Лекция 8

Протоколы IGRP(Internet Gateway Routing Protocol)

Протокол маршрутизаторов шлюза Интернет.

-фирменный протокол Cisco

IGRP похож на RIP за исключением следующего: Количество Hop увеличено до 255,количество hop не входит в метрику, но учитывается, метрика является составной и в общем случае вычисляется по *четырем параметрам*:

1)*Delay*-топологическая задержка(когда нет задержек, путь открыт)

(24) При идеальных условиях. Может находиться в интервале 10мкс-167сек

2*) Bandwidth*-полоса пропускания

Величина обратная скорости передачи информации(технической скорости)

Техническая скорость учитывает как информационные, так и служебные символы и является величиной, обратной к тактовой частоте передаче в двоичном канале. Информационная скорость учитывает только информационные символы, приведенные к двоичному каналу. Техническая скорость может быть как больше, так и меньше информационной.

Информационная скорость [бит/с]

Техническая скорость [Бод]

24 бита. Скорость от 1200 бит/с-10 Гбит/с

3)*Reliability*-надежность

Надежность передачи.измеряется как частное от R=L/M

l-количество успешно переданных пакетов

M-общее число переданных пакетов.

8 бит

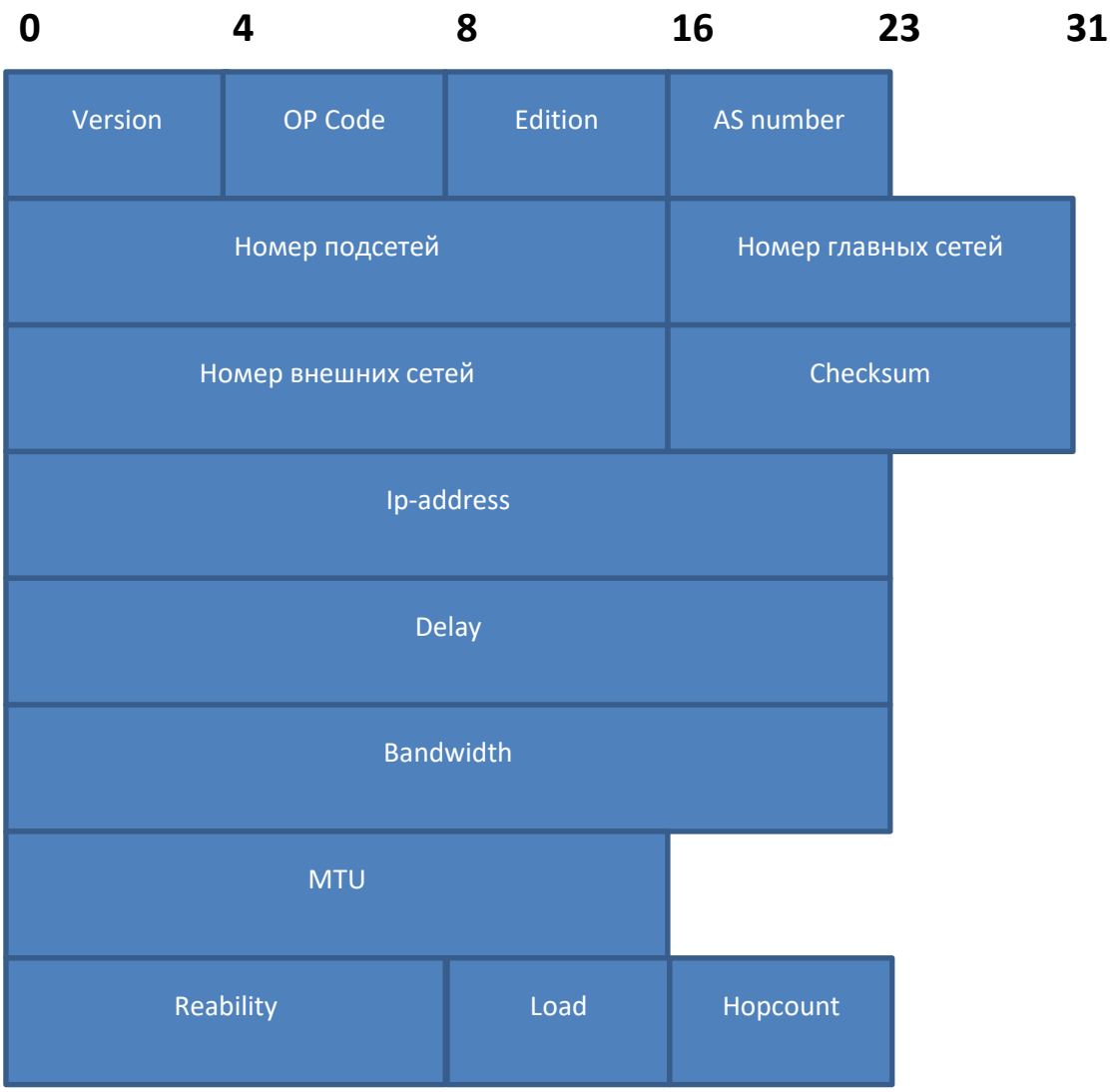
4)*Load-з*агрузка

Обозначает интенсивность передачи по каналу.

При 0-канал свободен

При 255-канал всегда занят

Формат IGRP-сообщения



1.*Version* – номер версии

2.*OP code*-операционный код

1-пакет пакет корректировки

2-запрос

3.*Edition*-номер выпуска

Сначала работы устанавливается ноль.В дальнейшем инкриминируется при кадом изменении маршрутной таблицы.Маршрутизатор-получатель сравнивает номер выпуска с последним номером,полученным от этого соседа.

Если последний номер совпадает с предыдущим,то такой анализ не производится.

4.*AS number*-номер автономной системы

Поскольку роутеры cisco могут быть одновременно подключены к нескольким автнономным системам,то для кждой автономной системы должна быть защищена своя копия программы маршрутизации.сообщается номер,которому принадлежит эта программа(соотв.интерфейса)

5,6,7-используется для работы с внешним протоколом маршрутизации

8.Checksum- контрольная сумма

9.IP-address-IP сети назначения

10,11,13,14

12.MTU-Maximum Transfer Unit максимальная длина IP-пакета на всем пути до сети назначения.

Таймеры IGRP

Аналогичны таймерам RIP со следующими изменениями:

-таймер регулярный рассылок около 90 секунд. Количество маршрутов в сообщении не ограничено, но длина сообщений не может превышать 1500 байт.

-таймер объявления маршрута в три раза больше,чем таймер регулярных рассылок

-таймер «сборки мусора » в семь раз больше таймера регулярных рассылок

90 секунд - значение по умолчанию. Администратор может это время изменить.

