, имеющим опыт работы в области IT технологий, обучающихся по направлению 230100 Информатика и вычислительная техника.

Рекомендовано к печати Ученым советом факультета Академии ЛИМТУ, протокол №6 от 23.12.2011



В 2009 году Университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа его развития на 2009–2018 годы. В 2011 году Университет получил наименование «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

© Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, 2012

© С.М. Платунова, 2012

Содержание

Введение	4
Тема 1. Модель OSI	5
Сетезависимые и сетенезависимые уровни	11
Тема 2. Стек протоколов TCP/IP	12
Тема 3. Адресация в IPv4-сетях	16
Тема 4. Протокол межсетевого взаимодействия IP	25
Тема 5. Протокол доставки пользовательских дейтаграмм UDP	35
Тема 6. Протокол надежной доставки сообщений TCP	37
Тема 7. Протоколы обмена маршрутной информацией стека TCP/IP	44
Тема 8. Маршрутизация небольших сетей	55
Конструкция маршрутизатора	56
Брандмауэр (firewall)	57
Тема 9. Протокол NAT	61
Перенаправление портов (Port mapping)	62
DMZ-зона	64
Методы аутентификации	64
DHCP-сервер	65
Виртуальные сети VPN	65
PPTP	66
L2TP	66
IPsec	66
Режимы функционирования VPN	66
Тема 10. Технология VPN	67
Основы туннелирования	68
Тема 11. Настройка интерфейса шлюза безопасности серии ZyWALL USG	69
Тема 12. Настройка балансировки внешних интерфейсов в аппаратном шлюзе серии ZyWALL USG	76
Тема 13. BWM (Bandwidth Management) в аппаратных шлюзах безопасности серии ZyWALL USG	78
Вопросы для самоконтроля	81
Литература	82

Введение

Организация взаимодействия между устройствами в сети является сложной задачей. Для решения сложных задач используется универсальный прием — декомпозиция, или разбиение одной сложной задачи на несколько более простых задач-модулей [Олифер Олифер Н.А., 2010]. Процедура декомпозиции включает в себя определение функций каждого модуля, решающего отдельную задачу, и интерфейсов между ними. В результате достигается логическое упрощение задачи, и появляется возможность модификации отдельных модулей без изменения остальной части системы.

При декомпозиции часто используется многоуровневый подход. Все множество модулей разбивают на уровни. Уровни образуют иерархию, то есть имеются выше- и нижележащие уровни. Множество модулей, составляющих каждый уровень, сформировано таким образом, что для выполнения своих задач они обращаются с запросами только к модулям непосредственно примыкающего нижележащего уровня. Результаты работы всех модулей, принадлежащих некоторому уровню, могут быть переданы только модулям соседнего вышележащего уровня.

Иерархическая декомпозиция задачи предполагает определение функции каждого уровня и интерфейсов между уровнями.

Интерфейс определяет набор функций, которые нижележащий уровень предоставляет вышележащему уровню. В результате иерархической декомпозиции достигается относительная независимость уровней и возможность их замены.

Средства сетевого взаимодействия тоже могут быть представлены в виде иерархически организованного множества модулей.

Модули, реализующие протоколы соседних уровней и находящиеся в одном узле, также взаимодействуют друг с другом в соответствии с определенными правилами и с помощью стандартизованных форматов сообщений. Эти правила называются интерфейсом.

Интерфейс определяет набор служб, предоставляемый данным уровнем соседнему уровню. Протокол и интерфейс выражают одно и то же понятие, но традиционно в сетях за ними закрепили разные области действия: протоколы определяют правила взаимодействия модулей одного уровня в разных узлах, а интерфейсы определяют правила взаимодействия модулей соседних уровней в одном узле.

Средства каждого уровня должны отрабатывать, во-первых, свой собственный протокол, а во-вторых, интерфейсы с соседними уровнями.

Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется *стеком* коммуникационных протоколов.

Протоколы реализуются компьютерами и другими сетевыми устройствами — концентраторами, мостами, коммутаторами,

маршрутизаторами и т. д. В зависимости от типа устройства в нем должны быть встроенные средства, реализующие некоторый набор протоколов.

Тема 1. Модель OSI

При реализации сетей стремятся использовать стандартные протоколы: фирменные, национальные или международные стандарты.

В начале восьмидесятых годов ряд международных организаций по стандартизации, в частности International Organization for Standardization (ISO), часто называемая также International Standards Organization, а также International Telecommunications Union (ITU) и некоторые другие, – разработали стандартную модель взаимодействия открытых систем (Open System Interconnection, OSI).

Модель OSI определяет различные уровни взаимодействия систем, дает им стандартные имена и указывает, какие функции должен выполнять каждый уровень. Модель OSI не содержит описаний реализаций конкретного набора протоколов.

Модель OSI описывает только системные средства взаимодействия, реализуемые операционной системой, системными утилитами, системными аппаратными средствами.

В модели OSI средства взаимодействия делятся на семь уровней: прикладной, представления, сеансовый, транспортный, сетевой, канальный и физический. Каждый уровень имеет дело с определенным аспектом взаимодействия сетевых устройств.

Физический уровень (physical layer) имеет дело с передачей потока битов по физическим каналам связи, таким как коаксиальный кабель, витая пара, оптоволоконный кабель, или цифровой территориальный канал. Функции физического уровня реализуются на всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом. Физический уровень не вникает в смысл информации, которую он передает. Для него эта информация представляет однородный поток битов, которые нужно доставить без искажений и в соответствии с заданной тактовой частотой.

Канальный уровень (data link layer) является первым уровнем, который работает в режиме коммутации пакетов. На этом уровне блок данных (PDU) носит название кадр (frame). Функции средств канального уровня определяются по-разному для локальных и глобальных сетей.

В локальных сетях канальный уровень должен обеспечивать доставку кадра между любыми узлами сети. При этом предполагается, что сеть имеет типовую топологию, например общую шину, кольцо, звезду или дерево.

В глобальных сетях канальный уровень должен обеспечивать доставку кадра только между двумя соседними узлами, соединенными

индивидуальной линией связи. Примерами двухточечных протоколов могут служить широко распространенные протоколы PPP и HDLC. На основе двухточечных связей могут быть построены сети произвольной топологии.

Одной из функций канального уровня является поддержание интерфейсов с нижележащим физическим уровнем и вышележащим сетевым уровнем. Сетевой уровень направляет канальному уровню пакет для передачи в сеть или принимает от него пакет, полученный из сети. Физический уровень используется канальным как инструмент, который принимает и передает в сеть последовательности битов.

Сетевой уровень отправителя передает канальному уровню пакет, а также указание, какому узлу его передать. Для решения этой задачи канальный уровень создает кадр, который имеет поле данных и заголовок. Канальный уровень помещает (инкапсулирует) пакет в поле данных кадра и заполняет соответствующей служебной информацией заголовок кадра. Важнейшей информацией заголовка кадра является адрес назначения, на основании которого коммутаторы сети будут продвигать пакет.

Одной из задач канального уровня является обнаружение и коррекция ошибок. Для этого канальный уровень фиксирует границы кадра, помещая специальную последовательность битов в его начало и конец, а затем добавляет к кадру контрольную сумму, которая называется также контрольной последовательностью кадра (Frame Check Sequence, FCS). Контрольная сумма вычисляется по некоторому алгоритму как функция от всех байтов кадра. По значению FCS узел назначения сможет определить, были или не были искажены данные кадра в процессе передачи по сети.

Однако прежде, чем переправить кадр физическому уровню для непосредственной передачи данных в сеть, канальному уровню может потребоваться решить еще одну важную задачу. Если в сети используется разделяемая среда, то прежде чем физический уровень начнет передавать данные, канальный уровень должен проверить доступность среды. Функции проверки доступности разделяемой среды иногда выделяют в отдельный подуровень управления доступом к среде (Media Access Control, MAC).

Если разделяемая среда освободилась (когда она не используется, то такая проверка, конечно, пропускается), кадр передается средствами физического уровня в сеть, проходит по каналу связи и поступает в виде последовательности битов в распоряжение физического уровня узла назначения. Этот уровень в свою очередь передает полученные биты «наверх» канальному уровню своего узла. Последний группирует биты в кадры, снова вычисляет контрольную сумму полученных данных и сравнивает результат с контрольной суммой, переданной в кадре. Если они совпадают, кадр считается правильным. Если же контрольные суммы не

совпадают, фиксируется ошибка. В функции канального уровня входит не только обнаружение ошибок, но и исправление их за счет повторной передачи поврежденных кадров. Однако эта функция не является обязательной и в некоторых реализациях канального уровня оно отсутствует, например в Ethernet, Token Ring, FDDI и Frame Relay.

Протокол канального уровня реализуется компьютерами, мостами, коммутаторами и маршрутизаторами. В компьютере функции канального уровня реализуются совместными усилиями сетевых адаптеров и их драйверов.

Протокол канального уровня обычно работает в пределах сети, являющейся одной из составляющих более крупной составной сети, объединенной протоколами сетевого уровня. Адреса, с которыми работает протокол канального уровня, используются для доставки кадров только в пределах этой сети, а для перемещения пакетов между сетями применяются уже адреса следующего, сетевого, уровня.

В локальных сетях канальный уровень поддерживает весьма мощный и законченный набор функций по пересылке сообщений между узлами сети. В некоторых случаях протоколы канального уровня локальных сетей оказываются самодостаточными транспортными средствами и могут допускать работу непосредственно поверх себя протоколов прикладного уровня или приложений без привлечения средств сетевого и транспортного уровней. Тем не менее, для качественной передачи сообщений в сетях с произвольной топологией функций канального уровня оказывается недостаточно.

Это утверждение в еще большей степени справедливо для глобальных сетей, в которых протокол канального уровня реализует достаточно простую функцию передачи данных между соседними узлами.

Сетевой уровень (network layer) служит для образования единой транспортной системы, объединяющей несколько сетей, называемой составной сетью, или интернетом.

Технология, позволяющая соединять в единую сеть множество сетей, в общем случае построенных на основе разных технологий, называется технологией множества взаимодействия (internetworking).

Функции сетевого уровня реализуются:

- группой протоколов;
- специальными устройствами – маршрутизаторами.

Одной из функций маршрутизатора является физическое соединение сетей. Маршрутизатор имеет несколько сетевых интерфейсов, подобных интерфейсам компьютера, к каждому из которых может быть подключена одна сеть. Таким образом, все интерфейсы маршрутизатора можно считать узлами разных сетей. Маршрутизатор может быть реализован программно, на базе универсального компьютера (например, типовая конфигурация Unix или Windows включает программный модуль маршрутизатора).

Однако чаще маршрутизаторы реализуются на базе специализированных аппаратных платформ. В состав программного обеспечения маршрутизатора входят протокольные модули сетевого уровня.

Данные, которые необходимо передать через составную сеть, поступают на сетевой уровень от вышележащего транспортного уровня. Эти данные снабжаются заголовком сетевого уровня. Данные вместе с заголовком образуют пакет – так называется PDU сетевого уровня. Заголовок пакета сетевого уровня имеет унифицированный формат, не зависящий от форматов кадров канального уровня тех сетей, которые могут входить в составную сеть, и несет наряду с другой служебной информацией данные об адресе назначения этого пакета.

Для того чтобы протоколы сетевого уровня могли доставлять пакеты любому узлу составной сети, эти узлы должны иметь адреса, уникальные в пределах данной составной сети. Такие адреса называются сетевыми, или глобальными. Каждый узел составной сети, который намерен обмениваться данными с другими узлами составной сети, должен иметь сетевой адрес наряду с адресом, назначенным ему на канальном уровне. Например, компьютер в сети Ethernet, входящей в составную сеть, имеет адрес канального уровня MAC1 и адрес сетевого уровня NET-A1; аналогично в сети АТМ узел, адресуемый идентификаторами виртуальных каналов ID1 и ID2, имеет сетевой адрес NET-A2. В пакете в качестве адреса назначения должен быть указан адрес сетевого уровня, на основании которого определяется маршрут пакета. Определение маршрута является важной задачей сетевого уровня. Маршрут описывается последовательностью сетей (или маршрутизаторов), через которые должен пройти пакет, чтобы попасть к адресату. Маршрутизатор собирает информацию о топологии связей между сетями и на ее основании строит таблицы коммутации, которые в данном случае носят специальное название таблиц маршрутизации.

В соответствии с многоуровневым подходом сетевой уровень для решения своей задачи обращается к нижележащему канальному уровню. Весь путь через составную сеть разбивается на участки от одного маршрутизатора до другого, причем каждый участок соответствует пути через отдельную сеть.

Для того чтобы передать пакет через очередную сеть, сетевой уровень помещает его в поле данных кадра соответствующей канальной технологии, указывая в заголовке кадра канальный адрес интерфейса следующего маршрутизатора. Сеть, используя свою канальную технологию, доставляет кадр с инкапсулированным в него пакетом по заданному адресу. Маршрутизатор извлекает пакет из прибывшего кадра и после необходимой обработки передает пакет для дальнейшей транспортировки в следующую сеть, предварительно упаковав его в новый кадр канального уровня в общем случае другой технологии. Таким

образом, сетевой уровень играет роль координатора, организующего совместную работу сетей, построенных на основе разных технологий.

В общем случае функции сетевого уровня шире, чем обеспечение обмена в пределах составной сети. Так, сетевой уровень решает задачу создания надежных и гибких барьеров на пути нежелательного трафика между сетями.

На сетевом уровне определяются два вида протоколов. Первый вид – маршрутизируемые протоколы – реализуют продвижение пакетов через сеть. Именно эти протоколы обычно имеют в виду, когда говорят о протоколах сетевого уровня. Однако часто к сетевому уровню относят и другой вид протоколов, называемых маршрутизирующими протоколами, или протоколами маршрутизации. С помощью этих протоколов маршрутизаторы собирают информацию о топологии межсетевых соединений, на основании которой осуществляется выбор маршрута продвижения пакетов.

Транспортный уровень (transport layer) обеспечивает приложениям или верхним уровням стека – прикладному, представления и сеансовому – передачу данных с той степенью надежности, которая им требуется. Модель OSI определяет пять классов транспортного сервиса от низшего класса 0 до высшего класса 4. Эти виды сервиса отличаются качеством предоставляемых услуг: срочностью, возможностью восстановления прерванной связи, наличием средств мультиплексирования нескольких соединений между различными прикладными протоколами через общий транспортный протокол, а главное – способностью к обнаружению и исправлению ошибок передачи, таких как искажение, потеря и дублирование пакетов.

Выбор класса сервиса транспортного уровня определяется:

1. тем, в какой степени задача обеспечения надежности решается самими приложениями и протоколами более высоких, чем транспортный, уровней.
2. насколько надежной является система транспортировки данных в сети, обеспечиваемая уровнями, расположенными ниже транспортного, – сетевым, канальным и физическим.

Если качество каналов передачи связи очень высокое и вероятность возникновения ошибок, не обнаруженных протоколами более низких уровней, невелика, то разумно воспользоваться одним из облегченных сервисов транспортного уровня, не обремененных многочисленными проверками, квитированием и другими приемами повышения надежности. Если же транспортные средства нижних уровней очень ненадежны, то целесообразно обратиться к наиболее развитому сервису транспортного уровня, который работает, используя максимум средств обнаружения и устранения ошибок, включая предварительное установление логического

соединения, контроль доставки сообщений по контрольным суммам и циклической нумерации пакетов, установление тайм-аутов доставки и т. п.

Все протоколы, начиная с транспортного уровня и выше, реализуются программными средствами конечных узлов сети – компонентами их сетевых операционных систем. В качестве примера транспортных протоколов можно привести протоколы TCP и UDP стека TCP/IP..

Протоколы нижних четырех уровней обобщенно называют сетевым транспортом, или транспортной подсистемой, так как они полностью решают задачу транспортировки сообщений с заданным уровнем качества в составных сетях с произвольной топологией и различными технологиями. Оставшиеся три верхних уровня решают задачи предоставления прикладных сервисов, используя нижележащую транспортную подсистему.

Сеансовый уровень (session layer) обеспечивает управление взаимодействием сторон: фиксирует, какая из сторон является активной в настоящий момент, и предоставляет средства синхронизации сеанса. Эти средства позволяют в ходе длинных передач сохранять информацию о состоянии этих передач в виде контрольных точек, чтобы в случае отказа можно было вернуться назад к последней контрольной точке, а не начинать все с начала. На практике немногие приложения используют сеансовый уровень, и он редко реализуется в виде отдельных протоколов. Функции этого уровня часто объединяют с функциями прикладного уровня и реализуют в одном протоколе.

Уровень представления (presentation layer), как явствует из его названия, обеспечивает представление передаваемой по сети информации, не меняя при этом ее содержания. За счет уровня представления информация, передаваемая прикладным уровнем одной системы, всегда понятна прикладному уровню другой системы. С помощью средств данного уровня протоколы прикладных уровней могут преодолеть синтаксические различия в представлении данных или же различия в кодах символов, например кодов ASCII. На этом уровне могут выполняться шифрование и дешифрирование данных, благодаря которым секретность обмена данными обеспечивается сразу для всех прикладных служб. Примером является протокол SSL (Secure Socket Layer – слой защищенных сокетов), который обеспечивает секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

Прикладной уровень (application layer) – это в действительности просто набор разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры или гипертекстовые веб-страницы, а также организуют свою совместную работу, например, по протоколу электронной почты.

Единица данных, которой оперирует прикладной уровень, обычно называется сообщением.

Существует очень большое разнообразие протоколов и соответствующих служб прикладного уровня. Например, несколько наиболее распространенных реализаций сетевых файловых служб: NFS и FTP в стеке TCP/IP, SMB в Microsoft Windows.

Для представления структуры средств межсетевого взаимодействия сетей с коммутацией каналов также используется многоуровневый подход, в соответствии с которым существуют протоколы нескольких уровней, образующих иерархию. Однако общей справочной модели, подобной модели OSI, для сетей с коммутацией каналов не существует. Для сетей с коммутацией пакетов сети с коммутацией каналов предоставляют сервис физического уровня, несмотря на то, что сами они устроены достаточно сложно и поддерживают собственную иерархию протоколов.

Сетезависимые и сетезависимые уровни

Функции всех уровней модели OSI могут быть отнесены к одной из двух групп: либо к функциям, зависящим от конкретной технической реализации сети, либо к функциям, ориентированным на работу с приложениями.

Три нижних уровня — физический, канальный и сетевой — являются сетезависимыми, то есть протоколы этих уровней тесно связаны с технической реализацией сети и используемым коммуникационным оборудованием. Например, переход на оборудование FDDI означает полную смену протоколов физического и канального уровней во всех узлах сети.

Три верхних уровня — прикладной, представительный и сеансовый — ориентированы на приложения и мало зависят от технических особенностей построения сети. На протоколы этих уровней не влияют изменения в топологии сети, замена оборудования или переход на другую сетевую технологию. Так, переход от Ethernet на высокоскоростную технологию 100VG-AnyLAN не потребует никаких изменений в программных средствах, реализующих функции прикладного, представительного и сеансового уровней.

Транспортный уровень является промежуточным, он скрывает все детали функционирования нижних уровней от верхних. Это позволяет разрабатывать приложения, не зависящие от технических средств непосредственной транспортировки сообщений.

Компьютер с установленной на нем сетевой ОС взаимодействует с другим компьютером с помощью протоколов всех семи уровней. Это взаимодействие компьютеры осуществляют опосредованно через различные коммуникационные устройства: концентраторы, модемы, мосты, коммутаторы, маршрутизаторы, мультиплексоры. В зависимости от типа

коммуникационное устройство может работать либо только на физическом уровне (повторитель), либо на физическом и канальном (мост), либо на физическом, канальном и сетевом, иногда захватывая и транспортный уровень (маршрутизатор).

Модель OSI представляет хотя и очень важную, но только одну из многих моделей коммуникаций. Эти модели и связанные с ними стеки протоколов могут отличаться количеством уровней, их функциями, форматами сообщений, службами, поддерживаемыми на верхних уровнях, и прочими параметрами.

Тема 2. Стек протоколов TCP/IP

Transmission Control Protocol/Internet Protocol (TCP/IP) - это промышленный стандарт стека протоколов, разработанный для глобальных сетей. Стандарты TCP/IP опубликованы в серии документов, названных Request for Comment (RFC). Документы RFC описывают внутреннюю работу сети Internet. Некоторые RFC описывают сетевые сервисы или протоколы и их реализацию, в то время как другие обобщают условия применения. Стандарты TCP/IP всегда публикуются в виде документов RFC, но не все RFC определяют стандарты.

Стек был разработан по инициативе Министерства обороны США (Department of Defence, DoD) для связи экспериментальной сети ARPAnet с другими спутниковыми сетями как набор общих протоколов для разнородной вычислительной среды. Сеть ARPA поддерживала разработчиков и исследователей в военных областях. В сети ARPA связь между двумя компьютерами осуществлялась с использованием протокола Internet Protocol (IP), который и по сей день является одним из основных в стеке TCP/IP и фигурирует в названии стека.

Большой вклад в развитие стека TCP/IP внес университет Беркли, реализовав протоколы стека в своей версии ОС UNIX. Широкое распространение ОС UNIX привело и к широкому распространению протокола IP и других протоколов стека. На этом же стеке работает всемирная информационная сеть Internet, чье подразделение Internet Engineering Task Force (IETF) вносит основной вклад в совершенствование стандартов стека, публикуемых в форме спецификаций RFC.

Лидирующая роль стека TCP/IP объясняется следующими его свойствами:

- наиболее заверченный стандартный и в то же время популярный стек сетевых протоколов, имеющий многолетнюю историю.
- почти все большие сети передают основную часть своего трафика с помощью протокола TCP/IP.
- метод получения доступа к сети Internet.

- стек служит основой для создания intranet- корпоративной сети, использующей транспортные услуги Internet и гипертекстовую технологию WWW, разработанную в Internet.
- все современные операционные системы поддерживают стек TCP/IP.
- гибкая технология для соединения разнородных систем, как на уровне транспортных подсистем, так и на уровне прикладных сервисов.
- устойчивая масштабируемая межплатформенная среда для приложений клиент-сервер.

Структура стека TCP/IP. Краткая характеристика протоколов

Так как стек TCP/IP был разработан до появления модели взаимодействия открытых систем ISO/OSI, то, хотя он также имеет многоуровневую структуру, соответствие уровней стека TCP/IP уровням модели OSI достаточно условно. Структура протоколов TCP/IP приведена на рисунке 2.1. Протоколы TCP/IP делятся на 4 уровня. Самый нижний (уровень IV) соответствует физическому и каналному уровням модели OSI. Этот уровень в протоколах TCP/IP не регламентируется, но поддерживает все популярные стандарты физического и канального уровня: для локальных сетей это Ethernet, Token Ring, FDDI, Fast Ethernet, 100VG-AnyLAN, для глобальных сетей - протоколы соединений "точка-точка" SLIP и PPP, протоколы территориальных сетей с коммутацией пакетов X.25, frame relay. Разработана также специальная спецификация, определяющая использование технологии ATM в качестве транспорта канального уровня. Обычно при появлении новой технологии локальных или глобальных сетей она быстро включается в стек TCP/IP за счет разработки соответствующего RFC, определяющего метод инкапсуляции пакетов IP в ее кадры.

Уровень III - это уровень межсетевого взаимодействия, который занимается передачей пакетов с использованием различных транспортных технологий локальных сетей, территориальных сетей, линий специальной связи и т. п.