Характеристики стабильности TGRP

IGRP обладает рядом характеристик предназначенных для повышения своей стабильности. В их число входит:

1)Временные удерживания изменений

Используются для того, чтобы помешать регулярным сообщениям о корректировке незаконно восстановить в правах маршрут, который, возможно, был испорчен. Когда какой-нибудь роутер выходит из строя, соседние роутеры обнаруживают это через отсутствие сообщений. Далее эти роутеры вычисляют новые маршруты и отправляют сообщение о корректировке своим соседям, внеся соответствующие изменения. Результатом этой деятельности является запуск целой волны корректировки, которые обновляют маршрутные таблицы во всех роутерах сети. Произошедшие изменения в таблице маршрутизации в разных роутерах происходят неодновременно. Поэтому

возможно, что какое-нибудь устройство, которое еще не было оповещено о неисправности в сети, может отправить регулярное сообщение, в котором уже отказавший маршрут будет считаться исправным. Роутер, уже получивший сообщение об отказе и получивший это сообщение с устаревшей информацией, может внести этот неисправный маршрут в свою таблицу и рекламировать его своим соседям. Команды о временном удерживании изменений предписывают роутерам после изменения маршрута некоторое время удерживаться от каких-либо изменений этого маршрута. Период удерживания изменений обычно рассчитывается так, чтобы он был больше периода времени, необходимого для корректировки всей сети в соответствии с какими-либо изменениями маршрутизации.

2)расщепленные горизонты

Понятие о расщепленных горизонтах проистекает из того факта, что никогда не бывает полезным отправлять информацию о маршруте в том направлении, из которого она пришла.

3)корректировки отмены

В то время как расщепленные горизонты позволяют предотвратить зацикливание между соседними роутерами, корректировки отмены маршрута предназначены для борьбы с более крупными петлями. Большое увеличение метрики обычно указывает на появление маршрутных петель. В этом случае посылаются корректировки отмены, чтобы удалить этот маршрут. В реализации IGRP(фирмы cisco) эти сообщения увеличиваются больше, чем на 10%.

Метрика IGRP

Метрика = [K1*пропускная_способность + (K2*пропускная_способность)/(256 - загрузка) + K3*задержка] * [K5/(надежность + K4)].

Если K5 == 0, член надежности отбрасывается. По умолчанию в IGRP K1 == K3 == 1, K2 == K4 == K5 == 0, а загрузка лежит в интервале от 1 до 255.

RIP & IGRP

-динамический

-внутренний

- RIP одно маршрутный

IGRP многомаршрутный

-дистанционно-векторный

Протокол OSPE(Open Shortest Path First)

(кратчайший путь открывается первым)

(Не таблица, а топологическое дерево)

Этот протокол работает на основе алгоритма Dijkstra's algorithm который вычисляет кратчайшие пути от какой-то вершины графа(сеть) до всех остальных вершин.

Шаг 1:расстояние выбранной вершины назначаем 0,а всем остальным ∞

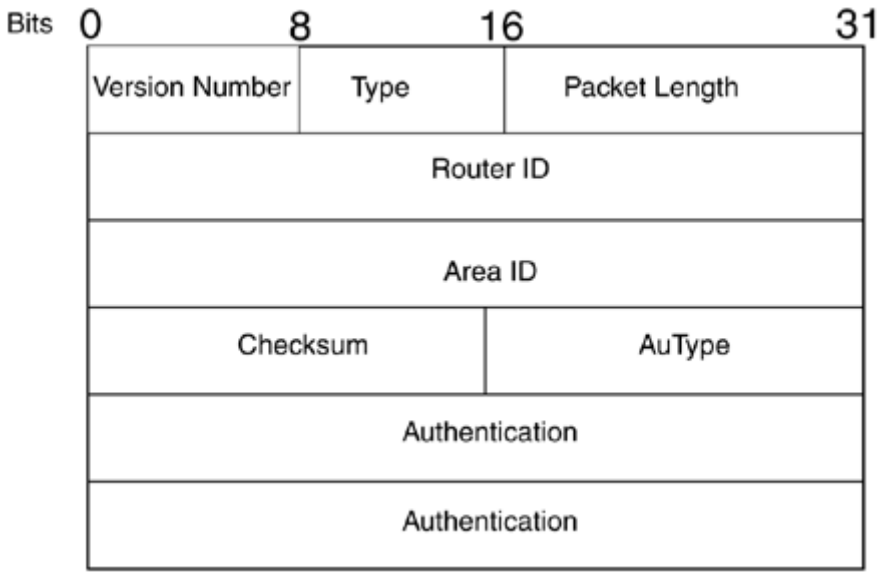
Шаг 2: обходим все соседние вершины по очереди и изменяем все расстояния до них, выбирая наименьшее расстояние.

Шаг 3: убираем рассмотренную вершину и выбираем вершину с наименьшим расстоянием

Значение ребра в $M = \frac{10^8 \, бит/с}{скорость\_бит/с}$

В процессе своей работы роутеры направляют в сеть сообщения разных типов. Формат этих сообщений различен, но все они начинаются со стандартного заголовка.

Стандартный заголовок.



Версия-определяет версию протокола.В настоящее время-версия 2

Тип- идентифицирует функции сообщения(тип сообщения).Всего может быть 5 типов сообщения.

1.Hello(используется для проверки доступности роутера)

2. Data Base Description(описание топологической базы данных)

3. Link State Request(запрос состояния канала)

4. Link State Update(изменение состояния канала)

5. Link State Acknowledgment(подтверждение получения сообщения о статусе канала)

Длина пакета-указывает длину сообщения в байтах, включая заголовок

Идентификатор области-32-х разрядный код бидентифицирующий область, которой этот пакет принадлежит. Все OSPF-пакеты ассоциируются с той или иной областью.

Контрольная сумма-содержит контрольную сумму всего пакета, включая заголовок

Тип идентификации-принимает значение «0» при отсутствии контроля доступа или идентифицирует тип идентификации.

Лекция 9

1)Hello

Отправляется через регулярные интервалы времени для установления и поддержания соседних взаимоотношений. На всех роутерах, подключенных к сети должны быть согласованны ключевые параметры пакетов этого типа: маска сети, периодов приветствования, сигнализации обрыва контакта.

2)Data Base Description

Сообщения описывают содержание топологической БД. Обмен этими сообщениями производится при инициализации смежных маршрутизаторов(т е имеющих одинаковую БД). При описании БД может использоваться несколько таких сообщений. Для обработки таких сообщений используются переклички(poll-response), в которой один из маршрутизаторов определяется как master, а другой как slave. Соответственно master отправляет эти сообщения, а slave должен отвечать за их получение.

3)Link State Request

Запрос о состоянии канала. Обмен такими сообщениями производится после того, как какой-нибудь роутер обнаружит, что часть его топологической БД устарела.

4)Link state Update

Сообщение корректировки состояния канала. Ответ на сообщение запроса о состоянии канала. Эти сообщения используются для тиражирования LSA(Link State Advertisement) объявление о состоянии канала. Всего 4 подтипа таких сообщений. Каждое из них несет информацию о части сети.

1.RLA(Router Link Advertisement)

-это сообщение о каналах роутера. Они описывают собранные данные о состоянии каналов роутера, связывающих его с конкретной областью. Любой роутер отправляет RLA-сообщение для каждой области, к которой он принадлежит(эти сообщения отправляются внутри области).

2.NLA(Network Link Advertisement)

сообщение о сетевых каналах. Они описывают все роутеры, которые подключены к сети с множественным доступом (например, Ethernet) и отправляют через область, содержащую данную сеть с множественным доступом.

3.SLA(Summery Link Advertisement)

Суммарные сообщения о каналах. Суммирует маршруты к пунктам назначения, находящихся вне какой-либо области, но в пределах данной АС(несколько сетей).Они генерируются роутерами границы области и отправляются через данную область. В

стержневую область посылаются сообщения только внутри областных роутеров. В других областях рекламируются как внутриобластные, так и межобластные маршруты.

4.AS external

Сообщения о внешних каналах автономной системы. Описывают какой-либо маршрут к одному из пунктов назначения, который является внешним для данной автономной системы. Сообщения о внешних канальных автономной системы генерируются граничными роутерами автономной системы. Этот тип сообщений является единственным, который продвигается во всех направлениях данной автономной системы.

5) Link State Acknowledgment

Сообщения подтверждения состояния канала. Подтверждает сообщения корректировки состояния канала. Сообщения корректировки состояния канала должно быть четко подтверждены, что является гарантией надежности процесса адресации сообщений корректировки состояния канала через какую-либо область.

Несмотря на то, что разные протоколы маршрутизации работают поверх как протокола сетевого уровня, так и транспортного уровня, все их принято относить к протоколам *сетевого уровня.*

Протоколы EGP, BGP

Являются внешними(меж доменными) протоколами маршрутизации, при помощи которых организуется передача данных между автономными системами, в каждй из которых может осуществляться разная маршрутная политика.

EGP-Exterior Gateway Protocol

(исторически первый) Внешний протокол шлюзов.

Основной функцией EGP является обеспечение взаимодействия различных автономных систем так, что для конечного пользователя все разнородные группы и домены сетей Internet будут казаться единым плоским пространством. Строго говоря, протокол EGP не является протоколом маршрутизации. Так как для его функционирования необходимым условием является отсутствие петель в маршрутах между АС, т е от одной АС до другой существует лишь один путь.

Такие протоколы называются *протоколами досягаемости.*

BGP(Border Gateway Protocol)

В отличие от EGP протокол BGP-для обнаружения маршрутных петель. BGP-протокол маршрутизации между АС, специально созданными для применения в Internet. Работает поверх протокола TCP. Это позволяет не нагружать сервисы обработки BGP механизмами фрагментации или обеспечение достоверности доставки пакета. Основным предназначением BGP является обеспечение обмена информацией с другими BGP системами о досягаемости сетей или хостов. Эта информация должна содержать набор маршрутов в данной сети, т е должны быть указаны все промежуточные автономные системы. Такой информации вполне достаточно для того, чтобы построить граф соединений между АС и проконтролировать возможные маршрутные петли. На основании этих данных BGP выбирает оптимальный маршрут и передает эту информацию своим соседям. Два соседних роутера BGP устанавливают друг с другом соединение, обмениваясь сообщениями открытия и согласования параметров соединения. Инициализация потока данных включает в себя передачу всей маршрутной таблицы. С изменением таблицы отправляются соответствующие корректировки. Периодически роутеры отправляют друг другу сообщения подтверждения своей работоспособности. А при возникновении ошибок передаются сообщения о них. BGP не требует периодического обновления всей маршрутной таблицы, хотя BGP поддерживает маршрутную таблицу всех возможных путей какой-либо конкретной сети. В своих сообщениях он объявляет только об основных оптимальных маршрутах. Показатели оптимальности метрики BGP представляют собой числа6 характеризующие степень предпочтения какого-нибудь конкретного маршрута. Эти показатели обычно определяются администратором сети с помощью конфигурации файлов. Степень предпочтения того или иного маршрута может базироваться на любом числе критериев:

-число промежуточных АС

-тип канала

-стабильность

-быстродействие

-надежность

И др.

Необходимо отметить, что, если в дистанционно-векторных протоколах рассматривается только один параметр оптимальности-метрика и процедура сравнения маршрутов сводится к сравнению двух чисел, то в BGP, где каждая из АС, встречающаяся на пути пакета может определить маршрут по своим собственным критериям, необходимо учитывать несколько параметров, одновременно виляющих на оптимизацию маршрута.

Для установления соединения, коррекции маршрута, уведомления друг друга BGP-маршрутизаторы используют систему сообщений.

Лекция 10.

Типы сообщений BGP.

Всего существует 4 типа сообщений:

1) *Open*
   Для установки отношений соседства и обмена базовыми параметрами. Отправляется сразу после установки TCP соединения.
2) *Up-date*
   Для обмена информацией маршрутизацией. В BGP передается не вся маршрутная таблица, а лишь то, что было изменено.
3) *Notification*
   Эти сообщения передаются тогда, когда в BGP возникают ошибки. После отправки этого сообщения сессия с соседом прерывается.
4) *Keep alive*
   Для поддержания отношений соседства а также для обнаружения неактивных соседей. Сообщения аналогичны Hallo(в OSPF) и отправляются через регулярные промежутки времени только соседям. Эти сообщения состоят только из заголовка и никаких данных не несут.

Протокол TCP(Transfer Control Protocol)

Протокол управления передачей (ключевой в работе сети Internet).

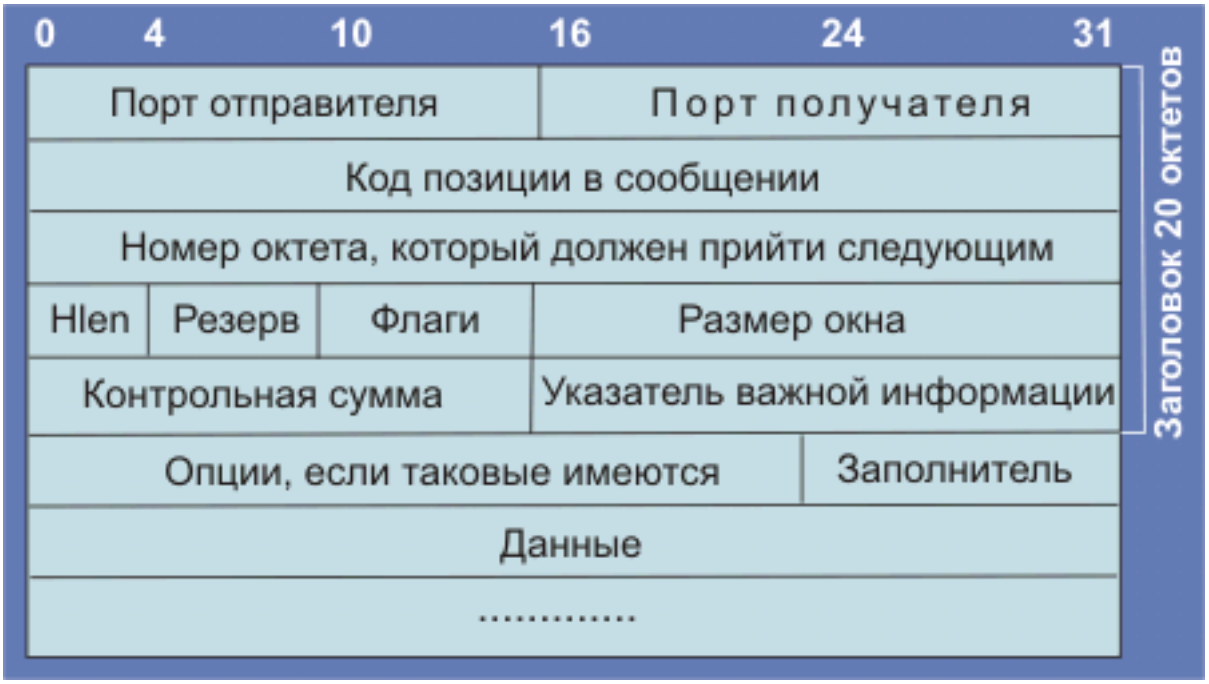
TCP-протокол:
-транспортного уровня
-полнодуплексный
-с установлением соединения