В качестве основного протокола сетевого уровня (в терминах модели OSI) в стеке используется протокол IP, который изначально проектировался как протокол передачи пакетов в составных сетях, состоящих из большого количества локальных сетей, объединенных как локальными, так и глобальными связями. Поэтому протокол IP хорошо работает в сетях со сложной топологией, рационально используя наличие в них подсистем и экономно расходуя пропускную способность низкоскоростных линий связи. Протокол IP является дейтаграммным протоколом, то есть он не гарантирует доставку пакетов до узла назначения, но старается это сделать.

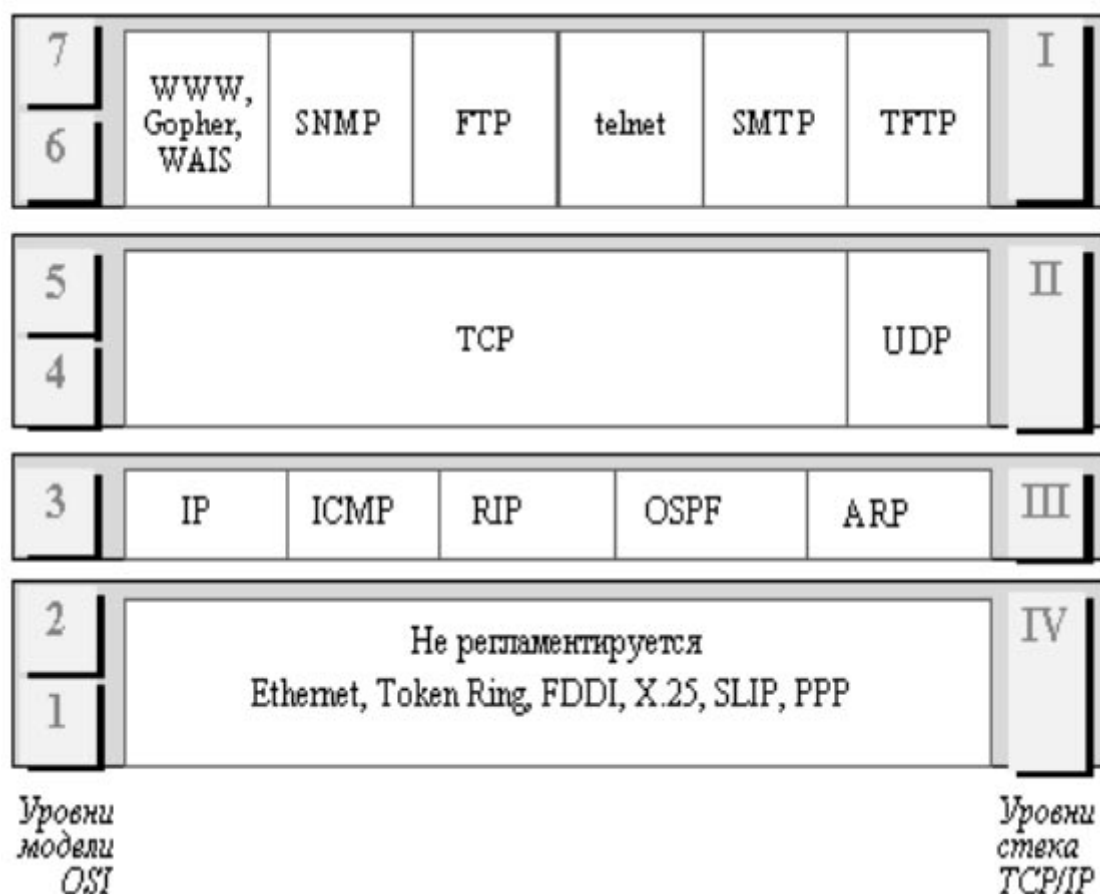


Рис. 2.1. Стек TCP/IP

К уровню межсетевого взаимодействия относятся и все протоколы, связанные с составлением и модификацией таблиц маршрутизации, такие как протоколы сбора маршрутной информации RIP (Routing Internet Protocol) и OSPF (Open Shortest Path First), а также протокол межсетевых управляющих сообщений ICMP (Internet Control Message Protocol). Последний протокол предназначен для обмена информацией об ошибках между маршрутизаторами сети и узлом - источником пакета. С помощью специальных пакетов ICMP сообщается о невозможности доставки пакета, о превышении времени жизни или продолжительности сборки пакета из фрагментов, об аномальных величинах параметров, об изменении маршрута пересылки и типа обслуживания, о состоянии системы и т.п.

Уровень II называется основным. На этом уровне функционируют протокол управления передачей TCP (Transmission Control Protocol) и протокол дейтаграмм пользователя UDP (User Datagram Protocol). Протокол TCP обеспечивает надежную передачу сообщений между удаленными прикладными процессами за счет образования виртуальных соединений. Протокол UDP обеспечивает передачу прикладных пакетов дейтаграммным способом, как и IP, и выполняет только функции связующего звена между сетевым протоколом и многочисленными прикладными процессами.

Уровень I называется прикладным. За долгие годы использования в сетях различных стран и организаций стек TCP/IP накопил большое количество протоколов и сервисов прикладного уровня. К ним относятся такие широко используемые протоколы, как протокол копирования файлов FTP, протокол эмуляции терминала telnet, почтовый протокол SMTP, используемый в электронной почте сети Internet, гипертекстовые сервисы доступа к удаленной информации, такие как WWW и многие другие. Остановимся несколько подробнее на некоторых из них.

Протокол пересылки файлов FTP (File Transfer Protocol) реализует удаленный доступ к файлу. Для того, чтобы обеспечить надежную передачу, FTP использует в качестве транспорта протокол с установлением соединений - TCP. Кроме пересылки файлов протокол FTP предлагает и другие услуги. Так, пользователю предоставляется возможность интерактивной работы с удаленной машиной, например, он может распечатать содержимое ее каталогов. Наконец, FTP выполняет аутентификацию пользователей. Прежде, чем получить доступ к файлу, в соответствии с протоколом пользователи должны сообщить свое имя и пароль. Для доступа к публичным каталогам FTP-архивов Internet парольная аутентификация не требуется, и ее обходят за счет использования для такого доступа предопределенного имени пользователя Anonymous.

В стеке TCP/IP протокол FTP предлагает наиболее широкий набор услуг для работы с файлами, однако он является и самым сложным для программирования. Приложения, которым не требуются все возможности FTP, могут использовать другой, более экономичный протокол - простейший протокол пересылки файлов TFTP (Trivial File Transfer Protocol). Этот протокол реализует только передачу файлов, причем в качестве транспорта используется более простой, чем TCP, протокол без установления соединения - UDP.

Протокол telnet обеспечивает передачу потока байтов между процессами, а также между процессом и терминалом. Наиболее часто этот протокол используется для эмуляции терминала удаленного компьютера. При использовании сервиса telnet пользователь фактически управляет удаленным компьютером так же, как и локальный пользователь, поэтому такой вид доступа требует хорошей защиты. Поэтому серверы telnet всегда используют как минимум аутентификацию по паролю, а иногда и более мощные средства защиты, например, систему Kerberos.

Протокол SNMP (Simple Network Management Protocol) используется для организации сетевого управления. Изначально протокол SNMP был разработан для удаленного контроля и управления маршрутизаторами Internet, которые традиционно часто называют также шлюзами. С ростом популярности протокол SNMP стали применять и для управления любым коммуникационным оборудованием - концентраторами, мостами,

сетевыми адаптерами и т.д. и т.п. Проблема управления в протоколе SNMP разделяется на две задачи.

Первая задача связана с передачей информации. Протоколы передачи управляющей информации определяют процедуру взаимодействия SNMP-агента, работающего в управляемом оборудовании, и SNMP-монитора, работающего на компьютере администратора, который часто называют также консолью управления. Протоколы передачи определяют форматы сообщений, которыми обмениваются агенты и монитор.

Вторая задача связана с контролируемыми переменными, характеризующими состояние управляемого устройства. Стандарты регламентируют, какие данные должны сохраняться и накапливаться в устройствах, имена этих данных и синтаксис этих имен. В стандарте SNMP определена спецификация информационной базы данных управления сетью. Эта спецификация, известная как база данных MIB (Management Information Base), определяет те элементы данных, которые управляемое устройство должно сохранять, и допустимые операции над ними.

Тема 3. Адресация в IPv4-сетях

Типы адресов: физический (MAC-адрес), сетевой (IPv4-адрес) и символьный (DNS-имя)

Каждый компьютер в сети TCP/IP имеет адреса трех уровней:

Локальный адрес узла, определяемый технологией, с помощью которой построена отдельная сеть, в которую входит данный узел. Для узлов, входящих в локальные сети - это MAC-адрес сетевого адаптера или порта маршрутизатора, например, 11-A0-17-3D-BC-01. Эти адреса назначаются производителями оборудования и являются уникальными адресами, так как управляются централизованно. Для всех существующих технологий локальных сетей MAC-адрес имеет формат 6 байтов: старшие 3 байта - идентификатор фирмы производителя, а младшие 3 байта назначаются уникальным образом самим производителем. Для узлов, входящих в глобальные сети, такие как X.25 или frame relay, локальный адрес назначается администратором глобальной сети.

IP-адрес, состоящий из 4 байт, например, 109.26.17.100. Этот адрес используется на сетевом уровне. Он назначается администратором во время конфигурирования компьютеров и маршрутизаторов. IP-адрес состоит из двух частей: номера сети и номера узла. Номер сети может быть выбран администратором произвольно, либо назначен по рекомендации специального подразделения Internet (Network Information Center, NIC), если сеть должна работать как составная часть Internet. Обычно провайдеры услуг Internet получают диапазоны адресов у подразделений NIC, а затем распределяют их между своими абонентами.

Номер узла в протоколе IP назначается независимо от локального адреса узла. Деление IP-адреса на поле номера сети и номера узла - гибкое, и граница между этими полями может устанавливаться весьма произвольно. Узел может входить в несколько IP-сетей. В этом случае узел должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

Символьный идентификатор-имя, например, *serv1.ibm.com*. Этот адрес назначается администратором и состоит из нескольких частей, например, имени машины, имени организации, имени домена. Такой адрес, называемый также DNS-именем, используется на прикладном уровне, например, в протоколах FTP или telnet.

Три основных класса IPv4-адресов

На рисунке 3.1 показана структура IPv4-адреса.

Класс А

0	N сети			N узла
---	--------	--	--	--------

Класс В

1	0	N сети		N узла
---	---	--------	--	--------

Класс С

1	1	0	N сети		N узла
---	---	---	--------	--	--------

Класс D

1	1	1	0	адрес группы multicast	
---	---	---	---	------------------------	--

Класс E

1	1	1	1	0	зарезервирован
---	---	---	---	---	----------------

Рис. 3.1. Структура IPv4-адреса

IPv4-адрес имеет длину 4 байта и обычно записывается в виде четырех чисел, представляющих значения каждого байта в десятичной форме, и разделенных точками, например:

128.10.2.30 - традиционная десятичная форма представления адреса,

10000000 00001010 00000010 00011110 - двоичная форма представления этого же адреса.

Адрес состоит из двух логических частей - номера сети и номера узла в сети. Какая часть адреса относится к номеру сети, а какая к номеру узла, определяется значениями первых битов адреса:

Если адрес начинается с 0, то сеть относят к классу А, и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса А имеют номера в диапазоне от 1 до 126. (Номер 0 не используется, а номер 127 зарезервирован для специальных целей, о чем

будет сказано ниже.) В сетях класса А количество узлов должно быть больше 216 , но не превышать 224.

Если первые два бита адреса равны 10, то сеть относится к классу В и является сетью средних размеров с числом узлов 28 - 216. В сетях класса В под адрес сети и под адрес узла отводится по 16 битов, то есть по 2 байта.

Если адрес начинается с последовательности 110, то это сеть класса С с числом узлов не больше 28. Под адрес сети отводится 24 бита, а под адрес узла - 8 битов.

Если адрес начинается с последовательности 1110, то он является адресом класса D и обозначает особый, групповой адрес - multicast. Если в пакете в качестве адреса назначения указан адрес класса D, то такой пакет должны получить все узлы, которым присвоен данный адрес.

Если адрес начинается с последовательности 11110, то это адрес класса E, он зарезервирован для будущих применений.

В таблице приведены диапазоны номеров сетей, соответствующих каждому классу сетей.

Класс	Наименьший адрес	Наибольший адрес
A	01.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Соглашения о специальных адресах: broadcast, multicast, loopback

В протоколе IPv4 существует несколько соглашений об особой интерпретации IP-адресов:

если IPv4-адрес состоит только из двоичных нулей,

0 0 0 0	0 0 0 0

то он обозначает адрес того узла, который сгенерировал этот пакет; если в поле номера сети стоят 0,

0 0 0 00	Номер узла
----------------	------------

то по умолчанию считается, что этот узел принадлежит той же самой сети, что и узел, который отправил пакет;

если все двоичные разряды IP-адреса равны 1,

```
1 1 1 1 .....1 1
```



то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета.

Такая рассылка называется ограниченным широковещательным сообщением (limited broadcast);

Номер сети	1111.....11
------------	-------------

если в поле адреса назначения стоят сплошные 1, то пакет, имеющий такой адрес рассылается всем узлам сети с заданным номером. Такая рассылка называется широковещательным сообщением (broadcast);

адрес 127.0.0.1 зарезервирован для организации обратной связи при тестировании работы программного обеспечения узла без реальной отправки пакета по сети. Этот адрес имеет название loopback.

Уже упоминавшаяся форма группового IP-адреса - multicast - означает, что данный пакет должен быть доставлен сразу нескольким узлам, которые образуют группу с номером, указанным в поле адреса. Узлы сами идентифицируют себя, то есть определяют, к какой из групп они относятся. Один и тот же узел может входить в несколько групп. Такие сообщения в отличие от широковещательных называются мультивещательными. Групповой адрес не делится на поля номера сети и узла и обрабатывается маршрутизатором особым образом.

В протоколе IP нет понятия широковещательности в том смысле, в котором оно используется в протоколах канального уровня локальных сетей, когда данные должны быть доставлены абсолютно всем узлам. Как ограниченный широковещательный IP-адрес, так и широковещательный IP-адрес имеют пределы распространения в интерсети - они ограничены либо сетью, к которой принадлежит узел - источник пакета, либо сетью, номер которой указан в адресе назначения. Поэтому деление сети с помощью маршрутизаторов на части локализует широковещательный шторм пределами одной из составляющих общую сеть частей просто потому, что нет способа адресовать пакет одновременно всем узлам всех сетей составной сети.

Отображение физических адресов на IP-адреса: протоколы ARP и RARP

В протоколе IP-адрес узла, то есть адрес компьютера или порта маршрутизатора, назначается произвольно администратором сети и прямо не связан с его локальным адресом, как это сделано, например, в протоколе IPX. Подход, используемый в IP, удобно использовать в крупных сетях и по причине его независимости от формата локального адреса, и по причине стабильности, так как в противном случае, при смене на

компьютере сетевого адаптера это изменение должны бы были учитывать все адресаты всемирной сети Internet (в том случае, конечно, если сеть подключена к Internet'у).

Локальный адрес используется в протоколе IP только в пределах локальной сети при обмене данными между маршрутизатором и узлом этой сети. Маршрутизатор, получив пакет для узла одной из сетей, непосредственно подключенных к его портам, должен для передачи пакета сформировать кадр в соответствии с требованиями принятой в этой сети технологии и указать в нем локальный адрес узла, например его MAC-адрес. В пришедшем пакете этот адрес не указан, поэтому перед маршрутизатором встает задача поиска его по известному IP-адресу, который указан в пакете в качестве адреса назначения. С аналогичной задачей сталкивается и конечный узел, когда он хочет отправить пакет в удаленную сеть через маршрутизатор, подключенный к той же локальной сети, что и данный узел.

Для определения локального адреса по IP-адресу используется протокол разрешения адреса Address Resolution Protocol, ARP. Протокол ARP работает различным образом в зависимости от того, какой протокол канального уровня работает в данной сети - протокол локальной сети (Ethernet, Token Ring, FDDI) с возможностью широковещательного доступа одновременно ко всем узлам сети, или же протокол глобальной сети (X.25, frame relay), как правило, не поддерживающий широковещательный доступ. Существует также протокол, решающий обратную задачу - нахождение IP-адреса по известному локальному адресу. Он называется реверсивный ARP - RARP (Reverse Address Resolution Protocol) и используется при старте бездисковых станций, не знающих в начальный момент своего IP-адреса, но знающих адрес своего сетевого адаптера.

В локальных сетях протокол ARP использует широковещательные кадры протокола канального уровня для поиска в сети узла с заданным IP-адресом.

Узел, которому нужно выполнить отображение IP-адреса на локальный адрес, формирует ARP запрос, вкладывает его в кадр протокола канального уровня, указывая в нем известный IP-адрес, и рассылает запрос широковещательно.

Все узлы локальной сети получают ARP запрос и сравнивают указанный там IP-адрес с собственным. В случае их совпадения узел формирует ARP-ответ, в котором указывает свой IP-адрес и свой локальный адрес и отправляет его уже направленно, так как в ARP запросе отправитель указывает свой локальный адрес. ARP-запросы и ответы используют один и тот же формат пакета.

Так как локальные адреса могут в различных типах сетей иметь различную длину, то формат пакета протокола ARP зависит от типа сети.

На рисунке 3.2 показан формат пакета протокола ARP для передачи по сети Ethernet.

Тип сети		Тип протокола
Длина локального адреса	Длина сетевого адреса	Операция
Локальный адрес отправителя (байты 0 - 3)		
Локальный адрес отправителя (байты 4 - 5)		IP-адрес отправителя (байты 0-1)
IP-адрес отправителя (байты 2-3)		Искомый локальный адрес (байты 0 - 1)
Искомый локальный адрес (байты 2-5)		
Искомый IP-адрес (байты 0 - 3)		

Рис. 3.2. Формат пакета протокола ARP

В поле типа сети для сетей Ethernet указывается значение 1. Поле типа протокола позволяет использовать пакеты ARP не только для протокола IP, но и для других сетевых протоколов. Для IP значение этого поля равно 080016.

Длина локального адреса для протокола Ethernet равна 6 байтам, а длина IP-адреса - 4 байтам. В поле операции для ARP запросов указывается значение 1 для протокола ARP и 2 для протокола RARP.

Узел, отправляющий ARP-запрос, заполняет в пакете все поля, кроме поля искомого локального адреса (для RARP-запроса не указывается искомый IP-адрес). Значение этого поля заполняется узлом, опознавшим свой IP-адрес.

В глобальных сетях администратору сети чаще всего приходится вручную формировать ARP-таблицы, в которых он задает, например, соответствие IP-адреса адресу узла сети X.25, который имеет смысл локального адреса. В последнее время наметилась тенденция автоматизации работы протокола ARP и в глобальных сетях. Для этой цели среди всех маршрутизаторов, подключенных к какой-либо глобальной сети, выделяется специальный маршрутизатор, который ведет ARP-таблицу для всех остальных узлов и маршрутизаторов этой сети. При таком централизованном подходе для всех узлов и маршрутизаторов вручную нужно задать только IP-адрес и локальный адрес выделенного маршрутизатора. Затем каждый узел и маршрутизатор регистрирует свои адреса в выделенном маршрутизаторе, а при необходимости установления соответствия между IP-адресом и локальным адресом узел обращается к выделенному маршрутизатору с запросом и автоматически получает ответ без участия администратора.

Отображение символьных адресов на IP-адреса: служба DNS

DNS (Domain Name System) - это распределенная база данных, поддерживающая иерархическую систему имен для идентификации узлов в сети Internet. Служба DNS предназначена для автоматического поиска IP-адреса по известному символьному имени узла. Спецификация DNS определяется стандартами RFC 1034 и 1035. DNS требует статической конфигурации своих таблиц, отображающих имена компьютеров в IP-адрес.

Протокол DNS является служебным протоколом прикладного уровня. Этот протокол несимметричен - в нем определены DNS-серверы и DNS-клиенты. DNS-серверы хранят часть распределенной базы данных о соответствии символьных имен и IP-адресов. Эта база данных распределена по административным доменам сети Internet. Клиенты сервера DNS знают IP-адрес сервера DNS своего административного домена и по протоколу IP передают запрос, в котором сообщают известное символьное имя и просят вернуть соответствующий ему IP-адрес.

Если данные о запрошенном соответствии хранятся в базе данного DNS-сервера, то он сразу посылает ответ клиенту, если же нет - то он посылает запрос DNS-серверу другого домена, который может сам обработать запрос, либо передать его другому DNS-серверу. Все DNS-серверы соединены иерархически, в соответствии с иерархией доменов сети Internet. Клиент опрашивает эти серверы имен, пока не найдет нужные отображения. Этот процесс ускоряется из-за того, что серверы имен постоянно кэшируют информацию, предоставляемую по запросам. Клиентские компьютеры могут использовать в своей работе IP-адреса нескольких DNS-серверов, для повышения надежности своей работы.

База данных DNS имеет структуру дерева, называемого доменным пространством имен, в котором каждый домен (узел дерева) имеет имя и может содержать поддомены. Имя домена идентифицирует его положение в этой базе данных по отношению к родительскому домену, причем точки в имени отделяют части, соответствующие узлам домена.

Корень базы данных DNS управляется центром Internet Network Information Center. Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международному стандарту ISO 3166. Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, а для различных типов организаций используются следующие аббревиатуры:

- com - коммерческие организации (например, microsoft.com);
- edu - образовательные (например, mit.edu);
- gov - правительственные организации (например, nsf.gov);
- org - некоммерческие организации (например, fidonet.org);
- net - организации, поддерживающие сети (например, nsf.net).

Каждый домен DNS администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Каждый домен имеет уникальное имя, а каждый из поддоменов имеет уникальное имя внутри своего домена. Имя домена может содержать до 63 символов. Каждый хост в сети Internet однозначно определяется своим полным доменным именем (fully qualified domain name, FQDN), которое включает имена всех доменов по направлению от хоста к корню. Пример полного DNS-имени: *citint.dol.ru*.

Автоматизация процесса назначения IP-адресов узлам сети - протокол DHCP

IP-адреса могут назначаться администратором сети вручную. Протокол Dynamic Host Configuration Protocol (DHCP) был разработан для того, чтобы освободить администратора от этого. Основным назначением DHCP является динамическое назначение IP-адресов. Однако, кроме динамического, DHCP может поддерживать и более простые способы ручного и автоматического статического назначения адресов.

В ручной процедуре назначения адресов активное участие принимает администратор, который предоставляет DHCP-серверу информацию о соответствии IP-адресов физическим адресам или другим идентификаторам клиентов. Эти адреса сообщаются клиентам в ответ на их запросы к DHCP-серверу.

При автоматическом статическом способе DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула наличных IP-адресов без вмешательства оператора. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера. Между идентификатором клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первичного назначения сервером DHCP IP-адреса клиенту. При всех последующих запросах сервер возвращает тот же самый IP-адрес.

При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, что дает возможность впоследствии повторно использовать IP-адреса другими компьютерами. Динамическое разделение адресов позволяет строить IP-сеть, количество узлов в которой намного превышает количество имеющихся в распоряжении администратора IP-адресов.

DHCP обеспечивает надежный и простой способ конфигурации сети TCP/IP, гарантируя отсутствие конфликтов адресов за счет централизованного управления их распределением. Администратор управляет процессом назначения адресов с помощью параметра

"продолжительности аренды" (lease duration), которая определяет, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от сервера DHCP в аренду.

Примером работы протокола DHCP может служить ситуация, когда компьютер, являющийся клиентом DHCP, удаляется из подсети. При этом назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс. Это свойство очень важно для мобильных пользователей.