Этот протокол обеспечивает надежный обмен данными между двумя узлами. В основе взаимодействия по протоколу TCP лежит понятие *socket*(соединить)
Socket=ip address + number of port

Где *port*- это либо идентификатор протокола прикладного уровня, либо непосредственно какого-то приложения.

Сегмент- протокольный блок данных.(Как в Ethernet- кадр, в IP-пакет)

Формат TCP сообщения



Порт источника- идентификатор либо протокола прикладного уровня, либо приложения, которое отправляет сегмент

Порт назначения- идентификатор либо протокола, либо приложения.

НАИЗУСТЬ!
23-Telnet
25-SMTP
110-POP
80-HTTP
21, 22-FTP
53-DNS
Номера портов с 0 по 255 зарезервированы за протоколами прикладного уровня общего назначения на сервере.
Номера портов с 255 по 1023 зарезервированы за протоколами с приложениями крупных компаний разработчиков ПО
Номера портов с 1024 по 5000 используются любыми разработчиками ПО.

Номер последовательности- протокол TCP является асинхронным(т е он передает данные по мере их поступления от приложения). Это поле указывает на номер первого байта в поле ДАННЫЕ в общем потоке данных. Первый номер последовательности берется из таймера компьютера (узла).
Номер подтверждения- это номер байта, который узел ожидает принять от другой стороны, подтверждая при этом, что все предыдущие байты приняты успешно.
Длина заголовка- количество 32х разрядных слов в заголовке.
Флаги:
Urgement*(URG)*- указатель. Если 1: поле «указатель» имеет смысл. Узел, получивший сегмент с установленным флагом URG должен немедленно(не дожидаясь получения всех данных) отправить указанное количество байт.

Acknowledgment*(ACK)*-подтверждение. Если 1: то поле номер подтверждения имеет смысл.

Push*(PSH)*- если флаг установлен, то все промежуточные узлы должны немедленно передавать данные, не оптимизируя сегменты.

Reset*(RST)*- используется для информирования удаленного узла о том, что компьютер перезагрузился и дальнейший обмен данными невозможен с прежними параметрами.

Synchronization- для установления соединения в начале сеанса.

Finish*(FIN)*- для разрыва соединения.

Размер окна- указывает другой стороне, сколько байт узел может принять в следующем сегменте.

Контрольная сумма- в отличие от IP вычисляется по всему сегменту.

Указатель границы срочных данных- количество байт в поле «Данные» , начиная с первого, которые должны быть переданы приложению немедленно по их получению.

Взаимодействие между узлами по протоколу TCP начинается с «установления соединения», которое осуществляется в три этапа(three hand shake)

1) Обычно инициатором установления соединения является клиент, который посылает серверу сегмент без данных с установленным флагом SYN и актуальным номером последовательности
2) Сервер отправляет клиенту сегмент, состоящий из одного заголовка с установленным флагом SYN, актуальным номером последовательности, номером подтверждения и установленным флагом ACK.
3) Клиент посылает сегмент с установленным флагом SYN, ACK, номер подтверждения и номер последовательности.
   В некоторых реализациях TCP уже на этом этапе клиент может начать передачу данных.после этого соединение считается установленным и протокол TCP переходит в следующее состояние: «передача данных» .
   Передача ведется асинхронно.
   По завершению обмена данными либо сервер, либо клиент инициирует разрыв соединения, который осуществляется в 2 этапа:
   1) Инициатор разрыва посылает сегмент с установленным флагом FIN.
   2) Другая сторона отвечает тем же. После чего соединение прерывается.

Лекция 11.

Таймеры TCP.

1) **Таймер контроля подтверждения.**
   При посылке очередного сегмента копия сегмента помещается в буфер и запускается этот таймер. Если по истечении работы таймера подтверждение принятия сегмента не получено, то этот сегмент отправляется вновь до тех пор, пока не будет получено подтверждение или соединение не будет разорвано.

2) **Таймер открытия окна**
   Может случиться такая ситуация, что в сегменте поле «размер окна»=0. Если получен такой сегмент, то запускается этот таймер. По истечении его работы отправляется сегмент ЗОНД(«данные»=0) с целью проверки, не открылось ли окно. Если окно открылось, то можно возобновить передачу данных.Если же окно по-прежнему закрыто, то таймер запускается вновь.

3) **Таймер контроля работоспособности.**
   Однажды установленное соединение в общем случае может существовать «вечно». Т е соединение установлено, но данные могут не передаваться. Для получения информации о работоспособности удаленного узла и существует этот таймер.
   После каждой отправки сегмента он запускается. Если он срабатывает ДО следующей отправки(т е данные в течение этого времени НЕ передавались), то посылается сегмент ЗОНД для проверки работоспособности.
   Возможны 4 ситуации:
   1.Удаленный узел работоспособен(в этом случае посылается подтверждение сегмента и таймер запускается вновь)
   2. Удаленный узел не работоспособен(если в течение нескольких сегментов нет подтверждения, то TCP-соединение разрывается принудительно.)
   3. Удаленный узел работоспособен, но недостижим(проблемы сети) См. пункт 2

4.Удаленный узел работоспособен, достижим, НО перезагрузился и в ответ будет получен сегмент RST и соединение должно быть принудительно закрыто.
Если потребность в передаче данных все еще существует, то соединение нужно устанавливать вновь.

4) **2MSL(Maximum Segment Lifetime)**
Этот таймер используется при закрытии соединения для того, чтобы избежать некорректной работы приложений.
В течение работы этого таймера не может быть использован порт, который был в этом соединении, для адекватности работы приложения. За время работы этого таймера все сегменты гарантированно «умрут».

RTT.

Все таймеры TCP для своей работы используют параметр ROUND-TO-TRIP(время прохождения сегмента туда и обратно), который постепенно измеряется и конвертируется во время существования соединения.
$T_{кп}$=RTT+Δ

Типы серверов и портов.