Протокол DHCP использует модель клиент-сервер. Во время старта системы компьютер-клиент DHCP, находящийся в состоянии "инициализация", посылает сообщение discover (исследовать), которое широковещательно распространяется по локальной сети и передается всем DHCP-серверам частной интерсети. Каждый DHCP-сервер, получивший это сообщение, отвечает на него сообщением offer (предложение), которое содержит IP-адрес и конфигурационную информацию.

Компьютер-клиент DHCP переходит в состояние "выбор" и собирает конфигурационные предложения от DHCP-серверов. Затем он выбирает одно из этих предложений, переходит в состояние "запрос" и отправляет сообщение request (запрос) тому DHCP-серверу, чье предложение было выбрано.

Выбранный DHCP-сервер посылает сообщение DHCP-acknowledgment (подтверждение), содержащее тот же IP-адрес, который уже был послан ранее на стадии исследования, а также параметр аренды для этого адреса. Кроме того, DHCP-сервер посылает параметры сетевой конфигурации. После того, как клиент получит это подтверждение, он переходит в состояние "связь", находясь в котором он может принимать участие в работе сети TCP/IP. Компьютеры-клиенты, которые имеют локальные диски, сохраняют полученный адрес для использования при последующих стартах системы. При приближении момента истечения срока аренды адреса компьютер пытается обновить параметры аренды у DHCP-сервера, а если этот IP-адрес не может быть выделен снова, то ему возвращается другой IP-адрес.

В протоколе DHCP описывается несколько типов сообщений, которые используются для обнаружения и выбора DHCP-серверов, для запросов информации о конфигурации, для продления и досрочного прекращения лицензии на IP-адрес. Все эти операции направлены на то, чтобы освободить администратора сети от утомительных рутинных операций по конфигурированию сети.

Однако использование DHCP несет в себе и некоторые проблемы. Во-первых, это проблема согласования информационной адресной базы в службах DHCP и DNS. Как известно, DNS служит для преобразования

символьных имен в IP-адреса. Если IP-адреса будут динамически изменяться сервером DHCP, то эти изменения необходимо также динамически вносить в базу данных сервера DNS. Хотя протокол динамического взаимодействия между службами DNS и DHCP уже реализован некоторыми фирмами (так называемая служба Dynamic DNS), стандарт на него пока не принят.

Во-вторых, нестабильность IP-адресов усложняет процесс управления сетью. Системы управления, основанные на протоколе SNMP, разработаны с расчетом на статичность IP-адресов. Аналогичные проблемы возникают и при конфигурировании фильтров маршрутизаторов, которые оперируют с IP-адресами.

Наконец, централизация процедуры назначения адресов снижает надежность системы: при отказе DHCP-сервера все его клиенты оказываются не в состоянии получить IP-адрес и другую информацию о конфигурации. Последствия такого отказа могут быть уменьшены путем использования в сети нескольких серверов DHCP, каждый из которых имеет свой пул IP-адресов.

Тема 4. Протокол межсетевого взаимодействия IP

Основу транспортных средств стека протоколов TCP/IP составляет протокол межсетевого взаимодействия - Internet Protocol (IP). К основным функциям протокола IP относятся:

перенос между сетями различных типов адресной информации в унифицированной форме,

сборка и разборка пакетов при передаче их между сетями с различным максимальным значением длины пакета.

Формат пакета IP Версия 4 (IPv4) показан на рис. 4.1.

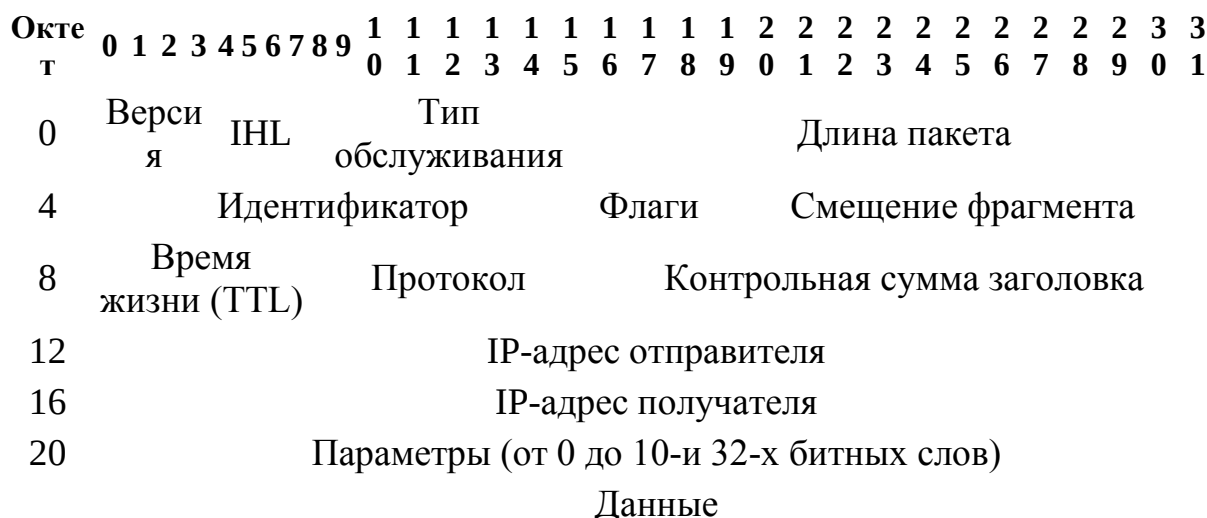


Рис.4.1. Формат пакета IPv4

Пакет IP состоит из заголовка и поля данных. Заголовок пакета имеет следующие поля:

Версия — для IPv4 значение поля должно быть равно 4.

IHL — (Internet Header Length) длина заголовка IP-пакета в 32-битных словах (dword). Именно это поле указывает на начало блока данных (англ. payload — полезный груз) в пакете. Минимальное корректное значение для этого поля равно 5.

Длина пакета - длина пакета в октетах, включая заголовок и данные. Минимальное корректное значение для этого поля равно 20, максимальное 65535.

Идентификатор — значение, назначаемое отправителем пакета и предназначенное для определения корректной последовательности фрагментов при сборке пакета. Для фрагментированного пакета все фрагменты имеют одинаковый идентификатор.

3 бита флагов. Первый бит должен быть всегда равен нулю, второй бит DF (don't fragment) определяет возможность фрагментации пакета и третий бит MF (more fragments) показывает, не является ли этот пакет последним в цепочке пакетов.

Смещение фрагмента — значение, определяющее позицию фрагмента в потоке данных. Смещение задается количеством восьми байтовых блоков, поэтому это значение требует умножения на 8 для перевода в байты.

Время жизни (TTL) — число маршрутизаторов, которые должен пройти этот пакет. При прохождении маршрутизатора это число уменьшается на единицу. Если значения этого поля равно нулю то, пакет должен быть отброшен и отправителю пакета может быть послано сообщение Time Exceeded (ICMP код 11 тип 0).

Протокол — идентификатор интернет-протокола следующего уровня указывает, данные какого протокола содержит пакет, например, TCP или ICMP (см. IANA protocol numbers и RFC 1700). В IPv6 называется «Next Header».

Контрольная сумма заголовка — вычисляется в соответствии с RFC 1071.

Поля Адрес источника (SOURCE IP ADDRESS) и Адрес назначения (DESTINATION IP ADDRESS) имеют одинаковую длину - 32 бита, и одинаковую структуру.

Поле Параметры является необязательным и используется обычно только при отладке сети. Это поле состоит из нескольких подполей, каждое из которых может быть одного из восьми predetermined типов.

В этих подполях можно указывать точный маршрут прохождения маршрутизаторов, регистрировать проходимые пакетом маршрутизаторы, помещать данные системы безопасности, а также временные отметки.

Так как число подполей может быть произвольным, то в конце поля Резерв должно быть добавлено несколько байт для выравнивания заголовка пакета по 32-битной границе.

Максимальная длина поля данных пакета ограничена разрядностью поля, определяющего эту величину, и составляет 65535 байтов, однако при передаче по сетям различного типа длина пакета выбирается с учетом максимальной длины пакета протокола нижнего уровня, несущего IP-пакеты. Если это кадры Ethernet, то выбираются пакеты с максимальной длиной в 1500 байтов, уместяющиеся в поле данных кадра Ethernet.

Управление фрагментацией

Протоколы транспортного уровня (протоколы TCP или UDP), пользующиеся сетевым уровнем для отправки пакетов, считают, что максимальный размер поля данных IP-пакета равен 65535, и поэтому могут передать ему сообщение такой длины для транспортировки через интернет.

В функции уровня IP входит разбиение слишком длинного для конкретного типа составляющей сети сообщения на более короткие пакеты с созданием соответствующих служебных полей, нужных для последующей сборки фрагментов в исходное сообщение.

В большинстве типов локальных и глобальных сетей определяется такое понятие как максимальный размер поля данных кадра или пакета, в которые должен инкапсулировать свой пакет протокол IP.

Эту величину обычно называют максимальной единицей транспортировки - Maximum Transfer Unit, MTU. Сети Ethernet имеют значение MTU, равное 1500 байт, сети FDDI - 4096 байт, а сети X.25 чаще всего работают с MTU в 128 байт.

IP-маршрутизаторы не собирают фрагменты пакетов в более крупные пакеты, даже если на пути встречается сеть, допускающая такое укрупнение.

Это связано с тем, что отдельные фрагменты сообщения могут перемещаться по интернету по различным маршрутам, поэтому нет гарантии, что все фрагменты проходят через какой-либо промежуточный маршрутизатор на их пути.

При приходе первого фрагмента пакета узел назначения запускает таймер, который определяет максимально допустимое время ожидания прихода остальных фрагментов этого пакета.

Если таймер истекает раньше прибытия последнего фрагмента, то все полученные к этому моменту фрагменты пакета отбрасываются, а в узел, пославший исходный пакет, направляется сообщение об ошибке с помощью протокола ICMP.

Маршрутизация с помощью IP-адресов

Рассматриваются принципы, на основании которых в сетях IP происходит выбор маршрута передачи пакета между сетями.

Сначала необходимо обратить внимание на тот факт, что не только маршрутизаторы, но и конечные узлы - компьютеры - должны принимать участие в выборе маршрута.

Пример, приведенный на рисунке 4.2, демонстрирует эту необходимость. Здесь в локальной сети имеется несколько маршрутизаторов, и компьютер должен выбирать, какому из них следует отправить пакет.



Рис. 4.2. Выбор маршрутизатора конечным узлом

Длина маршрута может существенно измениться в зависимости от того, какой маршрутизатор выберет компьютер для передачи своего пакета на сервер, расположенный, например, в Германии, если маршрутизатор 1 соединен выделенной линией с маршрутизатором в Копенгагене, а маршрутизатор 2 имеет спутниковый канал, соединяющий его с Токио.

Таблица 4.1. Таблица маршрутов

Адрес сети назначения	Адрес следующего маршрутизатора	Номер выходного порта Расстояние до сети назначения
56.0.0.0	198.21.17.7	120
56.0.0.0	213.34.12.4.	2130
116.0.0.0	213.34.12.4	21450
129.13.0.0	198.21.17.6	150
198.21.17.0	-	20
213. 34.12.0	-	10
default	198.21.17.7	1-

В стеке TCP/IP маршрутизаторы и конечные узлы принимают решения о том, кому передавать пакет для его успешной доставки узлу

назначения, на основании так называемых таблиц маршрутизации (routing tables).

Таблица 4.1. представляет собой типичный пример таблицы маршрутов, использующей IP-адреса сетей:

В этой таблице в столбце "Адрес сети назначения" указываются адреса всех сетей, которым данный маршрутизатор может передавать пакеты. В стеке TCP/IP принят так называемый одношаговый подход к оптимизации маршрута продвижения пакета (next-hop routing) - каждый маршрутизатор и конечный узел принимает участие в выборе только одного шага передачи пакета. Поэтому в каждой строке таблицы маршрутизации указывается не весь маршрут в виде последовательности IP-адресов маршрутизаторов, через которые должен пройти пакет, а только один IP-адрес - адрес следующего маршрутизатора, которому нужно передать пакет. Вместе с пакетом следующему маршрутизатору передается ответственность за выбор следующего шага маршрутизации. Одношаговый подход к маршрутизации означает распределенное решение задачи выбора маршрута. Это снимает ограничение на максимальное количество транзитных маршрутизаторов на пути пакета.

(Альтернативой одношаговому подходу является указание в пакете всей последовательности маршрутизаторов, которые пакет должен пройти на своем пути. Такой подход называется маршрутизацией от источника - Source Routing. В этом случае выбор маршрута производится конечным узлом или первым маршрутизатором на пути пакета, а все остальные маршрутизаторы только отрабатывают выбранный маршрут, осуществляя коммутацию пакетов, то есть передачу их с одного порта на другой. Алгоритм Source Routing применяется в сетях IP только для отладки, когда маршрут задается в поле Резерв (IP OPTIONS) пакета.)

В случае, если в таблице маршрутов имеется более одной строки, соответствующей одному и тому же адресу сети назначения, то при принятии решения о передаче пакета используется та строка, в которой указано наименьшее значение в поле "Расстояние до сети назначения".

При этом под расстоянием понимается любая метрика, используемая в соответствии с заданным в сетевом пакете классом сервиса. Это может быть количество транзитных маршрутизаторов в данном маршруте (количество хопов от *hop* - прыжок), время прохождения пакета по линиям связи, надежность линий связи, или другая величина, отражающая качество данного маршрута по отношению к конкретному классу сервиса. Если маршрутизатор поддерживает несколько классов сервиса пакетов, то таблица маршрутов составляется и применяется отдельно для каждого вида сервиса (критерия выбора маршрута).

Для отправки пакета следующему маршрутизатору требуется знание его локального адреса, но в стеке TCP/IP в таблицах маршрутизации принято использование только IP-адресов для сохранения их

универсального формата, не зависящего от типа сетей, входящих в интернет. Для нахождения локального адреса по известному IP-адресу необходимо воспользоваться протоколом ARP.

Конечный узел, как и маршрутизатор, имеет в своем распоряжении таблицу маршрутов унифицированного формата и на основании ее данных принимает решение, какому маршрутизатору нужно передавать пакет для сети N.

Решение о том, что этот пакет нужно вообще маршрутизировать, компьютер принимает в том случае, когда он видит, что адрес сети назначения пакета отличается от адреса его собственной сети (каждому компьютеру при конфигурировании администратор присваивает его IP-адрес или несколько IP-адресов, если компьютер одновременно подключен к нескольким сетям). Когда компьютер выбрал следующий маршрутизатор, то он просматривает кэш-таблицу адресов своего протокола ARP и, может быть, находит там соответствие IP-адреса следующего маршрутизатора его MAC-адресу. Если же нет, то по локальной сети передается широковещательный ARP-запрос и локальный адрес извлекается из ARP-ответа.

После этого компьютер формирует кадр протокола, используемого на выбранном порту, например, кадр Ethernet, в который помещает MAC-адрес маршрутизатора. Маршрутизатор принимает кадр Ethernet, извлекает из него пакет IP и просматривает свою таблицу маршрутизации для нахождения следующего маршрутизатора. При этом он выполняет те же действия, что и конечный узел.

Одношаговая маршрутизация обладает еще одним преимуществом - она позволяет сократить объем таблиц маршрутизации в конечных узлах и маршрутизаторах за счет использования в качестве номера сети назначения так называемого маршрута по умолчанию - *default*, который обычно занимает в таблице маршрутизации последнюю строку. Если в таблице маршрутизации есть такая запись, то все пакеты с номерами сетей, которые отсутствуют в таблице маршрутизации, передаются маршрутизатору, указанному в строке *default*. Поэтому маршрутизаторы часто хранят в своих таблицах ограниченную информацию о сетях интернета, пересылая пакеты для остальных сетей в порт и маршрутизатор, используемые по умолчанию. Подразумевается, что маршрутизатор, используемый по умолчанию, передаст пакет на магистральную сеть, а маршрутизаторы, подключенные к магистрали, имеют полную информацию о составе интернета.

Особенно часто приемом маршрутизации по умолчанию пользуются конечные узлы. Хотя они также в общем случае имеют в своем распоряжении таблицу маршрутизации, ее объем обычно незначителен, так как маршрутизация для компьютера - не основное занятие. Главная роль в маршрутизации пакетов в концепции протокола IP отводится,

естественно, маршрутизаторам, которые должны обладать гораздо более полными таблицами маршрутизации, чем конечные узлы.

Конечный узел часто вообще работает без таблицы маршрутизации, имея только сведения об IP-адресе маршрутизатора по умолчанию. При наличии одного маршрутизатора в локальной сети этот вариант - единственно возможный для всех конечных узлов. Но даже при наличии нескольких маршрутизаторов в локальной сети, когда проблема их выбора стоит перед конечным узлом, задание маршрута по умолчанию часто используется в компьютерах для сокращения объема их маршрутной таблицы.

Другим способом разгрузки компьютера от необходимости ведения больших таблиц маршрутизации является получение от маршрутизатора сведений о рациональном маршруте для какой-нибудь конкретной сети с помощью протокола ICMP.

Кроме маршрута *default*, в таблице маршрутизации могут встретиться два типа специальных записей - запись о специфичном для узла маршруте и запись об адресах сетей, непосредственно подключенных к портам маршрутизатора.

Специфичный для узла маршрут содержит вместо номера сети полный IP-адрес, то есть адрес, имеющий ненулевую информацию не только в поле номера сети, но и в поле номера узла. Предполагается, что для такого конечного узла маршрут должен выбираться не так, как для всех остальных узлов сети, к которой он относится. В случае, когда в таблице есть разные записи о продвижении пакетов для всей сети N и ее отдельного узла, имеющего адрес N,D, при поступлении пакета, адресованного узлу N,D, маршрутизатор отдаст предпочтение записи для N,D.

Записи в таблице маршрутизации, относящиеся к сетям, непосредственно подключенным к маршрутизатору, в поле "Расстояние до сети назначения" содержат нули.

Еще одним отличием работы маршрутизатора и конечного узла при выборе маршрута является способ построения таблицы маршрутизации. Если маршрутизаторы обычно автоматически создают таблицы маршрутизации, обмениваясь служебной информацией, то для конечных узлов таблицы маршрутизации создаются, как правило, вручную администраторами, и хранятся в виде постоянных файлов на дисках.

Существуют различные алгоритмы построения таблиц для одношаговой маршрутизации. Их можно разделить на три класса:

1. алгоритмы фиксированной маршрутизации,
2. алгоритмы простой маршрутизации,
3. алгоритмы адаптивной маршрутизации.

Независимо от алгоритма, используемого для построения таблицы маршрутизации, результат их работы имеет единый формат. За счет этого в

одной и той же сети различные узлы могут строить таблицы маршрутизации по своим алгоритмам, а затем обмениваться между собой недостающими данными, так как форматы этих таблиц фиксированы. Поэтому маршрутизатор, работающий по алгоритму адаптивной маршрутизации, может снабдить конечный узел, применяющий алгоритм фиксированной маршрутизации, сведениями о пути к сети, о которой конечный узел ничего не знает.

Фиксированная маршрутизация

Этот алгоритм применяется в сетях с простой топологией связей и основан на ручном составлении таблицы маршрутизации администратором сети. Алгоритм часто эффективно работает также для магистралей крупных сетей, так как сама магистраль может иметь простую структуру с очевидными наилучшими путями следования пакетов в подсети, присоединенные к магистрали.

Различают одномаршрутные таблицы, в которых для каждого адресата задан один путь, и многомаршрутные таблицы, определяющие несколько альтернативных путей для каждого адресата. При использовании многомаршрутных таблиц должно быть задано правило выбора одного из них. Чаще всего один путь является основным, а остальные - резервными.

Простая маршрутизация

Алгоритмы простой маршрутизации подразделяются на три подкласса:

Случайная маршрутизация - пакеты передаются в любом, случайном направлении, кроме исходного.

Лавинная маршрутизация - пакеты передаются во всех направлениях, кроме исходного (применяется в мостах для пакетов с неизвестным адресом доставки).

Маршрутизация по предыдущему опыту - таблицы маршрутов составляются на основании данных, содержащихся в проходящих через маршрутизатор пакетах. Именно так работают прозрачные мосты, собирая сведения об адресах узлов, входящих в сегменты сети. Такой способ маршрутизации обладает медленной адаптируемостью к изменениям топологии сети.

Адаптивная маршрутизация

Это основной вид алгоритмов маршрутизации, применяющихся маршрутизаторами в современных сетях со сложной топологией. Адаптивная маршрутизация основана на том, что маршрутизаторы

периодически обмениваются специальной топологической информацией об имеющихся в интерсети сетях, а также о связях между маршрутизаторами. Обычно учитывается не только топология связей, но и их пропускная способность и состояние.

Адаптивные протоколы позволяют всем маршрутизаторам собирать информацию о топологии связей в сети, оперативно отрабатывая все изменения конфигурации связей. Эти протоколы имеют распределенный характер, который выражается в том, что в сети отсутствуют какие-либо выделенные маршрутизаторы, которые бы собирали и обобщали топологическую информацию: эта работа распределена между всеми маршрутизаторами.

Структуризация сетей IP с помощью масок

Часто администраторы сетей испытывают неудобства, из-за того, что количество централизовано выделенных им номеров сетей недостаточно для того, чтобы структурировать сеть надлежащим образом, например, разместить все слабо взаимодействующие компьютеры по разным сетям.

В такой ситуации возможны два пути. Первый из них связан с получением от NIC дополнительных номеров сетей. Второй способ, употребляющийся более часто, связан с использованием так называемых масок, которые позволяют разделять одну сеть на несколько сетей.

Маска - это число, двоичная запись которого содержит единицы в тех разрядах, которые должны интерпретироваться как номер сети.

Например, для стандартных классов сетей маски имеют следующие значения:

255.0.0.0 - маска для сети класса А,
255.255.0.0 - маска для сети класса В,
255.255.255.0 - маска для сети класса С.

В масках, которые использует администратор для увеличения числа сетей, количество единиц в последовательности, определяющей границу номера сети, не обязательно должно быть кратным 8, чтобы повторять деление адреса на байты.

Пусть, например, маска имеет значение 255.255.192.0 (11111111 11111111 11000000 00000000). И пусть сеть имеет номер 129.44.0.0 (10000001 00101100 00000000 00000000), из которого видно, что она относится к классу В. После наложения маски на этот адрес число разрядов, интерпретируемых как номер сети, увеличилось с 16 до 18, то есть администратор получил возможность использовать вместо одного, централизованно заданного ему номера сети, четыре:

129.44.0.0 (10000001 00101100 00000000 00000000)
129.44.64.0 (10000001 00101100 01000000 00000000)
129.44.128.0 (10000001 00101100 10000000 00000000)

129.44.192.0 (10000001 00101100 11000000 00000000)

Например, IP-адрес 129.44.141.15 (10000001 00101100 10001101 00001111), который по стандартам IP задает номер сети 129.44.0.0 и номер узла 0.0.141.15, теперь, при использовании маски, будет интерпретироваться как пара:

129.44.128.0 - номер сети, 0.0. 13.15 - номер узла.

Таким образом, установив новое значение маски, можно заставить маршрутизатор по-другому интерпретировать IP-адрес. При этом два дополнительных последних бита номера сети часто интерпретируются как номера подсетей.

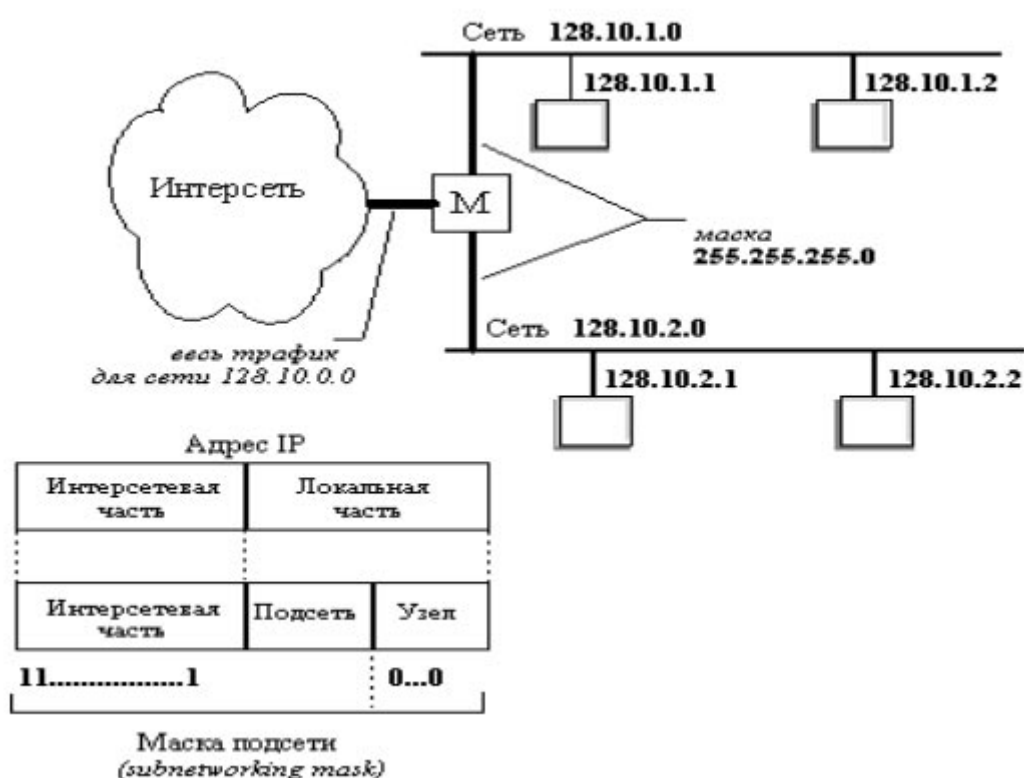


Рис. 4.3. Пример использования масок для структурирования сети

Пусть некоторая сеть относится к классу В и имеет адрес 128.10.0.0 (рис. 4.3). Этот адрес используется маршрутизатором, соединяющим сеть с остальной частью интерсети. И пусть среди всех станций сети есть станции, слабо взаимодействующие между собой. Их желательно было бы изолировать в разных сетях. Для этого сеть можно разделить на две сети, подключив их к соответствующим портам маршрутизатора, и задать для этих портов в качестве маски, например, число 255.255.255.0, то есть организовать внутри исходной сети с централизованно заданным номером две подсети класса С (можно было бы выбрать и другой размер для поля адреса подсети). Извне сеть по-прежнему будет выглядеть, как единая сеть класса В, а на местном уровне это будут две отдельные сети класса С.

Приходящий общий трафик будет разделяться местным маршрутизатором между подсетями. Необходимо заметить, что, если принимается решение об использовании механизма масок, то соответствующим образом должны быть сконфигурированы и маршрутизаторы, и компьютеры сети.

Тема 5. Протокол доставки пользовательских дейтаграмм UDP

Задачей протокола транспортного уровня UDP (User Datagram Protocol) является передача данных между прикладными процессами без гарантий доставки, поэтому его пакеты могут быть потеряны, продублированы или прийти не в том порядке, в котором они были отправлены.

Зарезервированные и доступные порты UDP

В то время, как задачей сетевого уровня является передача данных между произвольными узлами сети, задача транспортного уровня заключается в передаче данных между любыми прикладными процессами, выполняющимися на любых узлах сети. Действительно, после того, как пакет средствами протокола IP доставлен в компьютер-получатель, данные необходимо направить конкретному процессу-получателю. Каждый компьютер может выполнять несколько процессов, более того, прикладной процесс тоже может иметь несколько точек входа, выступающих в качестве адреса назначения для пакетов данных.

Пакеты, поступающие на транспортный уровень, организуются операционной системой в виде множества очередей к точкам входа различных прикладных процессов. В терминологии TCP/IP такие системные очереди называются портами. Таким образом, адресом назначения, который используется на транспортном уровне, является идентификатор (номер) порта прикладного сервиса. Номер порта, задаваемый транспортным уровнем, в совокупности с номером сети и номером компьютера, задаваемыми сетевым уровнем, однозначно определяют прикладной процесс в сети.

Назначение номеров портов прикладным процессам осуществляется либо централизованно, если эти процессы представляют собой популярные общедоступные сервисы, типа сервиса удаленного доступа к файлам TFTP (Trivial FTP) или сервиса удаленного управления telnet, либо локально для тех сервисов, которые еще не стали столь распространенными, чтобы за ними закреплять стандартные (зарезервированные) номера.

Централизованное присвоение сервисам номеров портов выполняется организацией Internet Assigned Numbers Authority. Эти номера затем закрепляются и опубликовываются в стандартах Internet. Например,

упомянутому выше сервису удаленного доступа к файлам TFTP присвоен стандартный номер порта 69.

Локальное присвоение номера порта заключается в том, что разработчик некоторого приложения просто связывает с ним любой доступный, произвольно выбранный числовой идентификатор, обращая внимание на то, чтобы он не входил в число зарезервированных номеров портов. В дальнейшем все удаленные запросы к данному приложению от других приложений должны адресоваться с указанием назначенного ему номера порта.

Мультиплексирование и демultipлексирование прикладных протоколов с помощью протокола UDP

Протокол UDP ведет для каждого порта две очереди: очередь пакетов, поступающих в данный порт из сети, и очередь пакетов, отправляемых данным портом в сеть.

Процедура обслуживания протоколом UDP запросов, поступающих от нескольких различных прикладных сервисов, называется мультиплексированием.

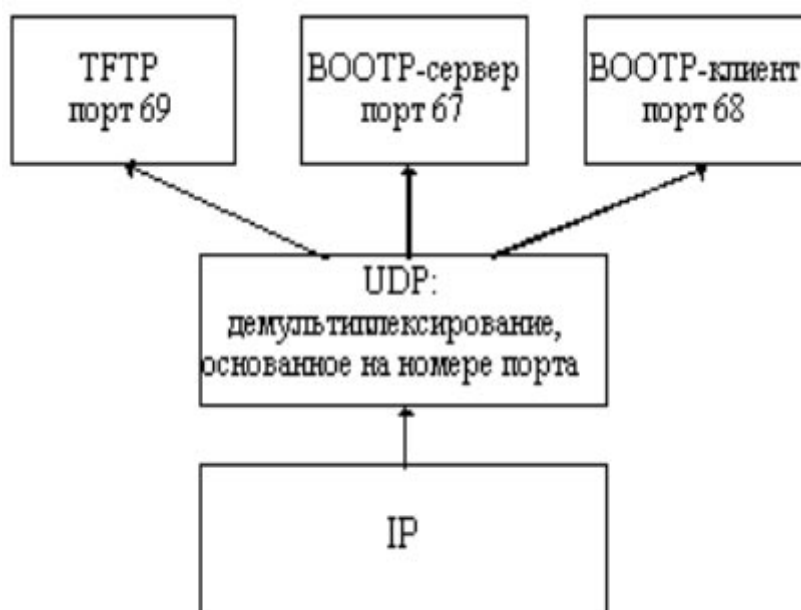


Рис. 5.1. Демultipлексирование пакетов протоколом UDP

Распределение протоколом UDP поступающих от сетевого уровня пакетов между набором высокоуровневых сервисов, идентифицированных номерами портов, называется демultipлексированием (рисунок 5.1).

Хотя к услугам протокола UDP может обратиться любое приложение, многие из них предпочитают иметь дело с другим, более сложным протоколом транспортного уровня TCP. Дело в том, что протокол UDP выступает простым посредником между сетевым уровнем и прикладными сервисами, и, в отличие от TCP, не берет на себя никаких

функций по обеспечению надежности передачи. UDP является дейтаграммным протоколом, то есть он не устанавливает логического соединения, не нумерует и не упорядочивает пакеты данных.

С другой стороны, функциональная простота протокола UDP обуславливает простоту его алгоритма, компактность и высокое быстродействие.

Поэтому те приложения, в которых реализован собственный, достаточно надежный, механизм обмена сообщениями, основанный на установлении соединения, предпочитают для непосредственной передачи данных по сети использовать менее надежные, но более быстрые средства транспортировки, в качестве которых по отношению к протоколу TCP и выступает протокол UDP.

Протокол UDP может быть использован и в том случае, когда хорошее качество каналов связи обеспечивает достаточный уровень надежности и без применения дополнительных приемов типа установления логического соединения и квитирования передаваемых пакетов.

Формат сообщений UDP

Единица данных протокола UDP называется UDP-пакетом или пользовательской дейтаграммой (user datagram). UDP-пакет состоит из заголовка и поля данных, в котором размещается пакет прикладного уровня. Заголовок имеет простой формат и состоит из четырех двухбайтовых полей:

- UDP source port - номер порта процесса-отправителя,
- UDP destination port - номер порта процесса-получателя,
- UDP message length - длина UDP-пакета в байтах,
- UDP checksum - контрольная сумма UDP-пакета

Не все поля UDP-пакета обязательно должны быть заполнены. Если посылаемая дейтаграмма не предполагает ответа, то на месте адреса отправителя могут помещаться нули. Можно отказаться и от подсчета контрольной суммы, однако следует учесть, что протокол IP подсчитывает контрольную сумму только для заголовка IP-пакета, игнорируя поле данных.

Тема 6. Протокол надежной доставки сообщений TCP

В стеке протоколов TCP/IP протокол TCP (Transmission Control Protocol) работает так же, как и протокол UDP, на транспортном уровне. Он обеспечивает надежную транспортировку данных между прикладными процессами путем установления логического соединения.

Сегменты TCP

Единицей данных протокола ТСР является сегмент. Информация, поступающая к протоколу ТСР в рамках логического соединения от протоколов более высокого уровня, рассматривается протоколом ТСР как неструктурированный поток байт.

Поступающие данные буферизуются средствами ТСР. Для передачи на сетевой уровень из буфера "вырезается" некоторая непрерывная часть данных, называемая сегментом.

В протоколе ТСР предусмотрен случай, когда приложение обращается с запросом о срочной передаче данных (бит PSH в запросе установлен в 1). В этом случае протокол ТСР, не ожидая заполнения буфера до уровня размера сегмента, немедленно передает указанные данные в сеть. О таких данных говорят, что они передаются вне потока - out of band.

Не все сегменты, посланные через соединение, будут одного и того же размера, однако оба участника соединения должны договориться о максимальном размере сегмента, который они будут использовать.

Этот размер выбирается таким образом, чтобы при упаковке сегмента в IP-пакет он помещался туда целиком, то есть максимальный размер сегмента не должен превосходить максимального размера поля данных IP-пакета. В противном случае пришлось бы выполнять фрагментацию, то есть делить сегмент на несколько частей, для того, чтобы он влез в IP-пакет.

Аналогичные проблемы решаются и на сетевом уровне. Для того, чтобы избежать фрагментации, должен быть выбран соответствующий максимальный размер IP-пакета.

Однако при этом должны быть приняты во внимание максимальные размеры поля данных кадров (MTU) всех протоколов канального уровня, используемых в сети. Максимальный размер сегмента не должен превышать минимальное значение на множестве всех MTU составной сети.

Порты и установление ТСР-соединений

В протоколе ТСР также, как и в UDP, для связи с прикладными процессами используются порты. Номера портам присваиваются аналогичным образом: имеются стандартные, зарезервированные номера (например, номер 21 закреплен за сервисом FTP, 23 - за telnet), а менее известные приложения пользуются произвольно выбранными локальными номерами.

Однако в протоколе ТСР порты используются несколько иным способом. Для организации надежной передачи данных предусматривается установление логического соединения между двумя прикладными процессами. В рамках соединения осуществляется обязательное

подтверждение правильности приема для всех переданных сообщений, и при необходимости выполняется повторная передача. Соединение в ТСР позволяет вести передачу данных одновременно в обе стороны, то есть полнодуплексную передачу.

Соединение в протоколе ТСР идентифицируется парой полных адресов обоих взаимодействующих процессов (оконечных точек). Адрес каждой из оконечных точек включает IP-адрес (номер сети и номер компьютера) и номер порта. Одна оконечная точка может участвовать в нескольких соединениях.

Установление соединения выполняется в следующей последовательности:

При установлении соединения одна из сторон является инициатором. Она посылает запрос к протоколу ТСР на открытие порта для передачи (active open).

После открытия порта протокол ТСР на стороне процесса-инициатора посылает запрос процессу, с которым требуется установить соединение.

Протокол ТСР на приемной стороне открывает порт для приема данных (passive open) и возвращает квитанцию, подтверждающую прием запроса.

Для того чтобы передача могла вестись в обе стороны, протокол на приемной стороне также открывает порт для передачи (active port) и также передает запрос к противоположной стороне.

Сторона-инициатор открывает порт для приема и возвращает квитанцию. Соединение считается установленным. Далее происходит обмен данными в рамках данного соединения.

Концепция квитирования

В рамках соединения правильность передачи каждого сегмента должна подтверждаться квитанцией получателя. Квитирование - это один из традиционных методов обеспечения надежной связи. Идея квитирования состоит в следующем.

Для того, чтобы можно было организовать повторную передачу искаженных данных отправитель нумерует отправляемые единицы передаваемых данных (далее для простоты называемые кадрами).

Для каждого кадра отправитель ожидает от приемника так называемую положительную квитанцию - служебное сообщение, извещающее о том, что исходный кадр был получен и данные в нем оказались корректными. Время этого ожидания ограничено - при отправке каждого кадра передатчик запускает таймер, и если по его истечению положительная квитанция не получена, то кадр считается утерянным.

В некоторых протоколах приемник, в случае получения кадра с искаженными данными должен отправить отрицательную квитанцию - явное указание того, что данный кадр нужно передать повторно.

Существуют два подхода к организации процесса обмена положительными и отрицательными квитанциями: с простоями и с организацией "окна".

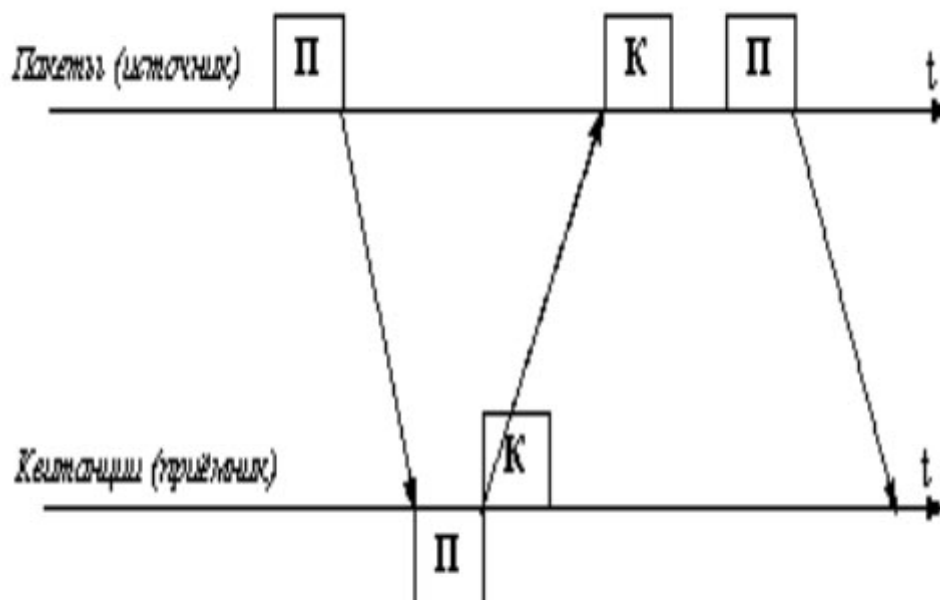


Рис. 6.1. Метод подтверждения корректности передачи кадров с простоем источника

Метод с простоями требует, чтобы источник, пославший кадр, ожидал получения квитанции (положительной или отрицательной) от приемника и только после этого посылал следующий кадр (или повторял искаженный).

Из рис. 6.1 видно, что в этом случае производительность обмена данными существенно снижается - хотя передатчик и мог бы послать следующий кадр сразу же после отправки предыдущего, он обязан ждать прихода квитанции.

Снижение производительности для этого метода коррекции особенно заметно на низкоскоростных каналах связи, то есть в территориальных сетях.

Во втором методе для повышения коэффициента использования линии источнику разрешается передать некоторое количество кадров в непрерывном режиме, то есть в максимально возможном для источника темпе, без получения на эти кадры ответных квитанций. Количество кадров, которые разрешается передавать таким образом, называется размером окна. Рисунок 6.2 иллюстрирует данный метод для размера окна в W кадров. Обычно кадры при обмене нумеруются циклически, от 1 до W . При отправке кадра с номером 1 источнику разрешается передать еще $W-1$ кадров до получения квитанции на кадр 1. Если же за это время квитанция

на кадр 1 так и не пришла, то процесс передачи приостанавливается, и по истечению некоторого тайм-аута кадр 1 считается утерянным (или квитанция на него утеряна) и он передается снова.

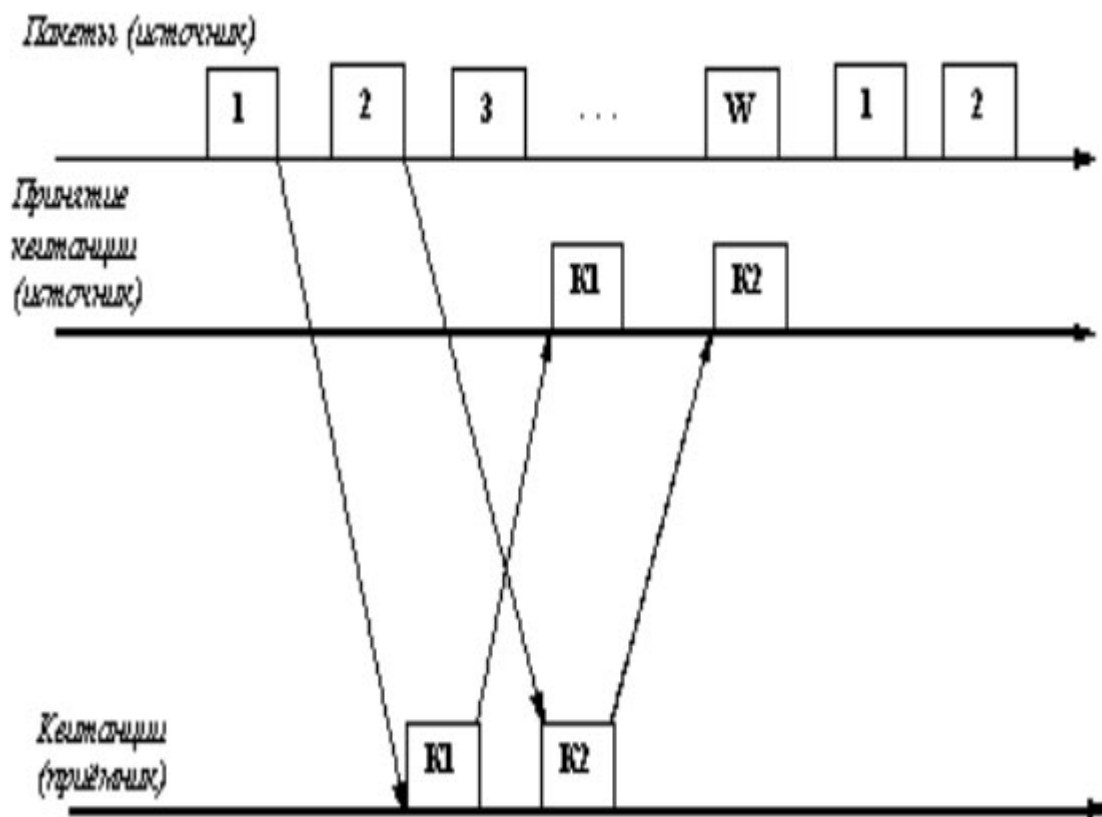


Рис. 6.2. Метод "окна" - непрерывная отправка пакетов

Если же поток квитанций поступает более-менее регулярно, в пределах допуска в W кадров, то скорость обмена достигает максимально возможной величины для данного канала и принятого протокола.

Этот алгоритм называют алгоритмом скользящего окна. Действительно, при каждом получении квитанции окно перемещается (скользит), захватывая новые данные, которые разрешается передавать без подтверждения.

Реализация скользящего окна в протоколе TCP

В протоколе TCP реализована разновидность алгоритма квитирования с использованием окна. Особенность этого алгоритма состоит в том, что, хотя единицей передаваемых данных является сегмент, окно определено на множестве нумерованных байт неструктурированного потока данных, поступающих с верхнего уровня и буферизуемых протоколом TCP.

Квитанция посылается только в случае правильного приема данных, отрицательные квитанции не посылаются. Таким образом, отсутствие квитанции означает либо прием искаженного сегмента, либо потерю сегмента, либо потерю квитанции.

В качестве квитанции получатель сегмента отсылает ответное сообщение (сегмент), в которое помещает число, на единицу превышающее максимальный номер байта в полученном сегменте.

Если размер окна равен W , а последняя квитанция содержала значение N , то отправитель может посылать новые сегменты до тех пор, пока в очередной сегмент не попадет байт с номером $N+W$. Этот сегмент выходит за рамки окна, и передачу в таком случае необходимо приостановить до прихода следующей квитанции.

Выбор тайм-аута

Выбор времени ожидания (тайм-аута) очередной квитанции является важной задачей, результат решения которой влияет на производительность протокола TCP.

Тайм-аут не должен быть слишком коротким, чтобы по возможности исключить избыточные повторные передачи, которые снижают полезную пропускную способность системы.

Но он не должен быть и слишком большим, чтобы избежать длительных простоев, связанных с ожиданием несуществующей или "заблудившейся" квитанции.

При выборе величины тайм-аута должны учитываться скорость и надежность физических линий связи, их протяженность и многие другие подобные факторы.

В протоколе TCP тайм-аут определяется с помощью достаточно сложного адаптивного алгоритма, идея которого состоит в следующем.

При каждой передаче засекается время от момента отправки сегмента до прихода квитанции об его приеме (время оборота).

Получаемые значения времен оборота усредняются с весовыми коэффициентами, возрастающими от предыдущего замера к последующему. Это делается с тем, чтобы усилить влияние последних замеров. В качестве тайм-аута выбирается среднее время оборота, умноженное на некоторый коэффициент.

Практика показывает, что значение этого коэффициента должно превышать 2. В сетях с большим разбросом времени оборота при выборе тайм-аута учитывается и дисперсия этой величины.

Реакция на перегрузку сети

Варьируя величину окна, можно повлиять на загрузку сети. Чем больше окно, тем большую порцию неподтвержденных данных можно послать в сеть. Если сеть не справляется с нагрузкой, то возникают очереди в промежуточных узлах-маршрутизаторах и в конечных узлах-компьютерах.

При переполнении приемного буфера конечного узла "перегруженный" протокол TCP, отправляя квитанцию, помещает в нее новый, уменьшенный размер окна.