Порт может быть открыт *активно* или *пассивно.*
Пассивно ион открывается,как правило, насервере и ждет входящих запросов на соединение.
Активно же порт обычно открывается на клиентае,который тут же отправляет запрос на установление соединения.(Three-hand-shake)

Сервер может обрабатывать каждый запрос в порядке очереди,а может путем создания для каждого запроса копии сервера,которая работает с этим запросом. По окончании работы с запросом копия уничтожается.
Протокол TCP работает, как *цифровой автомат* с 11ю состояниями, переход между которыми строго регламентирован.

UDP(User Datagramm Protocol)

-протокол транспортного уровня, который выполняет функции аналогичные TCP, но без установления соединения.
Широко использкется протоколами реального времени для передачи потоковых данных.(аудио,видео) Т е его рекомендуется примнять там,где время доставки данных важнее,чем получение данных с ошибками или их потери.
Формат заголовка UDP

Биты	0 - 15	16 - 31
0-31	Порт отправителя (Source port)	Порт получателя (Destination port)
32-63	Длина датаграммы (Length)	Контрольная сумма (Checksum)
64-...	Данные (Data)	

В UDP-datagramma.

Утилита traceroute.

-маленькая подпрограмма.Узкоспециализированная.
Во всех ОС эта утилита называется командой.
На узле-отправителе генерируется UDP-datagramma с номером порта получателя 35 367, который запрещено использовать для приложений.
Эта UDP-datagramma инкапсулируется в IP-пакет, где поле TTL=1.

Прикладные протоколы.

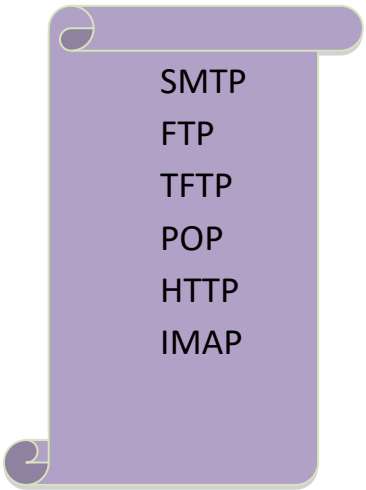
Telnet(23).

Протокол telnet один из первых протоколов Internet и предназначен для управления удаленным хостом (осуществление входа в ОС удаленного хоста).
Протокол telnet изначально был ориентирован на применении гетерогенных(неоднородных) сетях с присущим им разнообразием произвольных компьютеров с различной ОС.
Концептуальной основаой telnet, во многом обеспечивающей ео универсальность, служит понятие сетевого виртуального терминала NVT.
NVT- абстрактное устройство, определяющее простейший,ообщенный полнофункциональный терминал.

NVT отведена роль эталона, с которого и на который клиент и сервер проецируют поддерживаемые ими реальные терминалы.
NVT определен как символьное устройство с клавиатурой и прнтером. Вводимые с клавиатуры данные посылаются серверу, а получаемые от сервера данные печатаются на принтере.

Код NVT ASCII. (American Standart Code for Information Interchange).

Стандартизирован ANSI.
Терминал NVT ASCII называют повсеместно принятый в сетях с протоколами семейства TCP/IP метод кодирования текстовых сообщений, основанный на американском варианте таблицы 7-ми битовых символов ASCII.
Каждый символ пересылается в 8-ми разрядном байте с нулевым старшим битом. Признак конца строки передается парой битов.
CR-возврат коретки 10
LF-перевод строки 13
Код NVT ASCII используется во многих протоколах прикладного уровня.



Команды Telnet.

Telnet является байт-ориентированным протоколом. Команды передаются в символах ASCII в обоих направлениях.

Префиксом команды служит код OxFF(255).

IAC(Internet AS Command).

Команды:

1)EOF(конец файла)

2)EOR(конец записи)

3)BRK(код клавиши break)

4)EC(стереть символ)

5)EL(стереть строку)

6)WILL 251(могу использовать опцию)

7)WONT 252(не могу использовать опцию)

8)DO 253(разрешаю партнеру использовать опцию)

9)DON'T 254(запрещаю партнеру использовать опцию)

10)IAC 255(префиксный байт команды)

Установление опций.

Предполагается, что обе стороны взаимодействуют в Telnet по протоколу NVT.Сразу после установления TCP-соединения в сеансе Telnet наступает обязательная фаза предварительных переговоров. Стороны договариваются о дополнительных ограничениях и возможностях, определяемых опциями в Telnet. Договаривающиеся стороны равноправны. Каждая может предложить или разрешить другой стороне использовать или наоборот запретить использовать любую из опций.

Диалоги по поводу конкретной опции: одна сторона высылает другой объявление одного из 4х типов(6-9).

По правилам Telnet на предложение применить ту или иную опцию (WILL/DO) другая сторона в зависимости от обстоятельств может ответить либо соглашением, либо отказом. Однако всякому объявлению, отрицающему применение некоторой опции (WONT/DONT) другая сторона обязана подчиниться и подтвердить своим ответом.

6 сценариев.

ОТПРАВИТЕЛЬ	ПОЛУЧАТЕЛЬ	КОММЕНТАРИЙ
1) WILL	DO	Отправитель хочет применить опцию. Получатель согласен ее применить.
2) WILL	DONT	Отправитель хочет применить опцию. Получатель НЕ согласен ее применить.
3) DO	WILL	Отправитель хочет, чтобы получатель применил опцию. Получатель согласен ее применить.
4) DO	WONT	Отправитель хочет, чтобы получатель применил опцию. Получатель НЕ согласен ее применить.
5) WONT	DONT	Отправитель НЕ хочет, применять опцию. Получатель обязан согласиться.
6) DONT	WONT	Отправитель НЕ хочет, чтобы получатель применил опцию. Получатель отказывается ее применять.

Формально обе, взаимодействующие по протоколу Telnet стороны, выступают равноправными партнерами как на фазе переговоров об опциях, так и по большей части и в процессе работы в сеансе Telnet.

Однако в функциональном смысле взаимодействия при удаленном входе стороны существенно различны: *клиент решает свои задачи, сервер свои.*

Субопции Telnet.

Для некоторых опций по мимо разрешения(enable) или запрета(disable) на их проведение требуется задание дополнительных уточняющих параметров. Например в опции «тип терминала» клиент должен сообщить каким именно терминалом располагает пользователь.

Для подобных случаев в Telnet предусмотрены переговоры с привлечением субопций.

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Режимы передачи Telnet.

В настоящее время протокол Telnet предоставляет клиентам и серверам возможности 4х режимов обмена:

1) Полудуплекс(half-duplex-mode)
   Использовался как основной. Но со временем его область применения сузилась.
   NVT был изначально определен как half-duplex-mode, которое буферизирует все вводимые с клавиатуры символы и не высылает их серверу до тех пор, пока не получит от него команду «продолжить передачу» GA(Go Ahead).
   При этом NVT сам выполняет эхопечать, а серверу посылаются лишь окончательно отредактированные(готовые) строки набранных с клавиатуры символов. Такой режим доступен любому простейшему терминалу(в этом смысле универсален).
   Но он становится не эффективным при работе с современным более совершенным терминалом, который способен поддерживать duplex.
2) Посимвольный режим(Character-at-a-time-mode)
   В этом режиме все вводимые с клавиатуры символы по одному пересылаются серверу, а в обратном направлении высылается эхосимвол. Такая передача имеет два минуса:
   -из-за задержки в сети время вывода символа на экран может быть значительным, что предоставляет неудобства.