Если он совсем отказывается от приема, то в квитанции указывается окно нулевого размера. Однако даже после этого приложение может послать сообщение на отказавшийся от приема порт.

Для этого, сообщение должно сопровождаться пометкой "срочно" (бит URG в запросе установлен в 1). В такой ситуации порт обязан принять сегмент, даже если для этого придется вытеснить из буфера уже находящиеся там данные.

После приема квитанции с нулевым значением окна протокол-отправитель время от времени делает контрольные попытки продолжить обмен данными.

Если протокол-приемник уже готов принимать информацию, то в ответ на контрольный запрос он посылает квитанцию с указанием ненулевого размера окна.

Другим проявлением перегрузки сети является переполнение буферов в маршрутизаторах. В таких случаях они могут централизованно изменить размер окна, посылая управляющие сообщения некоторым конечным узлам, что позволяет им дифференцированно управлять интенсивностью потока данных в разных частях сети.

Формат сообщений TCP

Сообщения протокола TCP называются сегментами и состоят из заголовка и блока данных. Заголовок сегмента имеет следующие поля:

Порт источника (SOURCE PORT) занимает 2 байта, идентифицирует процесс-отправитель;

Порт назначения (DESTINATION PORT) занимает 2 байта, идентифицирует процесс-получатель;

Последовательный номер (SEQUENCE NUMBER) занимает 4 байта, указывает номер байта, который определяет смещение сегмента относительно потока отправляемых данных;

Подтвержденный номер (ACKNOWLEDGEMENT NUMBER) занимает 4 байта, содержит максимальный номер байта в полученном сегменте, увеличенный на единицу; именно это значение используется в качестве квитанции;

Длина заголовка (HLEN) занимает 4 бита, указывает длину заголовка сегмента TCP, измеренную в 32-битовых словах.

Длина заголовка не фиксирована и может изменяться в зависимости от значений, устанавливаемых в поле Опции;

Резерв (RESERVED) занимает 6 битов, поле зарезервировано для последующего использования;

Кодовые биты (CODE BITS) занимают 6 битов, содержат служебную информацию о типе данного сегмента, задаваемую установкой в единицу соответствующих бит этого поля:

URG - срочное сообщение;

ACK - квитанция на принятый сегмент;

PSH - запрос на отправку сообщения без ожидания заполнения буфера;

RST - запрос на восстановление соединения;

SYN - сообщение используемое для синхронизации счетчиков переданных данных при установлении соединения;

FIN - признак достижения передающей стороной последнего байта в потоке передаваемых данных.

Окно (WINDOW) занимает 2 байта, содержит объявляемое значение размера окна в байтах;

Контрольная сумма (CHECKSUM) занимает 2 байта, рассчитывается по сегменту;

Указатель срочности (URGENT POINTER) занимает 2 байта, используется совместно с кодовым битом URG, указывает на конец данных, которые необходимо срочно принять, несмотря на переполнение буфера;

Опции (OPTIONS) - это поле имеет переменную длину и может вообще отсутствовать, максимальная величина поля 3 байта; используется для решения вспомогательных задач, например, при выборе максимального размера сегмента;

Заполнитель (PADDING) может иметь переменную длину, представляет собой фиктивное поле, используемое для доведения размера заголовка до целого числа 32-битовых слов.

Тема 7. Протоколы обмена маршрутной информацией стека TCP/IP

Все протоколы обмена маршрутной информацией стека TCP/IP относятся к классу адаптивных протоколов, которые в свою очередь делятся на две группы, каждая из которых связана с одним из следующих типов алгоритмов:

дистанционно-векторный алгоритм (Distance Vector Algorithms, DVA),

алгоритм состояния связей (Link State Algorithms, LSA).

В алгоритмах дистанционно-векторного типа каждый маршрутизатор периодически и широковещательно рассылает по сети вектор расстояний от себя до всех известных ему сетей.

Под расстоянием обычно понимается число промежуточных маршрутизаторов через которые пакет должен пройти прежде, чем попадет в соответствующую сеть. Может использоваться и другая метрика,

учитывающая не только число перевалочных пунктов, но и время прохождения пакетов по связи между соседними маршрутизаторами. Получив вектор от соседнего маршрутизатора, каждый маршрутизатор добавляет к нему информацию об известных ему других сетях, о которых он узнал непосредственно (если они подключены к его портам) или из аналогичных объявлений других маршрутизаторов, а затем снова рассылает новое значение вектора по сети. В конце-концов, каждый маршрутизатор узнает информацию об имеющихся в интересах сетей и о расстоянии до них через соседние маршрутизаторы.

Дистанционно-векторные алгоритмы хорошо работают только в небольших сетях. В больших сетях они засоряют линии связи интенсивным ширококестельным трафиком, к тому же изменения конфигурации могут отрабатываться по этому алгоритму не всегда корректно, так как маршрутизаторы не имеют точного представления о топологии связей в сети, а располагают только обобщенной информацией - вектором дистанций, к тому же полученной через посредников. Работа маршрутизатора в соответствии с дистанционно-векторным протоколом напоминает работу моста, так как точной топологической картины сети такой маршрутизатор не имеет.

Наиболее распространенным протоколом, основанным на дистанционно-векторном алгоритме, является протокол RIP.

Алгоритмы состояния связей обеспечивают каждый маршрутизатор информацией, достаточной для построения точного графа связей сети. Все маршрутизаторы работают на основании одинаковых графов, что делает процесс маршрутизации более устойчивым к изменениям конфигурации. Ширококестельная рассылка используется здесь только при изменениях состояния связей, что происходит в надежных сетях не так часто.

Для того, чтобы понять, в каком состоянии находятся линии связи, подключенные к его портам, маршрутизатор периодически обменивается короткими пакетами со своими ближайшими соседями. Этот трафик также ширококестельный, но он циркулирует только между соседями и поэтому не так засоряет сеть.

Протоколом, основанным на алгоритме состояния связей, в стеке TCP/IP является протокол OSPF.

Дистанционно-векторный протокол RIP

Протокол RIP (Routing Information Protocol) представляет собой один из старейших протоколов обмена маршрутной информацией, однако он до сих пор чрезвычайно распространен в вычислительных сетях. Помимо версии RIP для сетей TCP/IP, существует также версия RIP для сетей IPX/SPX компании Novell.

В этом протоколе все сети имеют номера (способ образования номера зависит от используемого в сети протокола сетевого уровня), а все

маршрутизаторы - идентификаторы. Протокол RIP широко использует понятие "вектор расстояний". Вектор расстояний представляет собой набор пар чисел, являющихся номерами сетей и расстояниями до них в хопах.

Вектора расстояний итерационно распространяются маршрутизаторами по сети, и через несколько шагов каждый маршрутизатор имеет данные о достижимых для него сетях и о расстояниях до них. Если связь с какой-либо сетью обрывается, то маршрутизатор отмечает этот факт тем, что присваивает элементу вектора, соответствующему расстоянию до этой сети, максимально возможное значение, которое имеет специальный смысл - "связи нет". Таким значением в протоколе RIP является число 16.

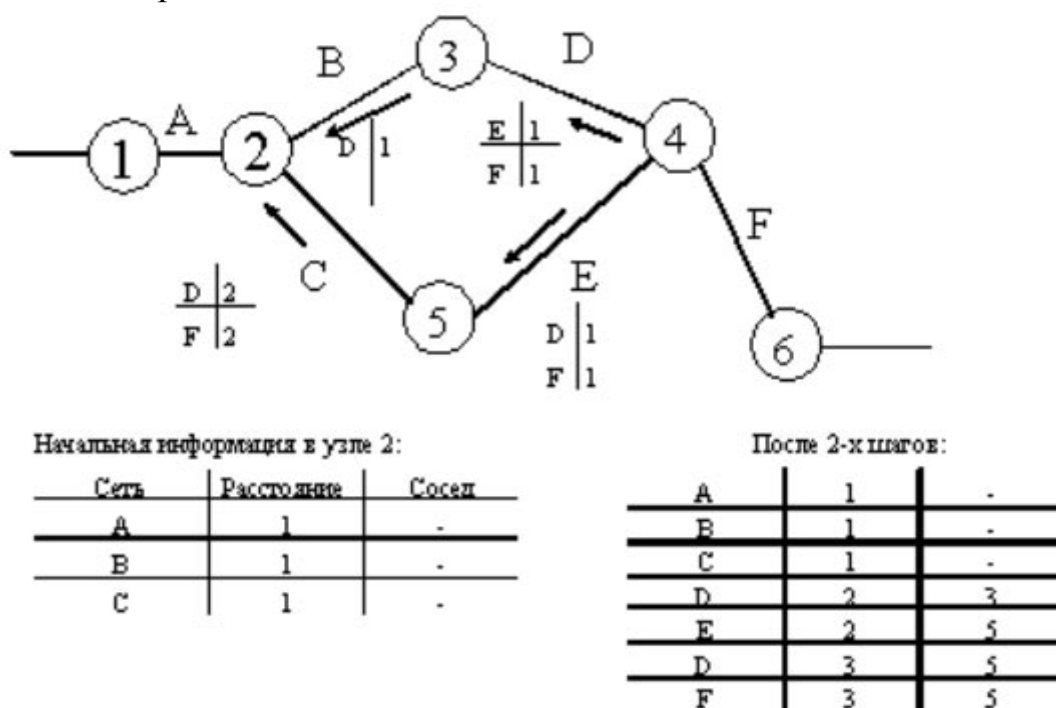


Рис. 7.1. Обмен маршрутной информацией по протоколу RIP

На рис. 7.1 приведен пример сети, состоящей из шести маршрутизаторов, имеющих идентификаторы от 1 до 6, и из шести сетей от A до F, образованных прямыми связями типа "точка-точка".

На рисунке приведена начальная информация, содержащаяся в топологической базе маршрутизатора 2, а также информация в этой же базе после двух итераций обмена маршрутными пакетами протокола RIP. После определенного числа итераций маршрутизатор 2 будет знать о расстояниях до всех сетей интерсети, причем у него может быть несколько альтернативных вариантов отправки пакета к сети назначения. Пусть в нашем примере сетью назначения является сеть D.

При необходимости отправить пакет в сеть D маршрутизатор просматривает свою базу данных маршрутов и выбирает порт, имеющий

наименьшее расстояния до сети назначения (в данном случае порт, связывающий его с маршрутизатором 3).

Для адаптации к изменению состояния связей и оборудования с каждой записью таблицы маршрутизации связан таймер. Если за время тайм-аута не придет новое сообщение, подтверждающее этот маршрут, то он удаляется из маршрутной таблицы.

При использовании протокола RIP работает эвристический алгоритм динамического программирования Беллмана-Форда, и решение, найденное с его помощью является не оптимальным, а близким к оптимальному. Преимуществом протокола RIP является его вычислительная простота, а недостатками - увеличение трафика при периодической рассылке широковещательных пакетов и неоптимальность найденного маршрута.

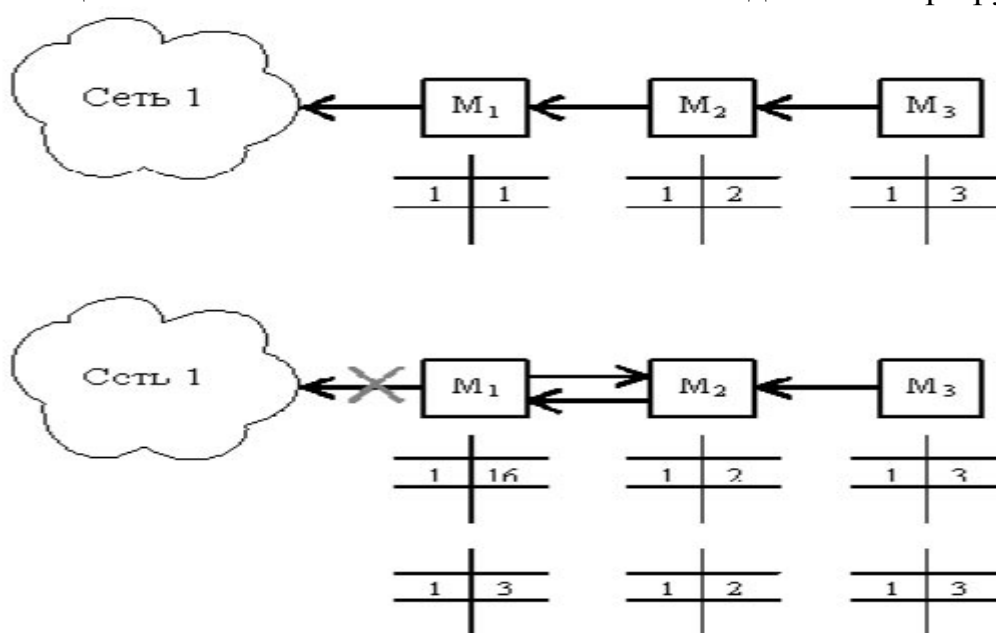


Рис. 7.2. Пример неустойчивой работы сети при использовании протокола RIP

На рис. 7.2 показан случай неустойчивой работы сети по протоколу RIP при изменении конфигурации - отказе линии связи маршрутизатора M₁ с сетью 1. При работоспособном состоянии этой связи в таблице маршрутов каждого маршрутизатора есть запись о сети с номером 1 и соответствующим расстоянием до нее. При обрыве связи с сетью 1 маршрутизатор M₁ отмечает, что расстояние до этой сети приняло значение 16. Однако получив через некоторое время от маршрутизатора M₂ маршрутное сообщение о том, что от него до сети 1 расстояние составляет 2 хопа, маршрутизатор M₁ наращивает это расстояние на 1 и отмечает, что сеть 1 достижима через маршрутизатор 2. В результате пакет, предназначенный для сети 1, будет циркулировать между маршрутизаторами M₁ и M₂ до тех пор, пока не истечет время хранения записи о сети 1 в маршрутизаторе 2, и он не передаст эту информацию маршрутизатору M₁.

Для исключения подобных ситуаций маршрутная информация об известной маршрутизатору сети не передается тому маршрутизатору, от которого она пришла.

Существуют и другие, более сложные случаи нестабильного поведения сетей, использующих протокол RIP, при изменениях в состоянии связей или маршрутизаторов сети.

Комбинирование различных протоколов обмена. Протоколы EGP и BGP сети Internet

Большинство протоколов маршрутизации, применяемых в современных сетях с коммутацией пакетов, ведут свое происхождение от сети Internet и ее предшественницы - сети ARPANET. Для того, чтобы понять их назначение и особенности, полезно сначала познакомиться со структурой сети Internet, которая наложила отпечаток на терминологию и типы протоколов. Internet изначально строилась как сеть, объединяющая большое количество существующих систем. С самого начала в ее структуре выделяли магистральную сеть (core backbone network), а сети, присоединенные к магистрали, рассматривались как автономные системы (autonomous systems). Магистральная сеть и каждая из автономных систем имели свое собственное административное управление и собственные протоколы маршрутизации. Общая схема архитектуры сети Internet показана на рис. 7.3. Далее маршрутизаторы будут называться шлюзами для следования традиционной терминологии Internet.

Шлюзы, которые используются для образования подсетей внутри автономной системы, называются внутренними шлюзами (interior gateways), а шлюзы, с помощью которых автономные системы присоединяются к магистрали сети, называются внешними шлюзами (exterior gateways). Непосредственно друг с другом автономные системы не соединяются.

Соответственно, протоколы маршрутизации, используемые внутри автономных систем, называются протоколами внутренних шлюзов (interior gateway protocol, IGP), а протоколы, определяющие обмен маршрутной информацией между внешними шлюзами и шлюзами магистральной сети - протоколами внешних шлюзов (exterior gateway protocol, EGP). Внутри магистральной сети также может использоваться любой собственный внутренний протокол IGP.

Смысл разделения всей сети Internet на автономные системы в ее многоуровневом представлении, что необходимо для любой крупной системы, способной к расширению в больших масштабах.

Внутренние шлюзы могут использовать для внутренней маршрутизации достаточно подробные графы связей между собой, чтобы выбрать наиболее рациональный маршрут. Однако, если информация такой степени детализации будет храниться во всех маршрутизаторах сети, то топологические базы данных так разрастутся, что потребуют наличия

памяти гигантских размеров, а время принятия решений о маршрутизации непременно возрастет.

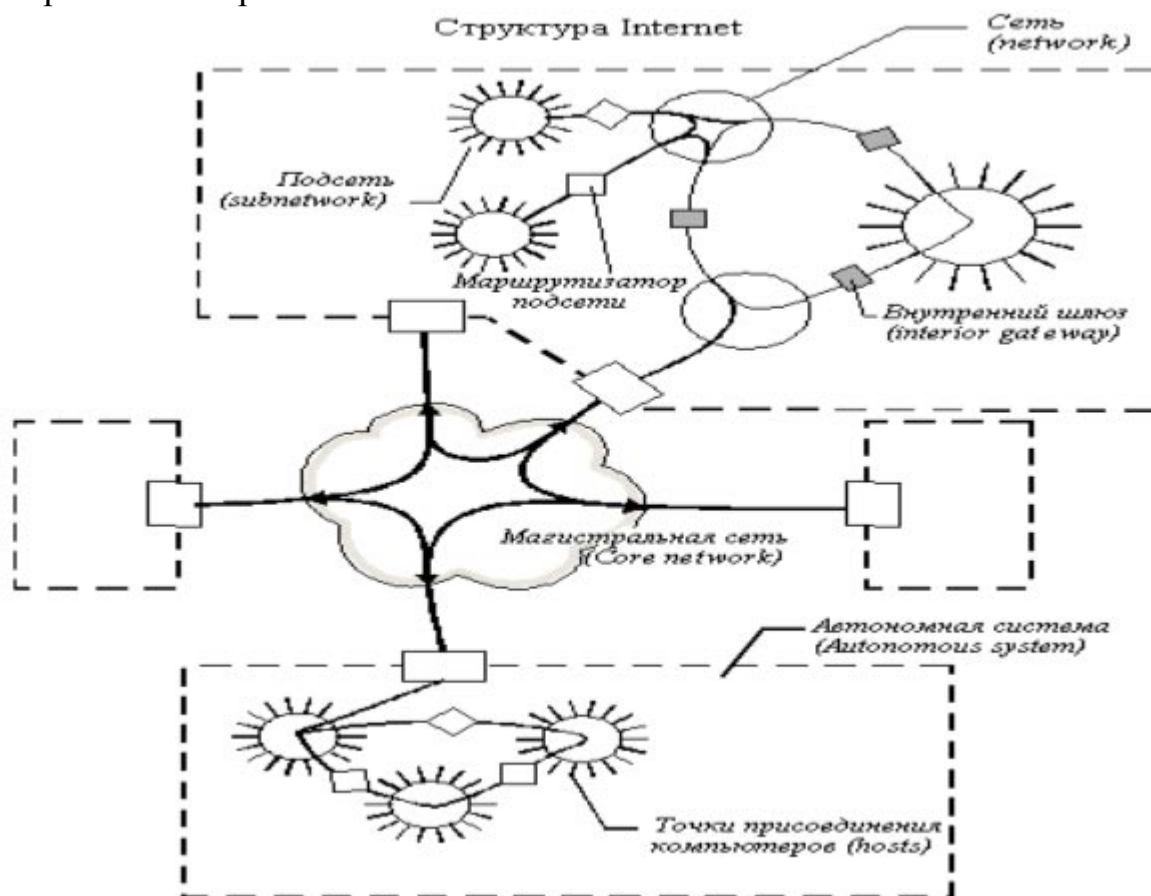


Рис. 7.3. Архитектура сети Internet

Поэтому детальная топологическая информация остается внутри автономной системы, а автономную систему как единое целое для остальной части Internet представляют внешние шлюзы, которые сообщают о внутреннем составе автономной системы минимально необходимые сведения - количество IP-сетей, их адреса и внутреннее расстояние до этих сетей от данного внешнего шлюза.

При инициализации внешний шлюз узнает уникальный идентификатор обслуживаемой им автономной системы, а также таблицу достижимости (reachability table), которая позволяет ему взаимодействовать с другими внешними шлюзами через магистральную сеть.

Затем внешний шлюз начинает взаимодействовать по протоколу EGP с другими внешними шлюзами и обмениваться с ними маршрутной информацией, состав которой описан выше. В результате, при отправке пакета из одной автономной системы в другую, внешний шлюз данной системы на основании маршрутной информации, полученной от всех внешних шлюзов, с которыми он общается по протоколу EGP, выбирает наиболее подходящий внешний шлюз и отправляет ему пакет.

В протоколе EGP определены три основные функции:

установление соседских отношений,
подтверждение достижимости соседа,
обновление маршрутной информации.

Каждая функция работает на основе обмена сообщениями запрос-ответ.

Так как каждая автономная система работает под контролем своего административного штата, то перед началом обмена маршрутной информацией внешние шлюзы должны согласиться на такой обмен.

Сначала один из шлюзов посылает запрос на установление соседских отношений (acquisition request) другому шлюзу.

Если тот согласен на это, то он отвечает сообщением подтверждение установления соседских отношений (acquisition confirm), а если нет - то сообщением отказ от установления соседских отношений (acquisition refuse), которое содержит также причину отказа.

После установления соседских отношений шлюзы начинают периодически проверять состояние достижимости друг друга. Это делается либо с помощью специальных сообщений (привет (hello) и Я-услышал-тебя (I-heard-you)), либо встраиванием подтверждающей информации непосредственно в заголовок обычного маршрутного сообщения.

Обмен маршрутной информацией начинается с посылки одним из шлюзов другому сообщения запрос данных (poll request) о номерах сетей, обслуживаемых другим шлюзом и расстояниях до них от него.

Ответом на это сообщение служит сообщение обновленная маршрутная информация (routing update). Если же запрос оказался некорректным, то в ответ на него отсылается сообщение об ошибке.

Все сообщения протокола EGP передаются в поле данных IP-пакетов. Сообщения EGP имеют заголовок фиксированного формата.

Поля Тип и Код совместно определяют тип сообщения, а поле Статус - информацию, зависящую от типа сообщения.

Поле Номер автономной системы - это номер, назначенный той автономной системе, к которой присоединен данный внешний шлюз. Поле Номер последовательности служит для синхронизации процесса запросов и ответов. Поле IP-адрес исходной сети в сообщениях запроса и обновления маршрутной информации обозначает сеть, соединяющую два внешних шлюза. Сообщение об обновленной маршрутной информации содержит список адресов сетей, которые достижимы в данной автономной системе. Этот список упорядочен по внутренним шлюзам, которые подключены к исходной сети и через которые достижимы данные сети, а для каждого шлюза он упорядочен по расстоянию до каждой достижимой сети от исходной сети, а не от данного внутреннего шлюза.

Для примера, приведенного на рисунке 7.4, внешний шлюз R2 в своем сообщении указывает, что сеть 4 достижима с помощью шлюза R3 и расстояние ее равно 2, а сеть 2 достижима через шлюз R2 и ее расстояние

равно 1 (а не 0, как если бы шлюз измерял ее расстояние от себя, как в протоколе RIP). Протокол EGP имеет достаточно много ограничений, связанных с тем, что он рассматривает магистральную сеть как одну неделимую магистраль.

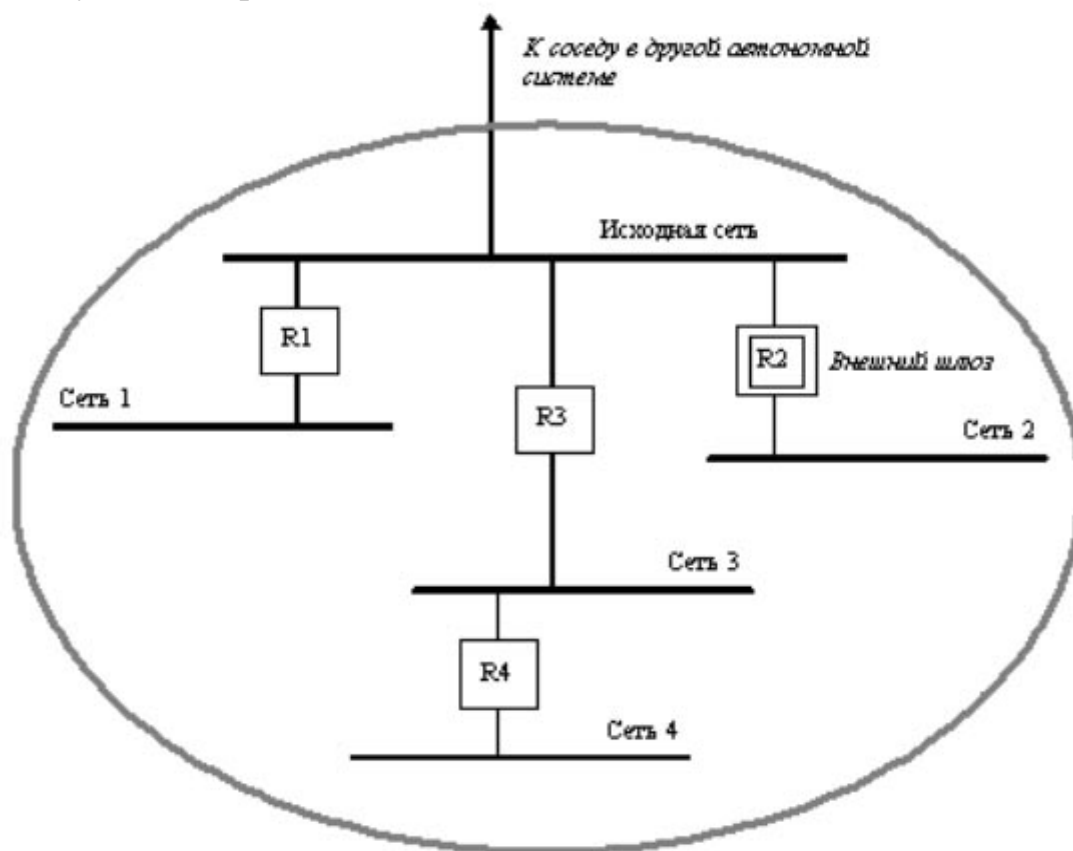


Рис. 7.4. Пример автономной системы

Развитием протокола EGP является протокол BGP (Border Gateway Protocol), имеющий много общего с EGP и используемый наряду с ним в магистральной сети Internet.

Протокол состояния связей OSPF

Протокол OSPF (Open Shortest Path First) является достаточно современной реализацией алгоритма состояния связей (он принят в 1991 году) и обладает многими особенностями, ориентированными на применение в больших гетерогенных сетях.

Протокол OSPF вычисляет маршруты в IP-сетях, сохраняя при этом другие протоколы обмена маршрутной информацией.

Непосредственно связанные (то есть достижимые без использования промежуточных маршрутизаторов) маршрутизаторы называются "соседями". Каждый маршрутизатор хранит информацию о том, в каком состоянии по его мнению находится сосед. Маршрутизатор полагается на соседние маршрутизаторы и передает им пакеты данных только в том случае, если он уверен, что они полностью работоспособны. Для

выяснения состояния связей маршрутизаторы-соседи достаточно часто обмениваются короткими сообщениями HELLO.

Для распространения по сети данных о состоянии связей маршрутизаторы обмениваются сообщениями другого типа. Эти сообщения называются router links advertisement - объявление о связях маршрутизатора (точнее, о состоянии связей). OSPF-маршрутизаторы обмениваются не только своими, но и чужими объявлениями о связях, получая в конце-концов информацию о состоянии всех связей сети. Эта информация и образует граф связей сети, который, естественно, один и тот же для всех маршрутизаторов сети.

Кроме информации о соседях, маршрутизатор в своем объявлении перечисляет IP-подсети, с которыми он связан непосредственно, поэтому после получения информации о графе связей сети, вычисление маршрута до каждой сети производится непосредственно по этому графу по алгоритму Дэйкстры. Более точно, маршрутизатор вычисляет путь не до конкретной сети, а до маршрутизатора, к которому эта сеть подключена. Каждый маршрутизатор имеет уникальный идентификатор, который передается в объявлении о состояниях связей. Такой подход дает возможность не тратить IP-адреса на связи типа "точка-точка" между маршрутизаторами, к которым не подключены рабочие станции.

Маршрутизатор вычисляет оптимальный маршрут до каждой адресуемой сети, но запоминает только первый промежуточный маршрутизатор из каждого маршрута. Таким образом, результатом вычислений оптимальных маршрутов является список строк, в которых указывается номер сети и идентификатор маршрутизатора, которому нужно переслать пакет для этой сети. Указанный список маршрутов и является маршрутной таблицей, но вычислен он на основании полной информации о графе связей сети, а не частичной информации, как в протоколе RIP.

Описанный подход приводит к результату, который не может быть достигнут при использовании протокола RIP или других дистанционно-векторных алгоритмов. RIP предполагает, что все подсети определенной IP-сети имеют один и тот же размер, то есть, что все они могут потенциально иметь одинаковое число IP-узлов, адреса которых не перекрываются. Более того, классическая реализация RIP требует, чтобы выделенные линии "точка-точка" имели IP-адрес, что приводит к дополнительным затратам IP-адресов.

В OSPF такие требования отсутствуют: сети могут иметь различное число хостов и могут перекрываться. Под перекрытием понимается наличие нескольких маршрутов к одной и той же сети. В этом случае адрес сети в пришедшем пакете может совпасть с адресом сети, присвоенным нескольким портам.

Если адрес принадлежит нескольким подсетям в базе данных маршрутов, то продвигающий пакет маршрутизатор использует наиболее специфический маршрут, то есть адрес подсети, имеющей более длинную маску.

Например, если рабочая группа ответвляется от главной сети, то она имеет адрес главной сети наряду с более специфическим адресом, определяемым маской подсети. При выборе маршрута к хосту в подсети этой рабочей группы маршрутизатор найдет два пути, один для главной сети и один для рабочей группы. Так как последний более специфичен, то он и будет выбран. Этот механизм является обобщением понятия "маршрут по умолчанию", используемого во многих сетях.

Использование подсетей с различным количеством хостов является вполне естественным. Например, если в здании или кампусе на каждом этаже имеются локальные сети, и на некоторых этажах компьютеров больше, чем на других, то администратор может выбрать размеры подсетей, отражающие ожидаемые требования каждого этажа, а не соответствующие размеру наибольшей подсети.

В протоколе OSPF подсети делятся на три категории:

"хост-сеть", представляющая собой подсеть из одного адреса,

"тупиковая сеть", которая представляет собой подсеть, подключенную только к одному маршрутизатору,

"транзитная сеть", которая представляет собой подсеть, подключенную к более чем одному маршрутизатору.

Транзитная сеть является для протокола OSPF особым случаем. В транзитной сети несколько маршрутизаторов являются взаимно и одновременно достижимыми. В широкоэвентельных локальных сетях, таких как Ethernet или Token Ring, маршрутизатор может послать одно сообщение, которое получают все его соседи. Это уменьшает нагрузку на маршрутизатор, когда он посылает сообщения для определения существования связи или обновленные объявления о соседях. Однако, если каждый маршрутизатор будет перечислять всех своих соседей в своих объявлениях о соседях, то объявления займут много места в памяти маршрутизатора. При определении пути по адресам транзитной подсети может обнаружиться много избыточных маршрутов к различным маршрутизаторам. На вычисление, проверку и отбраковку этих маршрутов уйдет много времени.

Когда маршрутизатор начинает работать в первый раз (то есть устанавливается), он пытается синхронизировать свою базу данных со всеми маршрутизаторами транзитной локальной сети, которые по определению имеют идентичные базы данных. Для упрощения и оптимизации этого процесса в протоколе OSPF используется понятие "выделенного" маршрутизатора, который выполняет две функции.

Во-первых, выделенный маршрутизатор и его резервный "напарник" являются единственными маршрутизаторами, с которыми новый маршрутизатор будет синхронизировать свою базу. Синхронизировав базу с выделенным маршрутизатором, новый маршрутизатор будет синхронизирован со всеми маршрутизаторами данной локальной сети.

Во-вторых, выделенный маршрутизатор делает объявление о сетевых связях, перечисляя своих соседей по подсети. Другие маршрутизаторы просто объявляют о своей связи с выделенным маршрутизатором. Это делает объявления о связях (которых много) более краткими, размером с объявление о связях отдельной сети.

Для начала работы маршрутизатора OSPF нужен минимум информации - IP-конфигурация (IP-адреса и маски подсетей), некоторая информация по умолчанию (default) и команда на включение. Для многих сетей информация по умолчанию весьма похожа. В то же время протокол OSPF предусматривает высокую степень программируемости.

Интерфейс OSPF (порт маршрутизатора, поддерживающего протокол OSPF) является обобщением подсети IP. Подобно подсети IP, интерфейс OSPF имеет IP-адрес и маску подсети. Если один порт OSPF поддерживает более, чем одну подсеть, протокол OSPF рассматривает эти подсети так, как если бы они были на разных физических интерфейсах, и вычисляет маршруты соответственно.

Интерфейсы, к которым подключены локальные сети, называются широковещательными (broadcast) интерфейсами, так как они могут использовать широковещательные возможности локальных сетей для обмена сигнальной информацией между маршрутизаторами.

Интерфейсы, к которым подключены глобальные сети, не поддерживающие широковещание, но обеспечивающие доступ ко многим узлам через одну точку входа, например сети X.25 или frame relay, называются нешироковещательными интерфейсами с множественным доступом или NBMA (non-broadcast multi-access).

Они рассматриваются аналогично широковещательным интерфейсам за исключением того, что широковещательная рассылка эмулируется путем послышки сообщения каждому соседу.

Так как обнаружение соседей не является автоматическим, как в широковещательных сетях, NBMA-соседи должны задаваться при конфигурировании вручную.

Как на широковещательных, так и на NBMA-интерфейсах могут быть заданы приоритеты маршрутизаторов для того, чтобы они могли выбрать выделенный маршрутизатор.

Интерфейсы "точка-точка", подобные PPP, несколько отличаются от традиционной IP-модели. Хотя они и могут иметь IP-адреса и подмаски, но необходимости в этом нет.

В простых сетях достаточно определить, что пункт назначения достижим и найти маршрут, который будет удовлетворительным.

В сложных сетях обычно имеется несколько возможных маршрутов. Иногда хотелось бы иметь возможности по установлению дополнительных критериев для выбора пути: например, наименьшая задержка, максимальная пропускная способность или наименьшая стоимость (в сетях с оплатой за пакет).

По этим причинам протокол OSPF позволяет сетевому администратору назначать каждому интерфейсу определенное число, называемое метрикой, чтобы оказать нужное влияние на выбор маршрута.

Число, используемое в качестве метрики пути, может быть назначено произвольным образом по желанию администратора. Но по умолчанию в качестве метрики используется время передачи бита в 10-ти наносекундных единицах (10 Мб/с Ethernet'у назначается значение 10, а линии 56 Кб/с - число 1785). Вычисляемая протоколом OSPF метрика пути представляет собой сумму метрик всех проходимых в пути связей; это очень грубая оценка задержки пути.

Если маршрутизатор обнаруживает более, чем один путь к удаленной подсети, то он использует путь с наименьшей стоимостью пути.

В протоколе OSPF используется несколько временных параметров, и среди них наиболее важными являются интервал сообщения HELLO и интервал отказа маршрутизатора (router dead interval).

HELLO - это сообщение, которым обмениваются соседние, то есть непосредственно связанные маршрутизаторы подсети, с целью установить состояние линии связи и состояние маршрутизатора-соседа.

В сообщении HELLO маршрутизатор передает свои рабочие параметры и говорит о том, кого он рассматривает в качестве своих ближайших соседей. Маршрутизаторы с разными рабочими параметрами игнорируют сообщения HELLO друг друга, поэтому неверно сконфигурированные маршрутизаторы не будут влиять на работу сети.

Каждый маршрутизатор шлет сообщение HELLO каждому своему соседу по крайней мере один раз на протяжении интервала HELLO.

Если интервал отказа маршрутизатора истекает без получения сообщения HELLO от соседа, то считается, что сосед неработоспособен, и распространяется новое объявление о сетевых связях, чтобы в сети произошел пересчет маршрутов.

Тема 8. Маршрутизация небольших сетей

Рассматриваются распространенные функции маршрутизаторов класса SOHO. Многие модели маршрутизаторов поддерживают и другие функции, которые, несмотря на различные названия, имеют одни и те же назначения. К примеру, это могут быть возможность блокирования определенных URL, запрет отклика на сканирование командой Ping,

перевод маршрутизатора в режим Stealth, при котором он становится невидимым из внешней сети, и многое другое.

Маршрутизатор осуществляет надежную защиту внутренней сети. К тому же его применение экономически более выгодно, чем использование в качестве маршрутизатора ПК (если, конечно, учитывать цены необходимого ПО).

Средств для защиты от разного рода атак извне (то есть из Интернета) на локальную сеть разработано очень много, но все они имеют один серьезный недостаток: чтобы реализовать такого рода защиту, необходимо выделить ПК, на котором будет установлено и настроено специализированное ПО. Если речь идет о достаточно крупной сети, насчитывающей не менее сотни ПК, то такое решение вполне оправданно, в случае же небольших сетей класса SOHO выделять компьютер для организации защиты сети весьма накладно.

Кроме того, необходимо помнить, что профессиональные пакеты, реализующие защиту сети, стоят довольно дорого, поэтому для сегмента SOHO, возможно, имеет смысл обратиться к альтернативным решениям. Речь идет о маршрутизаторах класса SOHO, которые называют также Интернет-серверами и Интернет-маршрутизаторами.

Конструкция маршрутизатора

Поскольку маршрутизаторы являются пограничными сетевыми устройствами, то есть устанавливаются на границе между двумя сетями или между локальной сетью и Интернетом, выполняя роль сетевого шлюза, то они должны иметь как минимум два порта (см. рисунок).

К одному из этих портов подключается локальная сеть, и этот порт называется внутренним LAN-портом. Ко второму порту подключается внешняя сеть (Интернет), и этот порт называется внешним WAN-портом. Как правило, маршрутизаторы класса SOHO имеют один WAN-порт и несколько (от одного до четырех) внутренних LAN-портов, которые объединяются в коммутатор. В большинстве случаев WAN-порт коммутатора имеет интерфейс 10/100Base-TX, и к нему может подключаться xDSL-модем с соответствующим интерфейсом либо сетевой Ethernet-кабель.

В некоторых моделях маршрутизаторов, кроме WAN-порта, есть последовательный порт для подключения аналогового модема.

Как правило, этот порт предназначен для создания резервного низкоскоростного соединения по коммутируемой линии с провайдером.

Учитывая широкое распространение беспроводных сетей, создан целый класс так называемых беспроводных маршрутизаторов. Эти устройства, кроме классического маршрутизатора с WAN- и LAN-портами, содержат интегрированную точку беспроводного доступа, поддерживающую протокол IEEE 802.11a/b/g. Беспроводной сегмент сети,

которую позволяет организовать точка доступа, относится к внутренней сети с точки зрения маршрутизатора, и в этом смысле компьютеры, подключаемые к маршрутизатору беспроводным образом, ничем не отличаются от компьютеров сети, подключенных к LAN-порту.

Любой маршрутизатор, как устройство сетевого уровня, имеет свой IP-адрес. Кроме того, IP-адрес есть и у WAN-порта. К примеру, маршрутизатор может иметь следующий IP-адрес:

- IP-адрес: 192.168.1.254;
- маска подсети: 255.255.255.0.

При этом у WAN-порта может быть такой адрес:

- IP-адрес: 10.0.0.254;
- маска подсети: 255.255.255.0.

Компьютеры, подключаемые к LAN-портам маршрутизатора, должны иметь IP-адрес той же подсети, что и сам маршрутизатор. Кроме того, в сетевых настройках этих ПК необходимо задать адрес шлюза по умолчанию, совпадающий с IP-адресом маршрутизатора. К примеру, в рассмотренном выше случае сетевые настройки ПК, подключаемого к LAN-порту, могут быть следующими:

- IP-адрес: 192.168.1.10;
- маска подсети: 255.255.255.0;
- шлюз по умолчанию: 192.168.1.254.

Устройство, подключаемое к WAN-порту со стороны внешней сети, должно иметь IP-адрес из той же подсети, что и WAN-порт маршрутизатора. В нашем случае это могут следующие сетевые настройки:

- IP-адрес: 10.0.0.10;
- маска подсети: 255.255.255.0.

В рассмотренном выше примере использовался так называемый статический способ задания IP-адреса (Static IP), который поддерживают все маршрутизаторы. Его следует применять для ознакомления с возможностями работы маршрутизатора или для его тестирования. Однако в реальных условиях чаще используется динамический (Dynamic IP) способ задания IP-адреса, когда маршрутизатор выступает в роли DHCP-клиента, автоматически получая IP-адрес, адрес шлюза по умолчанию и сервера DNS от провайдера (DHCP-сервера). Этот способ обеспечивает провайдеру достаточную гибкость при конфигурировании своей сети и поддерживается всеми провайдерами.

Брандмауэр (firewall)

Поскольку маршрутизатор выполняет функцию шлюза между локальной сетью и Интернетом, было бы логично наделить его такой функцией, как защита внутренней сети от несанкционированного доступа. Поэтому практически все современные маршрутизаторы класса SOHO

имеют встроенные аппаратные брандмауэры, называемые также сетевыми экранами, или firewall. Конечно, существуют брандмауэры и в виде отдельных аппаратных решений, но интеграция брандмауэра с маршрутизатором позволяет снизить совокупную стоимость оборудования.

Брандмауэры анализируют весь трафик между двумя сетями, соединяемыми посредством маршрутизатора, на предмет соответствия его определенным критериям. Если трафик отвечает заданным условиям, то брандмауэр пропускает его через себя (производит маршрутизацию). В противном случае, то есть если не соблюдены установленные критерии, трафик блокируется брандмауэром. Брандмауэры фильтруют как входящий, так и исходящий трафик, а также позволяют управлять доступом к определенным сетевым ресурсам или приложениям. Они могут фиксировать все попытки несанкционированного доступа к ресурсам локальной сети и выдавать предупреждения о попытках проникновения.

Брандмауэры способны осуществлять фильтрацию сетевых пакетов, основываясь на адресах отправителя и получателя и номерах портов, — данная функция называется адресной фильтрацией. Кроме того, брандмауэры могут фильтровать специфические типы сетевого трафика, например HTTP, ftp или telnet, а также способны фильтровать трафик, основываясь на анализе атрибутов сетевых пакетов.

Существуют две методологии функционирования брандмауэров: согласно первой брандмауэр пропускает через себя весь трафик, за исключением того, который отвечает определенным критериям; вторая заключается в том, что брандмауэр, наоборот, блокирует весь трафик, кроме соответствующего определенным критериям.

Возможности брандмауэров и степень их интеллектуальности зависят от того, на каком уровне эталонной модели OSI они функционируют. Чем выше уровень OSI, на основе которой построен брандмауэр, тем выше обеспечиваемый им уровень защиты.

Модель OSI (Open System Interconnection) включает семь уровней сетевой архитектуры. Первый, самый нижний уровень — физический. За ним следуют канальный, сетевой, транспортный, сеансовый уровни, уровень представления и прикладной уровень, или уровень приложений.

Для того чтобы обеспечивать фильтрацию трафика, брандмауэр должен работать как минимум на третьем уровне модели OSI, то есть на сетевом уровне. На этом уровне происходит маршрутизация пакетов на основе преобразования MAC-адресов в сетевые адреса. С точки зрения протокола TCP/IP этот уровень соответствует уровню IP (Internet Protocol). Получая информацию сетевого уровня, брандмауэры способны определить адрес источника и получателя пакета и проверить, допустима ли передача трафика между данными адресатами. Однако информации сетевого уровня недостаточно для анализа содержимого пакета. Брандмауэры, функционирующие на транспортном уровне модели OSI, получают больше

информации о пакетах и являются более интеллектуальными схемами защиты сетей. Если брандмауэры работают на уровне приложений, им доступна полная информация о сетевых пакетах, поэтому такие брандмауэры обеспечивают наиболее надежную сетевую защиту.

Профессиональные брандмауэры захватывают каждый входящий пакет, прежде чем он будет передан адресату и принят его операционной системой. Благодаря этому очень сложно получить контроль над компьютером, защищенным таким брандмауэром.

Все брандмауэры можно условно разделить на четыре категории в соответствии с теми уровнями модели OSI, на которых они работают:

- пакетный фильтр (packet filter);
- шлюз сеансового уровня (circuit-level gateway);
- шлюз прикладного уровня (application-level gateway);
- Stateful Packet Inspection.

Пакетные фильтры

Брандмауэры типа пакетных фильтров являются самыми простыми наименее интеллектуальными. Они работают на сетевом уровне модели OSI или на IP-уровне стека протоколов TCP/IP. Такие брандмауэры в обязательном порядке присутствуют в любом маршрутизаторе, поскольку все маршрутизаторы могут работать как минимум на третьем уровне модели OSI.

В пакетных фильтрах каждый пакет, прежде чем быть переданным, анализируется на предмет соответствия критериям передачи или блокировки передачи. В зависимости от пакета и сформированных критериев передачи брандмауэр может передать пакет, отвергнуть его или послать уведомление инициатору передачи. Критерии, или правила, передачи пакетов могут формироваться на основе IP-адресов источника и получателя, номеров портов источника и получателя и используемых протоколов.

Преимуществом пакетных фильтров является их низкая цена. Кроме того, они практически не влияют на скорость маршрутизации, то есть не оказывают негативного влияния на производительность маршрутизатора.

Шлюзы сеансового уровня

Шлюзы сеансового уровня — это брандмауэры, работающие на сеансовом уровне модели OSI или на уровне TCP (Transport Control Protocol) стека протоколов TCP/IP. Они отслеживают процесс установления TCP-соединения (организацию сеансов обмена данными между узлами сети) и позволяют определить, является ли данный сеанс связи легитимным. Данные, передаваемые удаленному компьютеру во внешней сети через шлюз на сеансовом уровне, не содержат информации об источнике передачи, то есть все выглядит таким образом, как будто

данные отправляются самим брандмауэром, а не компьютером во внутренней (защищаемой) сети. Все брандмауэры на основе NAT-протокола являются шлюзами сеансового уровня (протокол NAT будет описан далее).

К преимуществам шлюзов сеансового уровня относится их низкая цена, к тому же они не оказывают существенного влияния на скорость маршрутизации. Однако шлюзы сеансового уровня не способны осуществлять фильтрацию отдельных пакетов.

Шлюзы прикладного уровня

Шлюзы прикладного уровня, которые также называются проху-серверами, функционируют на прикладном уровне модели OSI, отвечая за доступ приложений в сеть.

На этом уровне решаются такие задачи, как перенос файлов, обмен почтовыми сообщениями и управление сетью. Получая информацию о пакетах на прикладном уровне, такие шлюзы могут реализовывать блокировку доступа к определенным сервисам.

Например, если шлюз прикладного уровня сконфигурирован как Web-проху, то любой трафик, относящийся к протоколам telnet, ftp, gopher, будет заблокирован.