-в несколько раз выше(по сравнению с half-duplex-mode) сетевой трафик.

Тем не менее в большинстве современных реализаций Telnet посимвольный режим принят в качестве основного.

3) Условно-строчный(line-at-a-time-mode)

   Использовал некоторые непредназначенные для этого опции Telnet для того, чтобы использовать передачу по строкам.

   В настоящее время практически не применяется и вместо него используется «законный», введенный в стандарт Telnet «построчный режим».

4) Построчный режим(line-mode)

   Регулируется специальной опцией line mode и поддерживается специальной реализацией Telnet.

   Передача осуществляется построчно. Эховывод на клиенте, но возможен duplex.

Telnet as universal client.

В большинстве современных ОС реализована утилита Telnet, которая позволяет соединиться с сервером не только по 23 порту, но и по любому другому.

Т е Telnet можно использовать для обмена почты, передачи файлов, отправки HTTP-запросов и получения WEB-страниц в текстовом виде.

Лекция 12.

Технология NAT(Network Address Table).

Технология NAT предназначена для того чтобы из внутренней сети с внутренним IP-адресом пакеты могли передаваться во внешнюю сеть(Internet) и обратно.

Существует три технологии работы NAT.

*1)**Статический NAT***

Применяется в том случае, когда провайдером выделено несколько внешних IP адресов и администратор жестко прописывает соответствие внешних и внутренних IP-адресов. При посылке пакета из внутренней сети во внешнюю сеть в IP-пакете адрес отправителя заменяется на один из выделенных провайдером IP-адресов. При получении пакета из внешней сети граничный шлюз заменяет IP-адрес получателя в IP-пакете на внутренний IP-адрес в соответствии с таблицей и передает модифицированный пакет во внутреннюю сеть.

*2)**Динамический NAT***

Внешние IP-адреса из PULL'а предоставленных провайдером выделяются динамически при поступлении пакетов из внутренней сети с внешним адресом получателя. (нет жесткого соответствия)

*3)**Перегруженный NAT/Маскарад/PAT***

Технология «проброс портов» Port Forwading.

(статические IP-адреса)

Команды Telnet.

***Telne*t** является байт-ориентированным протоколом. Команды передаются в символах ASCII в обоих направлениях.

Префиксом команды служит код OxFF(255).

IAC(Internet AS Command).

Команды:

1)EOF(конец файла)

2)EOR(конец записи)

3)BRK(код клавиши break)

4)EC(стереть символ)

5)EL(стереть строку)

6)WILL 251(могу использовать опцию)

7)WONT 252(не могу использовать опцию)

8)DO 253(разрешаю партнеру использовать опцию)

9)DON'T 254(запрещаю партнеру использовать опцию)

10)IAC 255(префиксный байт команды)

Установление опций.

Предполагается, что обе стороны взаимодействуют в Telnet по протоколу NVT.Сразу после установления TCP-соединения в сеансе Telnet наступает обязательная фаза предварительных переговоров. Стороны договариваются о дополнительных ограничениях и возможностях, определяемых опциями в Telnet. Договаривающиеся стороны равноправны. Каждая может предложить или разрешить другой стороне использовать или наоборот запретить использовать любую из опций.

Диалоги по поводу конкретной опции: одна сторона высылает другой объявление одного из 4х типов(6-9).

По правилам Telnet на предложение применить ту или иную опцию (WILL/DO) другая сторона в зависимости от обстоятельств может ответить либо соглашением, либо отказом. Однако всякому объявлению, отрицающему применение некоторой опции (WONT/DONT) другая сторона обязана подчиниться и подтвердить своим ответом.

6 сценариев.

ОТПРАВИТЕЛЬ	ПОЛУЧАТЕЛЬ	КОММЕНТАРИЙ
1) WILL	DO	Отправитель хочет применить опцию. Получатель согласен ее применить.
2) WILL	DONT	Отправитель хочет применить опцию. Получатель НЕ согласен ее применить.
3) DO	WILL	Отправитель хочет, чтобы получатель применил опцию. Получатель согласен ее применить.
4) DO	WONT	Отправитель хочет, чтобы получатель применил опцию. Получатель НЕ согласен ее применить.
5) WONT	DONT	Отправитель НЕ хочет, применять опцию. Получатель обязан согласиться.
6) DONT	WONT	Отправитель НЕ хочет, чтобы получатель применил опцию. Получатель отказывается ее применять.

Формально обе, взаимодействующие по протоколу Telnet стороны, выступают равноправными партнерами как на фазе переговоров об опциях, так и по большей части и в процессе работы в сеансе Telnet.

Однако в функциональном смысле взаимодействия при удаленном входе стороны существенно различны: *клиент решает свои задачи, сервер свои.*

Субопции Telnet.

Для некоторых опций по мимо разрешения(enable) или запрета(disable) на их проведение требуется задание дополнительных уточняющих параметров. Например в опции «тип терминала» клиент должен сообщить каким именно терминалом располагает пользователь.

Для подобных случаев в Telnet предусмотрены переговоры с привлечением субопций.

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Режимы передачи Telnet.

В настоящее время протокол Telnet предоставляет клиентам и серверам возможности 4х режимов обмена:

1) Полудуплекс(half-duplex-mode)

   Использовался как основной. Но со временем его область применения сузилась.

   NVT был изначально определен как half-duplex-mode, которое буферизирует все вводимые с клавиатуры символы и не высылает их серверу до тех пор, пока не получит от него команду «продолжить передачу» GA(Go Ahead).

   При этом NVT сам выполняет эхопечать, а серверу посылаются лишь окончательно отредактированные(готовые) строки набранных с клавиатуры символов. Такой режим доступен любому простейшему терминалу(в этом смысле универсален).

   Но он становится не эффективным при работе с современным более совершенным терминалом, который способен поддерживать duplex.

2) Посимвольный режим(Character-at-a-time-mode)

   В этом режиме все вводимые с клавиатуры символы по одному пересылаются серверу, а в обратном направлении высылается эхосимвол. Такая передача имеет два минуса:

   -из-за задержки в сети время вывода символа на экран может быть значительным, что предоставляет неудобства.

   -в несколько раз выше(по сравнению с half-duplex-mode) сетевой трафик.

   Тем не менее в большинстве современных реализаций Telnet посимвольный режим принят в качестве основного.

3) Условно-строчный(line-at-a-time-mode)

   Использовал некоторые непредназначенные для этого опции Telnet для того, чтобы использовать передачу по строкам.

   В настоящее время практически не применяется и вместо него используется «законный», введенный в стандарт Telnet «построчный режим».

4) Построчный режим(line-mode)

   Регулируется специальной опцией line mode и поддерживается специальной реализацией Telnet.

   Передача осуществляется построчно. Эховывод на клиенте, но возможен duplex.

Telnet as universal client.

В большинстве современных ОС реализована утилита Telnet, которая позволяет соединиться с сервером не только по 23 порту, но и по любому другому.

Т е Telnet можно использовать для обмена почты, передачи файлов, отправки HTTP-запросов и получения WEB-страниц в текстовом виде.

Лекция 13.

(продолжение лекции 4)

Глобальный адрес.

	/23	/32	/48	/64
реестр	ISP	площадка	подсеть	Идентификатор интерфейса

Реестр- глобальный адрес.

ISP- интернет-провайдер. Ему выдаются IP-адреса и он ими распоряжается.

Площадка- то, что провайдер выдает нам.

Например 192.168.3.0/24 не можем менять.

Подсеть- то, чем мы располагаем.

Локальный адрес.

FE	Ø	Идентификатор интерфейса

FE означает, что адрес задан именно локально.

Идентификатор позволяет быть IP-адресу уникальным.

Задание IP-адресов.

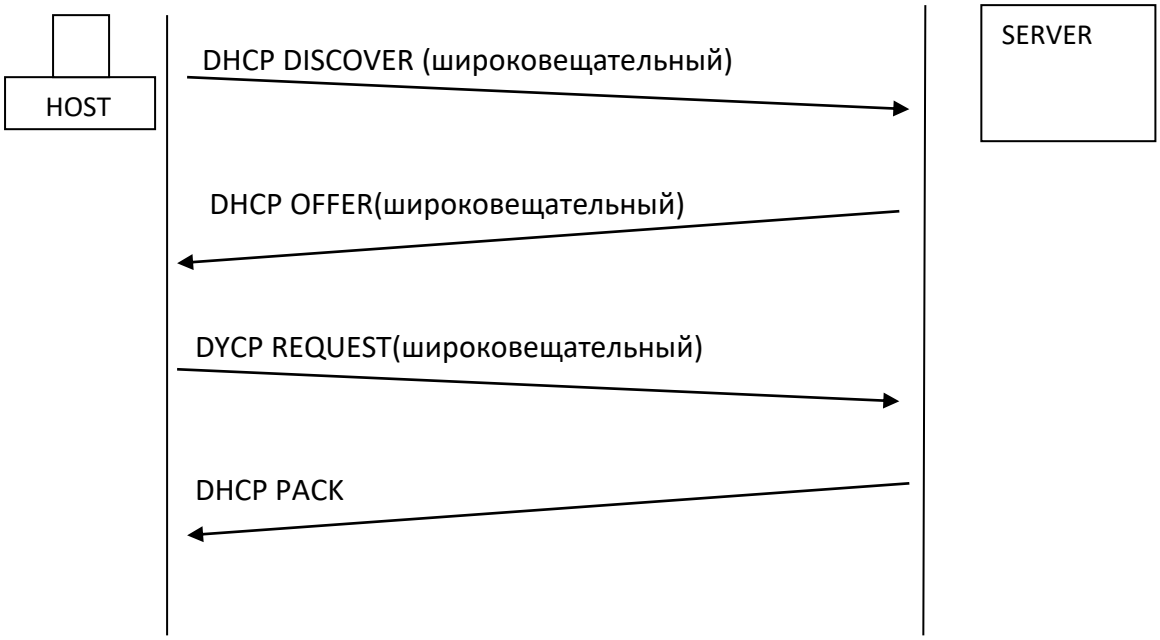
1) В ручную.(сознательно меняем сами)
2) Назначение по стандарту EUI-64.
   (->идентификатор интерфейса.Берется MAC-адресс и разделяется на две части, вставляя между этими частями последовательность FFFE).
3) Автоконфигурация без сохранения состояния.

(по технологии Plug-n-Play)

4) DHCP(Dynamic Host Configuration Protocol)

Протокол позволяет в динамическом режиме получать IP-адреса.

Как работает в IPv4?



DHCP-сервер

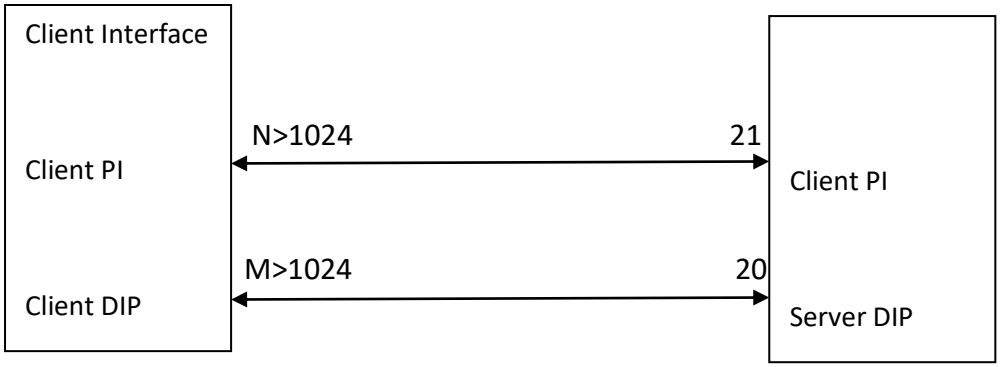
1) В ручную настраиваются статические адреса.
2) Автоматическое назначение IP-адресов(есть некий PULL IP, из которго раздаем IP. Как только выдаем IP-адрес, то записываем пару IP=MAC и после этого этот IP-адрес никому не даем.)
3) Автоматическое распределение динамических адресов.(есть PULL IP. Выделяется на определенное «время аренды».*Нет никакой связки IP=MAC*)

Как работает в IPv6?

FTP(File Transfer Protocol)

Работает поверх TCP.

Client



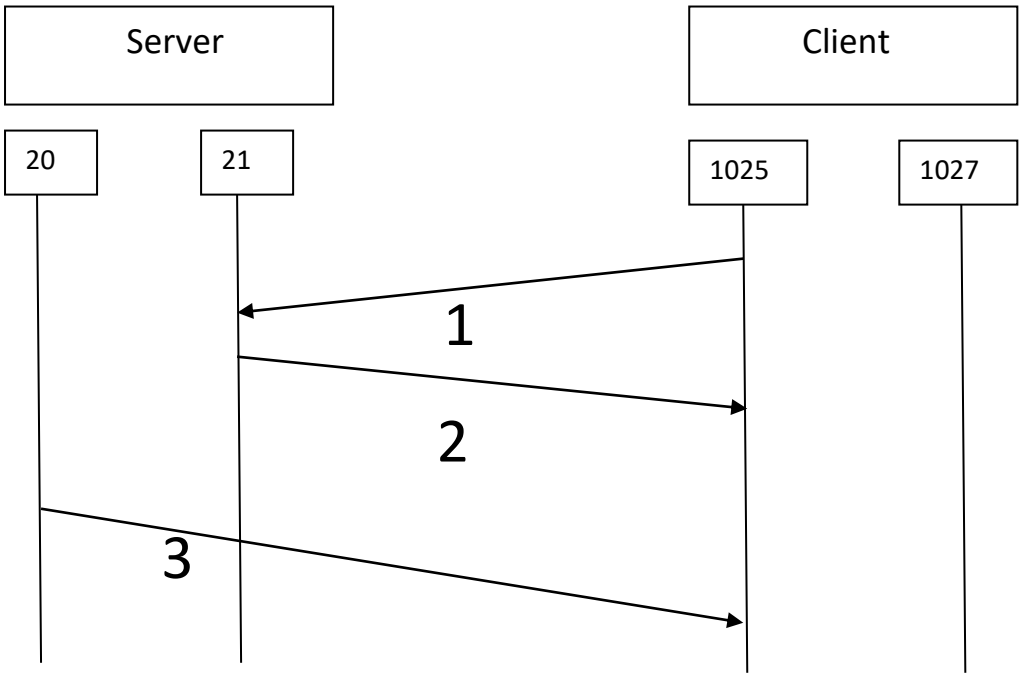
Client Interface- виртуальный терминал, взаимодействует с Client PI
PI-интерпретатор команд
Client/Server- модули, которые обеспечивают обмен данными.
DTP-Data Transfer Process