Поскольку данные брандмауэры анализируют пакеты на прикладном уровне, они способны осуществлять фильтрацию специфических команд, например http:post, get и т.д.

Эта функция недоступна ни пакетным фильтрам, ни шлюзам сеансового уровня. Шлюзы прикладного уровня могут также использоваться для регистрации активности отдельных пользователей и для установления ими сеансов связи. Эти брандмауэры предлагают более надежный способ защиты сетей по сравнению со шлюзами сеансового уровня и пакетными фильтрами, однако в значительно большей степени оказывают влияние на уменьшение скорости маршрутизации.

SPI-брандмауэры

Брандмауэры типа Stateful Packet Inspection (SPI) объединяют в себе преимущества пакетных фильтров, шлюзов сеансового уровня и шлюзов прикладного уровня.

Фактически это многоуровневые брандмауэры, которые работают одновременно на сетевом, сеансовом и прикладном уровнях.

SPI-брандмауэры осуществляют фильтрацию пакетов на сетевом уровне, определяют легитимность установления сеанса связи, основываясь на данных сеансового уровня, и анализируют содержимое пакетов, используя данные прикладного уровня.

SPI-брандмауэры обеспечивают наиболее надежную защиту сетей и применяются во многих современных маршрутизаторах.

Тема 9. Протокол NAT

Большинство современных маршрутизаторов поддерживают протокол NAT (Network Address Translation), базирующийся на сеансовом уровне и по сути представляющий собой протокол трансляции сетевых адресов. NAT позволяет реализовать множественный доступ компьютеров локальной (частной) сети (каждый из которых имеет собственный внутренний IP-адрес) в Интернет, используя всего один внешний IP-адрес WAN-порта маршрутизатора. При этом все компьютеры во внутренней локальной сети становятся невидимыми извне, но для каждого из них внешняя сеть является доступной. Протокол NAT пропускает в сеть только те данные из Интернета, которые поступили в результате запроса от компьютера из локальной сети.

Протокол NAT решает две главные задачи:

- помогает справиться с дефицитом IP-адресов, который становится все более острым по мере роста количества компьютеров;
- обеспечивает безопасность внутренней сети — компьютеры локальной сети, защищенные маршрутизатором с активированным NAT-протоколом (устройством NAT), становятся недоступными из внешней сети.

Хотя протокол NAT не заменяет брандмауэр, он все же является важным элементом безопасности.

Принцип работы протокола NAT достаточно прост. Когда клиент внутренней сети устанавливает связь с сервером внешней сети, открывается сокет, определяемый IP-адресом источника, портом источника, IP-адресом назначения, портом назначения и сетевым протоколом. Когда приложение передает данные через этот сокет, то IP-адрес источника и порт источника вставляются в пакет в поля параметров источника. Поля параметров пункта назначения будут содержать IP-адрес сервера и портсервера.

Устройство NAT (маршрутизатор) перехватывает исходящий из внутренней сети пакет и заносит в свою внутреннюю таблицу сопоставления портов источника и получателя пакета, используя IP-адрес назначения, порт назначения, внешний IP-адрес устройства NAT, внешний порт, сетевой протокол, а также внутренние IP-адрес и порт клиента. Затем устройство NAT транслирует пакет, преобразуя в пакете поля источника: внутренние IP-адрес и порт клиента заменяются внешними IP-адресом и портом устройства NAT.

Преобразованный пакет пересылается по внешней сети и в итоге попадает на заданный сервер. Получив пакет, сервер будет направлять ответные пакеты на внешний IP-адрес и порт устройства NAT (маршрутизатора), указывая в полях источника свои собственные IP-адрес и порт.

Устройство NAT принимает эти пакеты от сервера и анализирует их содержимое на основе своей таблицы сопоставления портов. Если в таблице будет найдено сопоставление порта, для которого IP-адрес источника, порт источника, порт назначения и сетевой протокол из входящего пакета совпадают с IP-адресом удаленного узла, удаленным портом и сетевым протоколом, указанным в сопоставлении портов, то NAT выполнит обратное преобразование: заменит внешний IP-адрес и внешний порт в полях назначения пакета на IP-адрес и внутренний порт клиента внутренней сети. Однако если в таблице сопоставления портов не находится соответствия, то входящий пакет отвергается и соединение разрывается.

В некоторых маршрутизаторах возможно отключение NAT-протокола. Однако имеются модели, где NAT-протокол активирован и отключить его невозможно. При этом важно, чтобы маршрутизатор мог частично обойти ограничения NAT-протокола. Дело в том, что не все сетевые приложения пользуются протоколами, способными взаимодействовать с NAT. Поэтому все маршрутизаторы имеют функции, позволяющие наложить ограничения на использование протокола NAT. Сервер, устанавливаемый во внутренней сети и являющийся прозрачным для протокола NAT, называют виртуальным сервером (Virtual Server). Прозрачным для протокола NAT может быть не весь сервер, а лишь определенные приложения, запускаемые на нем. Для того чтобы реализовать виртуальный сервер во внутренней сети, на маршрутизаторе используется технология перенаправления портов.

Перенаправление портов (Port mapping)

Для того чтобы сделать доступными из внешней сети определенные приложения, запускаемые на сервере во внутренней сети (например, Web-сервер или ftp-сервер), в маршрутизаторе необходимо задать соответствие между портами, используемыми определенными приложениями, и IP-адресами тех виртуальных серверов внутренней сети, на которых эти приложения работают. В этом случае говорят о перенаправлении портов (Port mapping). В результате любой запрос из внешней сети на IP-адрес WAN-порта маршрутизатора (но не виртуального сервера) по указанному порту будет автоматически перенаправлен на указанный виртуальный сервер.

Существует несколько способов конфигурирования виртуального сервера. В простейшем случае задается статическое перенаправление портов, то есть IP-адрес виртуального сервера, разрешенный порт приложения на этом виртуальном сервере (Private Port) и порт запроса (Public Port). Если, к примеру, открыт доступ к Web-серверу (порт 80), расположенному во внутренней сети с IP-адресом 192.168.1.10, то при обращении из внешней сети по адресу 10.0.0.254 (адрес WAN-порта) по

80-му порту этот пакет будет перенаправлен маршрутизатором на Web-сервер. Если же происходит обращение по тому же адресу, но по 21-му порту, то такой пакет будет отвергнут маршрутизатором.

Маршрутизаторы позволяют создавать несколько статических перенаправлений портов. Так, на одном виртуальном сервере можно открыть сразу несколько портов или создать несколько виртуальных серверов с различными IP-адресами. Однако при статическом перенаправлении портов нельзя перенаправлять один порт на несколько IP-адресов, то есть порт может соответствовать только одному IP-адресу. Таким образом, невозможно, например, сконфигурировать несколько Web-серверов с разными IP-адресами — для этого придется менять порт Web-сервера по умолчанию и при обращении по 80-му порту в настройке маршрутизатора в качестве Private Port указывать измененный порт Web-сервера.

Большинство моделей маршрутизаторов позволяют также задавать статическое перенаправление группы портов, то есть ставить в соответствие IP-адресу виртуального сервера сразу группу портов. Такая возможность полезна в том случае, если необходимо обеспечить работу приложений, использующих большое количество портов, например игр или аудио/видеоконференций. Количество перенаправляемых групп портов в разных моделях маршрутизаторов различно, но, как правило, их не менее десяти.

Статическое перенаправление портов позволяет лишь отчасти решить проблему доступа из внешней сети к сервисам локальной сети, защищаемой NAT-устройством. Однако существует и обратная задача — обеспечить пользователям локальной сети доступ во внешнюю сеть через NAT-устройство. Дело в том, что некоторые приложения (например, Интернет-игры, видеоконференции, Интернет-телефония и другие, требующие одновременного установления множества сессий) не совместимы с NAT-технологией. Для того чтобы решить эту проблему, используется так называемое динамическое перенаправление портов, которое задается на уровне отдельных сетевых приложений.

В случае если маршрутизатор поддерживает данную функцию, необходимо задать номер внутреннего порта (или интервал портов), связанный с конкретным приложением (как правило, его обозначают Trigger Port), и номер внешнего порта (Public Port), который будет сопоставляться с внутренним портом.

При активации динамического перенаправления портов маршрутизатор следит за исходящим трафиком из внутренней сети и запоминает IP-адрес компьютера, от которого этот трафик исходит. При поступлении данных обратно в локальный сегмент включается перенаправление портов, и данные пропускаются внутрь. По завершении

передачи перенаправление отключается, вследствие чего любой другой компьютер может создать новое перенаправление уже на свой IP-адрес.

Динамическое перенаправление портов используется в основном для служб, предусматривающих кратковременные запросы и передачу данных, поскольку если один компьютер применяет перенаправление данного порта, то другой в это же время перенаправление того же самого порта использовать не может. Если нужно настроить работу приложений, которым необходим постоянный поток данных и которые занимают порт на длительное время, то динамическое перенаправление помогает мало. Однако и в этом случае возможно решение проблемы, заключающееся в использовании демилитаризованной зоны.

DMZ-зона

Демилитаризованная зона (DMZ-зона) — это еще один способ перенаправления портов. Данную возможность предоставляет большинство современных маршрутизаторов. При размещении в зоне DMZ компьютера внутренней локальной сети он становится прозрачным для протокола NAT. Фактически это означает, что компьютер внутренней сети виртуально располагается до брандмауэра. Для ПК, находящегося в DMZ-зоне, осуществляется перенаправление всех портов на один внутренний IP-адрес, что позволяет организовать передачу данных из внешней сети во внутреннюю.

Если, к примеру, сервер с IP-адресом 192.168.1.10, находящийся во внутренней локальной сети, размещен в DMZ-зоне, а сама локальная сеть защищена NAT-устройством, то поступивший из внешней сети по адресу WAN-порта маршрутизатора запрос будет переадресован по любому порту на IP-адрес 192.168.1.10, то есть на адрес виртуального сервера в DMZ-зоне.

Методы аутентификации

В настоящее время существует множество технологий аутентификации пользователей, поддерживаемых маршрутизаторами. Впрочем, если говорить о коммутаторах класса SOHO, то наиболее распространенными методами аутентификации являются следующие:

- использование пароля и имени пользователя;
- применение MAC-адреса;
- использование протокола PPPoE.

Использование пароля и имени пользователя типично для коммутируемых соединений, когда маршрутизатор имеет дополнительный последовательный порт для подключения аналогового модема. В этом случае, как и при традиционной настройке удаленного соединения с применением аналогового модема, в маршрутизаторе при конфигурации

последовательного порта указываются номер телефона провайдера, имя пользователя и пароль.

Использование аутентификации по MAC-адресу встречается довольно редко и подразумевает привязку соединения к MAC-адресу маршрутизатора. Смысл данной технологии достаточно прост: каждое сетевое устройство имеет свой уникальный MAC-адрес длиной 6 байт, или 12 шестнадцатеричных цифр. Подлинность пользователя проверяется провайдером с использованием запроса MAC-адреса маршрутизатора.

Аутентификация по MAC-адресу имеет один подводный камень: при подключении модема к новому маршрутизатору или к компьютеру соединение перестает работать. Для того чтобы этого не происходило, многие модели маршрутизаторов позволяют задавать внешний MAC-адрес.

Использование протокола PPPoE (Point-to-Point Protocol over Ethernet) для аутентификации пользователей поддерживается практически всеми моделями современных маршрутизаторов. Этот протокол является расширением протокола PPP, который был специально разработан для применения протокола TCP/IP в последовательных соединениях, к которым относятся коммутируемые соединения. Фактически, данный протокол предлагает механизм инкапсуляции TCP-пакетов для их передачи по последовательным соединениям. К примеру, протокол PPP используется для организации коммутируемого доступа в Интернет.

PPPoE (как и PPP) не является протоколом аутентификации в чистом виде, однако механизм аутентификации можно рассматривать в качестве составной части этого протокола. При аутентификации по протоколу PPPoE требуется указать имя и пароль.

DHCP-сервер

Любой современный маршрутизатор не только может быть DHCP-клиентом, но и может иметь встроенный DHCP-сервер, что позволяет автоматически присваивать IP-адреса всем клиентам внутренней сети. В настройках DHCP-сервера, как правило, указываются начало и конец диапазона выделяемых IP-адресов. Кроме того, иногда в заданном диапазоне можно указать IP-адреса, которые не будут динамически присваиваться клиентам.

Виртуальные сети VPN

Большинство маршрутизаторов в той или иной степени поддерживают возможность создания виртуальных частных сетей (Virtual Private Networking, VPN), что позволяет организовывать защищенное соединение с локальной (внутренней) сетью извне.

Для создания VPN-сетей, как правило, используются три протокола: сквозной туннельный протокол (Point-to-Point Tunneling Protocol, PPTP),

протокол IPsec и туннельный протокол второго уровня (Layer 2 Tunneling Protocol, L2TP).

PPTP

Сквозной туннельный протокол, созданный корпорацией Microsoft, никак не меняет протокол PPP, но предоставляет для него новое транспортное средство.

PPTP определяет протокол управления вызовами, который позволяет серверу управлять удаленным коммутируемым доступом через телефонные сети общего пользования (PSTN) или цифровые каналы (ISDN),

либо инициализировать исходящие коммутируемые соединения. PPTP использует механизм общей маршрутной инкапсуляции (GRE) для передачи пакетов PPP, обеспечивая при этом контроль потоков и сетевых заторов.

Безопасность данных в PPTP может обеспечиваться при помощи протокола IPsec.

L2TP

Туннельный протокол второго уровня — это своего рода объединение протокола PPTP и протокола эстафетной передачи на втором уровне (Layer 2 Forwarding, L2F), разработанного компанией Cisco. Протокол L2F обеспечивает туннелирование протоколов канального уровня с использованием протоколов более высокого уровня, например IP.

Протоколы L2F и PPTP имеют сходную функциональность, поэтому компании Cisco и Microsoft решили совместно разработать единый стандартный протокол, который и получил название туннельного протокола второго уровня.

IPsec

IPsec — это протокол защиты сетевого трафика путем использования алгоритмов шифрования на IP-уровне. Данный протокол предусматривает два режима функционирования: транспортный и туннельный. В транспортном режиме протокол IPsec применяется к содержимому IP-пакетов, при этом их исходные заголовки остаются видимыми. Туннельный режим инкапсулирует исходные IP-пакеты в IPsec-пакеты с новыми заголовками IP и позволяет эффективно скрывать исходные IP-пакеты.

Режимы функционирования VPN

Существует два режима функционирования VPN: сквозной (Pass Through) и активный. В первом случае маршрутизатор без вмешательства передает входящий и исходящий VPN-трафики, пропуская через себя

инкапсулированные пакеты данных без просмотра их содержимого. Если маршрутизатор поддерживает режим VPN Pass Through, то необходимо только настроить соединение на VPN-клиентах (компьютеры во внутренней сети) таким образом, чтобы клиенты из внутренней сети могли свободно подключаться к серверу VPN снаружи. Однако при совместном использовании NAT- и VPN-туннелей могут возникать проблемы.

В активном режиме маршрутизатор выступает в роли сервера и может устанавливать VPN-соединение с узлом локальной сети, с другими шлюзами и маршрутизаторами или же в обоих направлениях.

Тема 10. Технология VPN

Технология VPN (Virtual Private Network - виртуальная частная сеть) завоевала всеобщее признание, и любой администратор считает своим долгом организовать VPN-каналы для сотрудников, работающих вне офиса.

VPN представляет собой объединение отдельных машин или локальных сетей в виртуальной сети, которая обеспечивает целостность и безопасность передаваемых данных. Она обладает свойствами выделенной частной сети и позволяет передавать данные между двумя компьютерами через промежуточную сеть (internetwork), например Internet.

VPN отличается рядом экономических преимуществ по сравнению с другими методами удаленного доступа. Во-первых, пользователи могут обращаться к корпоративной сети, не устанавливая с ней коммутируемое соединение, таким образом, отпадает надобность в использовании модемов. Во-вторых, можно обойтись без выделенных линий.

Имея доступ в Интернет, любой пользователь может без проблем подключиться к сети офиса своей фирмы. Следует заметить, что общедоступность данных совсем не означает их незащищенность. Система безопасности VPN - это броня, которая защищает всю корпоративную информацию от несанкционированного доступа. Прежде всего, информация передается в зашифрованном виде. Прочитать полученные данные может лишь обладатель ключа к шифру. Наиболее часто используемым алгоритмом кодирования является Triple DES, который обеспечивает тройное шифрование (168 разрядов) с использованием трех разных ключей.

Подтверждение подлинности включает в себя проверку целостности данных и идентификацию пользователей, задействованных в VPN. Первая гарантирует, что данные дошли до адресата именно в том виде, в каком были посланы. Самые популярные алгоритмы проверки целостности данных - MD5 и SHA1. Далее система проверяет, не были ли изменены данные во время движения по сетям, по ошибке или злонамеренно. Таким образом, построение VPN предполагает создание защищенных от

постороннего доступа туннелей между несколькими локальными сетями или удаленными пользователями.

Для построения VPN необходимо иметь на обоих концах линии связи программы шифрования исходящего и дешифрования входящего трафиков. Они могут работать как на специализированных аппаратных устройствах, так и на ПК с такими операционными системами как Windows, Linux или NetWare.

Управление доступом, аутентификация и шифрование - важнейшие элементы защищенного соединения.

Основы туннелирования

Туннелирование (tunneling), или инкапсуляция (encapsulation), - это способ передачи полезной информации через промежуточную сеть. Такой информацией могут быть кадры (или пакеты) другого протокола. При инкапсуляции кадр не передается в сгенерированном узлом-отправителем виде, а снабжается дополнительным заголовком, содержащим информацию о маршруте, позволяющую инкапсулированным пакетам проходить через промежуточную сеть (Internet). На конце туннеля кадры деинкапсулируются и передаются получателю.

Этот процесс (включающий инкапсуляцию и передачу пакетов) и есть туннелирование. Логический путь передвижения инкапсулированных пакетов в транзитной сети называется туннелем.

VPN работает на основе протокола PPP(Point-to-Point Protocol). Протокол PPP разработан для передачи данных по телефонным линиям и выделенным соединениям "точка-точка". PPP инкапсулирует пакеты IP, IPX и NetBIOS в кадры PPP и передает их по каналу "точка-точка". Протокол PPP может использоваться маршрутизаторами, соединенными выделенным каналом, или клиентом и сервером RAS, соединенными удаленным подключением.

Основные компоненты PPP:

Инкапсуляция - обеспечивает мультиплексирование нескольких транспортных протоколов по одному каналу;

Протокол LCP - PPP задает гибкий LCP для установки, настройки и проверки канала связи. LCP обеспечивает согласование формата инкапсуляции, размера пакета, параметры установки и разрыва соединения, а также параметры аутентификации. В качестве протоколов аутентификации могут использоваться PAP, CHAP и др.;

Протоколы управления сетью - предоставляют специфические конфигурационные параметры для соответствующих транспортных протоколов. Например, IPCP протокол управления IP.

Для формирования туннелей VPN используются протоколы PPTP, L2TP, IPsec, IP-IP.

Протокол PPTP - позволяет инкапсулировать IP-, IPX- и NetBEUI-трафик в заголовки IP для передачи по IP-сети, например Internet.

Протокол L2TP - позволяет шифровать и передавать IP-трафик с использованием любых протоколов, поддерживающих режим "точка-точка" доставки дейтаграмм. Например, к ним относятся протокол IP, ретрансляция кадров и асинхронный режим передачи (ATM).

Протокол IPsec - позволяет шифровать и инкапсулировать полезную информацию протокола IP в заголовки IP для передачи по IP-сетям.

Протокол IP-IP - IP-дейтаграмма инкапсулируется с помощью дополнительного заголовка IP. Главное назначение IP-IP - туннелирование многоадресного трафика в частях сети, не поддерживающих многоадресную маршрутизацию.

Для технической реализации VPN, кроме стандартного сетевого оборудования, понадобится шлюз VPN, выполняющий все функции по формированию туннелей, защите информации, контролю трафика, а нередко и функции централизованного управления.

На сегодняшний день VPN - это экономичное, надежное и общедоступное решение организации удаленного доступа. Каким бы ни было расстояние, VPN обеспечит соединение с любой точкой мира и сохранность передачи самых важных данных.

Тема 11. Настройка интерфейса шлюза безопасности серии ZyWALL USG

Настройка Ethernet-интерфейсов в ZyWALL USG производится в веб-конфигураторе в меню **Configuration > Interface >** на закладке **Ethernet**.

Интерфейсы бывают 3 типов: external (внешний интерфейс); internal *(внутренний интерфейс); general (обычный).

На устройствах ZyWALL USG 20/20W/50/100 выбирать тип интерфейса можно только при настройке VLAN и моста (Bridge).

В настройках интерфейсов тип на этих устройствах предустановлен и менять его нельзя.

На устройствах ZyWALL USG 300/1000/2000 на любом интерфейсе доступны для выбора все три типа.

Если установлен интерфейс **external** (внешний), тогда он автоматически в активном режиме включается в системный транк SYSTEM_DEFAULT_WAN_TRUNK, который является NAT-шлюзом по умолчанию для выхода в Интернет.

Это значит, по умолчанию трафик в Интернет от локальных (**internal**) интерфейсов будет маршрутизироваться и транслироваться (NAT) через этот **external**-интерфейс в зависимости от его метрики.

Edit Ethernet

Welcome admin | Logout

Hide Advanced Settings

General Settings

☒ Enable Interface

Interface Properties

Interface Type:

general

i

Interface Name:

ge5

Port:

none

Zone:

DMZ

i

MAC Address:

00:19:CB:B1:69:8E

Description:

(Optional)

IP Address Assignment

☐ Get Automatically

☒ Use Fixed IP Address

IP Address:

192.168.3.1

Subnet Mask:

255.255.255.0

Gateway:

(Optional)

Metric:

0

(0-15)

Interface Parameters

Egress Bandwidth:

1048576

Kbps

i

Ingress Bandwidth:

1048576

Kbps

MTU:

1500

Bytes

Connectivity Check

☐ Enable Connectivity Check

Check Method:

icmp

Check Period:

30

(5-30 seconds)

Check Timeout:

5

(1-10 seconds)

Check Fail Tolerance:

5

(1-10)

☒ Check Default Gateway

0.0.0.0

☐ Check this address

(Domain Name or IP Address)

Рис.11.1. Настройка интерфейса типа *general*

В этом режиме в настройках интерфейса можно включить режим DHCP-клиента для получения IP-адреса и поменять MAC-адрес интерфейса, но нельзя включить DHCP-сервер для выдачи IP-адресов.

Если установлен интерфейс **internal** (внутренний), тогда он автоматически правилами маршрутизации маршрутизируется в Интернет через транк SYSTEM_DEFAULT_WAN_TRUNK с интерфейсами типа **external**.

Также в настройках интерфейса **internal** нельзя указать адрес шлюза, метрику, включить режим DHCP-клиента для получения IP-адреса по DHCP и поменять MAC-адрес, но при этом можно настроить DHCP-сервер для раздачи IP-адресов, настроить привязку по MAC-адресам и запрет на нарушения правил IP/MAC привязки - **IP/MAC Binding**.

Если установлен интерфейс **general** (обычный), доступны все настройки.

При этом автоматические правила маршрутизации не создаются (их нужно прописывать через Policy Route), так же как и не происходит автоматического включения в транк.

При получении IP-адреса автоматически, нельзя будет настроить DHCP-сервер на том же интерфейсе.

Для настройки интерфейса требуется войти в меню **Configuration > Interface >** на закладку **Ethernet**, выбрать интерфейс и нажать **Edit**.

В открывшемся окне **Edit Ethernet** в левом верхнем углу нажать **Show Advanced Settings** для отображения дополнительных параметров.

На рис.11.1. описаны поля при настройке интерфейса типа *general* (так как в нем доступны все настройки).

В разделе **Interface Properties** настраиваются основные параметры интерфейса.

В поле **Interface Type** устанавливается тип интерфейса. Для ZyWALL USG 300/1000/2000 доступны все три типа **internal/external/general**). Для ZyWALL USG 20/20W/50/100 типы предустановлены, и поменять их нельзя.

В поле **Interface Name** указать имя интерфейса.

В поле **Port** можно указать физические порты, на которых работает этот конкретный интерфейс. Настраивается в меню **Configuration > Interface >** на закладке **Port Grouping**.

В поле **Zone** следует указать зону (настраивается в меню **Configuration > Zone**), за которой будет закреплен этот интерфейс. Это нужно для настройки Firewall и сервисов Anti-X.

В разделе **General Settings** (рис.11.2.) можно включить или выключить интерфейс, установив или убрав флажок в поле **Enable Interface**.

В поле **MAC Address** отображается MAC-адрес интерфейса. Если установлен тип **internal** или **general**, его можно поменять ниже в разделе

MAC Address Setting (данный раздел доступен, только если нажать **Show Advanced Settings**).

В разделе **IP Address Assignment** настраивается IP-адрес интерфейса.

Если указать значение **Get Automatically**, то интерфейс будет пытаться получить IP-адрес по DHCP. В дальнейшем в настройках нельзя будет включить DHCP-сервер.

Если указать значение **Use Fixed IP Address**, тогда нужно будет указать в поле **IP Address** - IP-адрес интерфейса, в поле **Subnet Mask** - маску подсети, в поле **Gateway** - IP-адрес шлюза (если он необходим) и в поле **Metric** - значение метрики (доступно, только если указан шлюз).

По метрике устройство определяет основной интерфейс для выхода в Интернет. В разделе **Interface Parameters** настраиваются параметры работы интерфейса.

Этот параметр необходим для настройки балансировки нагрузок между несколькими внешними интерфейсами и для настройки функции управления полосой пропускания **BWM** (Bandwidth Management).

В поле **Ingress Bandwidth** можно указать максимальную ширину канала в килобитах в секунду, которая может быть использована для приема трафика. Данный параметр реализован для будущего использования.

В поле **MTU** можно настроить размер MTU на интерфейсе (поле доступно, только если нажать **Show Advanced Settings**). В разделе **Connectivity Check** можно настроить проверку интерфейса путем отправки на заданный узел через этот интерфейс icmp/tcp-пакетов.

На основе настроенной проверки, в случае если она не прошла, интерфейсу назначается самая высокая метрика, и таким образом он исключается из маршрутизации и считается нерабочим для транка (**Trunk**) и для политики маршрутизации (**Policy Route**).

В поле **Egress Bandwidth** можно указать максимальную ширину канала в килобитах в секунду, которая может быть использована для отправки.

В разделе **DHCP Setting** настраиваются параметры DHCP-сервера (рис.11.2.) или перенаправления DHCP, а также привязка IP-MAC и включение функции **IP/MAC Binding**, которая отслеживает и запрещает трафик, если IP-адрес хоста не совпадает с привязанным MAC-адресом.

В поле **DHCP** можно выбрать значение **None** для отключения DHCP-сервера и функции перенаправления.

Значение **DHCP Server** включает DHCP-сервер, а значение **DHCP Relay** включает перенаправление DHCP-запросов.

Все поля при настройке DHCP-сервера необязательные. Если они не будут заполнены, устройство подставит значения по умолчанию (они не будут отображаться в настройках, но будут при этом работать).

DHCP Setting

DHCP: None

☐ Enable IP/MAC Binding

☐ Enable Logs for IP/MAC Binding Violation

Static DHCP Table

+ Add ✎ Edit ✖ Remove			
#	IP Address	MAC	Description
Page 1 of 1 Show 50 items No data to display			

RIP Setting

☐ Enable RIP

Direction: BiDir

Send Version: 2

Receive Version: 2

☐ V2-Broadcast

OSPF Setting

Area: none

Priority: 1 (0-255)

Link Cost: 10 (1-65535)

☐ Passive Interface

Authentication: None

MAC Address Setting

☒ Use Default MAC Address 00:19:CB:B1:69:8E

☐ Overwrite Default MAC Address 00:00:00:00:00:00

Clone by host

Related Setting

Configure [PPPoE/PPTP](#) i

Configure [WAN TRUNK](#) i

Configure [Policy Route](#) i

Рис. 11.2. Настройки сервера DHCP.

Параметры по умолчанию для каждого поля указаны ниже.

В поле **IP Pool Start Address (Optional)** можно указать начальный IP-адрес для раздачи адресов DHCP-клиентам, а в поле **Pool Size** размер пула IP-адресов (должен входить в подсеть, которая была настроена на интерфейсе). Если поле **IP Pool Start Address (Optional)** не заполнено, устройство будет раздавать IP-адреса начиная с самого младшего IP-адреса, не занятого под адрес интерфейса.

В полях **First, Second, Third DNS Server (Optional)** можно указать соответственно Первый, Второй, Третий DNS-сервер, который будет выслан DHCP-клиенту. В этих полях можно указать значение **Custom Defined** (назначенный вручную) и самостоятельно вписать IP-адрес DNS-сервера. Можно установить значение **From ISP** (от провайдера) и указать внешний (external) интерфейс, с которого нужно получать адреса DNS-сервера.

Если установить значение ZyWALL, то само устройство будет выполнять роль DNS-сервера, кэшируя DNS-имена и обращаясь к внешним DNS-серверам для разрешения запросов.

В системном транке можно указать только метод/алгоритм балансировки **Weighted Round Robin, Least Load First** или **Spillover**.

Если необходимо изменить интерфейсы, их режимы работы или порядок, необходимо создать новый транк в разделе **User Configuration**.

По умолчанию в этих полях установлен режим **Custom Defined** без указанного IP-адреса. Это значит, DHCP-клиенту не высылаются DNS-сервера!

В полях **First, Second WINS Server (Optional)** можно соответственно указать Первый, Второй WINS-сервер для разрешения WINS-имен (служба сопоставления NetBIOS-имен компьютеров с IP-адресами хостов).

В поле **Default Router (Optional)** можно указать шлюз, который будет высылаться DHCP-клиенту. По умолчанию это адрес интерфейса, на котором настраивается DHCP-сервер.

Если в поле **Default Router** будет выбрано значение **Custom Defined** и указан IP-адрес, отличный от IP-адреса интерфейса, тогда ZyWALL USG не будет являться основным шлюзом для DHCP-клиентов, и, соответственно, трафик от них через ZyWALL USG маршрутизироваться не будет без написания статических маршрутов на хосте.

В поле **Lease Time** можно указать время аренды IP-адреса. Можно установить значение **infinite** (без ограничения) или указать, на какое количество дней(days) / часов(hours) / минут(minutes) выдается IP-адрес DHCP-клиенту.

В поле **Enable IP/MAC Binding** можно установить флажок, в этом случае ZyWALL USG будет запрещать трафик с тех IP-адресов, для

которых существует настроенная привязка IP-MAC и MAC-адрес не совпадает.

В поле **Enable Logs for IP/MAC Binding Violation** можно установить флажок, чтобы при обнаружении несоответствия между MAC и IP в логах создавалась соответствующая запись.

В **Static DHCP Table** можно прописывать сочетания IP-MAC для DHCP-сервера и функции **IP/MAC Binding**.

Раздел **RIP Setting** появляется только при нажатии **Show Advanced Settings**. Здесь можно настроить RIP (обмен статическими маршрутами).

В поле **Direction** можно указать направление работы RIP. Можно указать значение **In-Only** (только на прием), **Out-Only** (только на передачу) или **BiDir** (в обе стороны).

В поле **Send Version** можно указать, по какой версии RIP отправляются маршруты: **1**, **2** или по обеим.

В поле **Receive Version** можно указать, по какой версии RIP принимаются маршруты: **1**, **2** или по обеим.

Если установить флажок в поле **V2-Broadcast**, при включении ZyWALL USG высылает маршруты бродкастом по подсети интерфейса.

Если флажок в поле **V2-Broadcast** не стоит, маршруты высылаются мультикастом.

В разделе **OSPF** настраиваются параметры, необходимые для работы OSPF.

В поле **Area** можно указать область (области настраиваются в **Configuration > Network > Routing Protocol > OSPF**). Значение **none** означает, что OSPF выключен.

В поле **Priority** можно указать приоритет. В поле **Link Cost** можно указать стоимость соединения.

Если установить флажок в поле **Passive Interface**, интерфейс не отправляет информацию о маршрутах, а только принимает.

В разделе **MAC Address Setting** можно задать MAC-адрес вручную. Доступно только для интерфейсов **general** и **external**.

При выборе значения **Use Default MAC Address** используется заводской MAC-адрес устройства.

Если требуется использовать другой MAC-адрес на устройстве или указать его вручную, требуется установить значение **Overwrite Default MAC Address** и вписать новый MAC-адрес или нажать кнопку **Clone by host** для клонирования MAC-адреса с хоста.

В окне **Clone MAC Address** указать IP-адрес компьютера, с которого требуется клонировать MAC-адрес, чтобы на интерфейсе заменить MAC-адрес аппаратного шлюза (используется по умолчанию) MAC-адресом сетевого адаптера компьютера.

В разделе **Related Setting** (Связанные настройки) представлены прямые ссылки на настройки PPP-интерфейсов (PPPoE/PPTP), транка (Trunk), политики маршрутизации (Policy Route).

Тема 12. Настройка балансировки внешних интерфейсов в аппаратном шлюзе серии ZyWALL USG

Объединение нескольких физических линий/каналов (2 или более внешних интерфейса) для балансировки осуществляется в веб-конфигураторе ZyWALL USG в меню **Configuration > Network > Trunk**.

Транки могут использоваться для настройки балансировки нагрузки WAN-трафика (для увеличения общей пропускной способности) и резервирования подключения.

Балансировка нагрузки распределяет трафик между несколькими WAN-интерфейсами.

Это позволяет улучшить качество обслуживания и максимальную пропускную способность при использовании нескольких WAN-подключений.

Например, имеются два WAN-подключения к Интернету с различной пропускной способностью.

Можно настроить транк, использующий методы балансировки нагрузки **Spillover** или **Weighted Round Robin** для чувствительного к задержкам трафика (например, видео или VoIP), для которого обычно требуется более высокая пропускная способность.

Для другого типа трафика (нечувствительного к задержкам) вы можете использовать метод балансировки нагрузки **Least Load First** для сбалансированного распределения трафика в соответствии с предельной пропускной способностью канала.

Если используется два WAN-интерфейса для подключения к двум разным провайдерам (например, для резервирования подключения или балансировки нагрузки), то можно использовать совместно политику маршрутов (Policy Route) и транки (Trunk).

Совместное использование транков и политики маршрутов позволит настроить направление определенного типа трафика через наилучшее WAN-подключение.

Если при использовании транка для резервирования внешних подключений один из WAN-интерфейсов становится неактивным, ZyWALL USG может отправить трафик через другой интерфейс транка.

Для настройки транка в аппаратном шлюзе серии ZyWALL USG зайти в веб-конфигуратор в меню **Configuration > Network > Trunk**. Транк на устройствах серии ZyWALL USG выполняет функции балансировки нагрузки трафика и/или резервирования внешних (external) интерфейсов. По умолчанию основным шлюзом является системный транк SYSTEM_DEFAULT_WAN_TRUNK (в разделе **System Default**), который

нельзя удалить или настроить и в который входят внешние (external) интерфейсы (например, wan1, wan2 или ge2, ge3), а также все туннельные ppp-подключения, которые можно настроить на устройстве (они тоже считаются внешними) (рис.12.1.).

The screenshot shows a window titled "Edit System Default". Inside, the "Name" field is set to "SYSTEM_DEFAULT_WAN_TRUNK" and the "Load Balancing Algorithm" is set to "Least Load First". Below this is a table with the following data:

#	Member	Mode	Ingress Bandwidth	Egress Bandwidth
1	wan1	Active	1048576 kbps	1048576 kbps
2	wan1_ppp	Active	1048576 kbps	1048576 kbps
3	cellular1	Active	1048576 kbps	1048576 kbps

Below the table, there are navigation controls: "Page 1 of 1", "Show 50 items", and "Displaying 1 - 3 of 3". At the bottom right, there are "OK" and "Cancel" buttons.

Рис.12.1. По умолчанию основным шлюзом является системный транк

В системном транке можно указать только метод/алгоритм балансировки **Weighted Round Robin**, **Least Load First** или **Spillover**.

Если необходимо изменить интерфейсы, их режимы работы или порядок, необходимо создать новый транк в разделе **User Configuration**.

Для добавления нового транка в список нажать кнопку **Add**. В поле **Name** указать имя транка.

В поле **Load Balancing Algorithm** установить алгоритм балансировки **Weighted Round Robin**, **Least Load First** или **Spillover**.

Если был выбран алгоритм балансировки **Least Load First** или **Spillover**, поле **Load Balancing Index(es)** становится активным.

В этом поле можно указать направление балансировки. Можно установить значение **Outbound**, **Inbound** или **Outbound+Inbound**.

Значение **Outbound** определяет направление трафика с внутреннего LAN-интерфейса на внешний WAN-интерфейс, а **Inbound** - обратное направление.

В столбце **Member** нужно указать внешний интерфейс, который будет входить в транк, и в столбце **Mode** установить режим его работы (**Active** или **Passive**).

Active – активный интерфейс для балансировки. **Passive** – неактивный интерфейс для резервирования (этот интерфейс будет использован, только когда все активные интерфейсы перестанут работать). В столбцах **Ingress/Egress Bandwidth** можно установить пропускную способность интерфейса. После создания транка в разделе **Default WAN Trunk** можно переключить шлюз по умолчанию с **SYSTEM_DEFAULT_WAN_TRUNK** на настроенный (созданный). Если требуется отключить маршрут по умолчанию, то перейти по ссылке **Show Advanced Settings** для отображения дополнительных настроек.

В поле **Enable Default SNAT** снять флажок Отключения маршрута по умолчанию. В этом случае всю маршрутизацию и трансляцию необходимо настраивать через политику маршрутов **Policy Route**.

Тема 13. BWM (Bandwidth Management) в аппаратных шлюзах безопасности серии ZyWALL USG

Функция BWM в ZyWALL USG позволяет распределять общую пропускную полосу по настроенным критериям. BWM работает в зависимости от направления сессии по интерфейсам. Таким образом, в настройках правила IP-адрес источника (Source) - это **IP-адрес инициатора сессии**, а IP-адрес назначения (Destination) - это **IP-адрес, на который сессия открывается**.

Входящий/Исходящий (*inbound/outbound*) трафик считается по отношению к инициатору сессии. Инициатор сессии посылает *outbound* (исходящий) трафик и принимает *inbound* (входящий) трафик. Для *outbound* (исходящего) трафика правила BWM применяются перед отправкой пакета на вышестоящий шлюз. Для *inbound* (входящего) трафика правила BWM применяются перед отправкой инициатору сессии.

Правила BWM настраиваются (рис.13.1) в веб-конфигураторе устройства в меню **Configuration > BWM**

Enable BWM – установленный флажок в этом поле включает функцию BWM на устройстве.

Enable Highest Bandwidth Priority for SIP Traffic – включает автоматическую приоритизацию SIP-трафика. При этом все правила контроля полосы пропускания для SIP-пакетов будут игнорироваться. Также SIP-трафик не будет отображаться в окне статистики **Monitor > AppPatrol Statistic** на закладке **Bandwidth Statistics**.

Для создания правила нажать кнопку **Add**. Далее производится настройка правила.

Enable – активирует правило BWM.

Description (Описание) – необязательное поле.

Add Policy

Create new Object

☒ Enable

Description:
(Optional)

Criteria

User:

Schedule:

Incoming Interface:

Outgoing Interface:

Source:

Destination:

DSCP Code:

Service Type:
☒ Service Object
☐ App Patrol Service

Service Object:

DSCP Marking

DSCP Marking
Inbound Marking:

Outbound Marking:

Bandwidth Shaping

Guaranteed Bandwidth
Inbound:
 kbps (0 : disabled)
Priority:

☐ Maximize Bandwidth Usage
Maximum:
 kbps

Outbound:
 kbps (0 : disabled)
Priority:

☐ Maximize Bandwidth Usage
Maximum:
 kbps

Related Setting

Log

Рис.13.1. Настройка правил BWM в веб-конфигураторе устройства

Раздел **Criteria** служит для описания критериев, по которым устройство должно выбрать трафик: для него будет выделяться определенная полоса и приоритет.

User – авторизованный на шлюзе безопасности пользователь.

Schedule – настройка для работы правила BWM по календарю (расписание настраивается в меню **Configuration > Object > Schedule**).

Incoming Interface (Входящий интерфейс) – интерфейс, с которого иницируются сессии, ограничиваемые правилом BWM.

Outgoing Interface (Исходящий интерфейс) – интерфейс, с которого уходят сессии, ограничиваемых правилом BWM.

Source – IP-адрес источника инициатора сессий, ограничиваемых правилом BWM. **Destination** – IP-адрес назначения сессий, ограничиваемых правилом BWM.

DSCP Code – DSCP-маркер для приоритизации трафика. Можно выбрать трафик только с определенным предустановленным маркером либо указать маркер вручную, установив значение *User Define (preserved* - значит, все пакеты с любыми маркерами или без, *default* - значения маркера 0, как правило это наиболее приоритетный трафик).

Service Type (Тип сервиса):

Service Object – здесь можно выбрать объект Service (**Configuration > Object > Service**), который содержит протокол и номер порта;

App Patrol Service - это объект **Application Patrol** (сигнатуры App Patrol обновляются вместе с IDP - это платная подписка на 1-2 года), который может выявлять определенный сервис или приложения по сигнатурам.

В разделе **DSCP Marking** можно указать, как маркировать пакет при передаче на следующий узел.

Inbound Marking – маркировка исходящих пакетов.

Outbound Marking – маркировка входящих пакетов.

В разделе **Bandwidth Shaping** настраиваются параметры полосы пропускания и приоритизации для выбранного по критериям трафика.

Guaranteed Bandwidth – гарантированная полоса пропускания.

Для входящего по отношению к устройству трафика *Inbound*:

Inbound – значения гарантированной полосы пропускания для входящего трафика.

Priority – приоритет для входящего трафика.

Maximum – максимальная полоса для входящего трафика по данному правилу. Неактивна, если используется **Maximize Bandwidth Usage** (в этом случае максимальная полоса не ограничивается).

Maximize Bandwidth Usage - при включении позволяет правилу использовать всю незадействованную полосу.

Для исходящего по отношению к устройству трафика *Outbound*:

Outbound – значения гарантированной полосы пропускания для исходящего трафика.

Priority – приоритет для исходящего трафика.

Maximum – максимальная полоса для исходящего трафика по данному правилу. Неактивна, если используется **Maximize Bandwidth Usage** (в этом случае максимальная полоса не ограничивается).

Maximize Bandwidth Usage - при включении позволяет правилу использовать всю незадействованную полосу.

В разделе **Related Setting** (Связанные настройки) в поле **Log** можно включить ведение логов (значение *log*) и включение сигнализации (значение *log alert*).

Созданные правила BWM обрабатываются по очереди в зависимости от значения приоритета. Если первым стоит правило с приоритетом 1 и включенным параметром **Maximize Bandwidth Usage**, следующие правила будут обрабатываться и использовать только ту полосу, которая не была использована первым правилом.

Для точной работы правил, с возможностью занимать неиспользованную полосу, нужно в настройках внешних интерфейсов (**Configuration > Interface** кнопка **Edit**) указывать **Egress Bandwidth** в соответствии с предоставляемой провайдером полосой.

Вопросы для самоконтроля

1. Описать модель OSI
2. Описать стек протоколов TCP/IP
3. В чем состоит адресация в IPv4-сетях?
4. Как работает протокол межсетевого взаимодействия IP?
5. Как работает протокол доставки пользовательских дейтаграмм UDP?
6. Как работает протокол надежной доставки сообщений TCP?
7. Как работают протоколы обмена маршрутной информацией стека TCP/IP?
8. В чем состоит маршрутизация небольших сетей?
9. В чем состоит конструкция маршрутизатора?
10. Как работает брандмауэр (firewall)?
11. Как работает протокол NAT?
12. В чем состоит перенаправление портов (Port mapping)
13. Что такое DMZ-зона?
14. Описать методы аутентификации
15. Как работает DHCP-сервер?
16. Описать виртуальные сети VPN.
17. Как работает протокол PPTP?
18. Как работает протокол L2TP?

19. Как работает протокол IPsec?
20. Каковы режимы функционирования VPN?
21. В чем состоит технология VPN?
22. В чем состоят основы туннелирования?

Литература

1. Бройдо В. Л., Ильина О. П., Вычислительные системы, сети и телекоммуникации: Учебник для вузов. 4-е изд., 4 издание, 2011 год, 560 стр., Издательский дом «Питер», ISBN 978-5-49807-875-5
2. Олифер В. Г., Олифер Н. А., Компьютерные сети. Принципы, технологии, — 4-е изд. — СПб. [и др.]: Питер, 2010. — 943, [1] с.: ил. — (Учебник для вузов). — Библиогр.: с. 917.- Алф. указ.: с. 918-943. — ISBN 978-5-49807-389-7((в пер.)).
3. Таненбаум Э. С., Уэзеролл Д., Компьютерные сети. 5-е изд.. 5 издание, 2012 год, 960 стр., ISBN 978-5-459-00342-0, Издательский дом «Питер»,
4. www.citforum.ru
5. www.dlink.ru
6. www.zyxel.ru



В 2009 году Университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа его развития на 2009–2018 годы. В 2011 году Университет получил наименование «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

КАФЕДРА АППАРАТНО-ПРОГРАММНЫХ КОМПЛЕКСОВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Кафедра Аппаратно-программных комплексов вычислительной техники осуществляет переподготовку и повышение квалификации специалистов с широким спектром образовательных программ по следующим направлениям:

- Системный инженер - специалист по эксплуатации аппаратно-программных комплексов вычислительной техники
- Системный администратор - специалист по эксплуатации компьютерных сетей и сопровождению программ 1С:Предприятие
- Обслуживание, диагностика и ремонт персональных компьютеров
- Администрирование вычислительных сетей
- Конфигурирование, администрирование и программирование в среде 1С:

На кафедре ведется подготовка магистров по направлению 230100 «Информатика и вычислительная техника»:

магистерская программа – «Системное администрирование аппаратно-программных комплексов и сетей» - 230100.68.13.

Кафедра является выпускающей по направлению 230100 «Информатика и вычислительная техника» на факультете ВиЗО.

Светлана Михайловна Платунова

Построение корпоративной сети с применением коммутационного
оборудования и настройкой безопасности

Учебное пособие по курсу
«Корпоративные сети»

В авторской редакции

Редакционно-издательский отдел Санкт-Петербургского национального
исследовательского университета информационных технологий, механики
и оптики

Зав. РИО

Н.Ф. Гусарова

Лицензия ИД № 00408 от 05.11.99

Подписано к печати

Заказ №

Тираж 100 экз.

Отпечатано на ризографе

Редакционно-издательский отдел
Санкт-Петербургского национального
исследовательского университета
информационных технологий, механики
и оптики
197101, Санкт-Петербург, Кронверкский пр., 49