FTP использует 2 TCP-соединения.
Одно TCP-соединение служит для передачи команд. 21-на стороне Server. На стороне Client высокий порт >1024.Это TCP-соединение работает в течение всего сеанса.

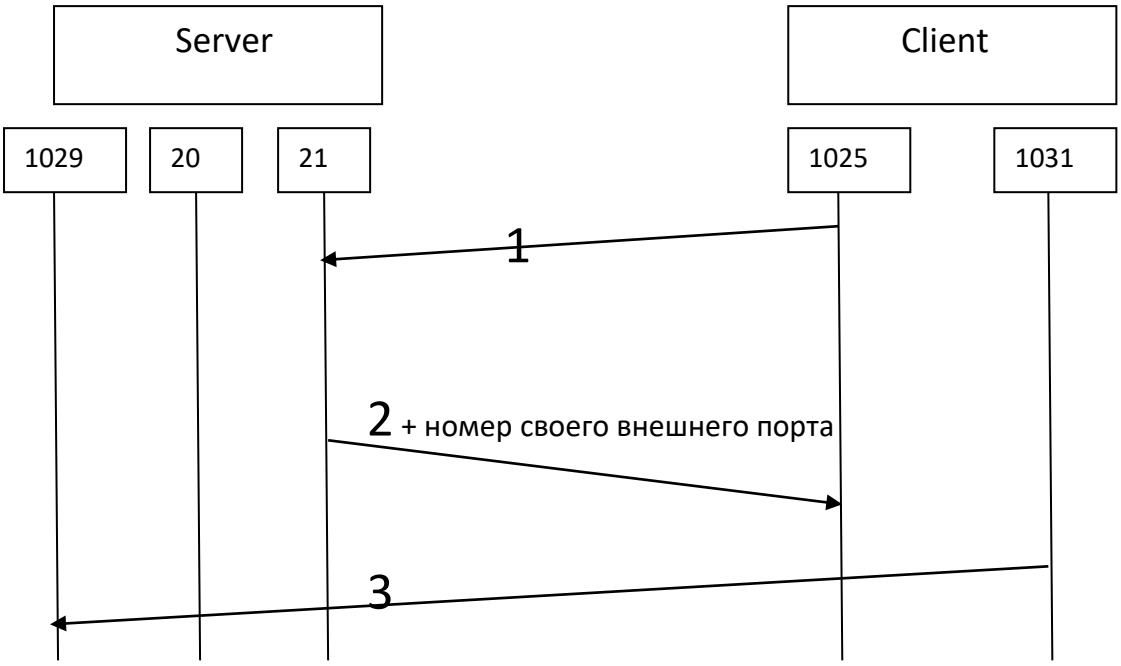
Второе TCP-соединение служит для передачи данных.Это соединение открыто только на одну транзакцию(на передачу одного файла).На Server или 20 или высокий порт L>1024.На Client высокий порт M>1024.

Активный и пассивный режимы.

1) Active



2)Passive



При передаче следующего файла заново выделяется номер порта.

1.порт 21 на сервере FTP всегда открыт пассивно.

2. если клиент хочет отправить/ получить данные, то он устанавливает TCP-соединение по каналу передачи команд. Затем при помощи FTP команд отправляет параметры передачи данных между клиентом и сервером. А именно: роль участника соединения(активная/пассивная), порт передаваемых данных, структуру данных, управляющие директивы.

3. После передачи всех параметров та сторона, которая является активной устанавливает соединение TCP.

4. После окончания передачи данных TCP-соединение передачи данных закрывается, а TCP- соединение передачи команд остается установленным для передачи других команд.

5. Закрыть это соединение можно только явно соответствующей командой от клиента.

Лекция 14

Команды FTP

1) Команды управления доступом к системе.
   -Если хотим доступ к системе: ->login->password
   -Смена активной директории:->Cwd (change work directory)
   -Реинцелизация(команда очищает все переменные пользователя и сбрасывает параметры соединения. Если в данный момент происходит передача данных, то она завершается с прежними параметрами):->REIN
   -Разрыв TCP-соединения: ->QUIT

2) Команды управления протоколом данных
   Устанавливают параметры передачи данных->PORT h1,h2,h3,h4,p1,p2
   где h1,h2,h3,h4-ip address p1,p2-port number

   например: порт 2041=1111 1111001 это 7 249 значит порт 2041 запишется как 7,249
   -перевод в пассивный режим ->PASV h1,h2,h3,h4,p1,p2
   Ответом будет IP интерфейса сервера и номер порта сервера.
   -Определяет тип передачи данных и может принимать тип ASCII, EBCDIC,IMAGE и др. ->TYPE
   -способ передачи (по умолчанию потоком) ->STPU
   То, что передаём, описание типа записи:->MODE

3) Команды FTP-сервиса
   -указать серверному модулю FTP передать клиенту указанный файл:->RETR filename
   -Указать серверному модулю FTP принять файл и сохранить в активной директории под указанным именем
   -Переименование файла(файл на сервере, но пишем на клиенте):->RNFR filename(старое имя) ->RNTO filename(новое имя)
   -Удаление файла:->DELE filename
   -создать директорию:->MKD dirname
   -Удалить директорию,но при условии,что она пустая :- >RMD dirname
   -Список файлов активной директории:->LIST
   -Прерывает выполнение предыдущей команды и закрывает канал передачи данных: ->ABOR

Протокол SMTP(Simple Mail Transfer Protocol)

Протокол прикладного уровня для передачи почты.Работает поверх TCP по 25 порту.
Он только *отправляет* почту. Для корректной работы протокола необходмо, чтобы и отправитель и получатель были подключены к сети.
Алгоритм работы:

1) установить TCP-соединение
2) посылается команда ->mail from: zzz@bmstu.ru
3) получение отклика 250 ОК
   (порт говорит, что произошло:
   2_ _ -все хорошо
   5_ _-не успешно

3_ _-не окончено)

4) посылается команда ->RCPT TO: yyy@bmstu.ru (получатель)
5) получение отклика -> 250 OK
6) ->DATA
7) enter
8) *your text or something else*
9) enter
10) . enter (вместе с 9: конец тела сообщения)
11) получение отклика -> 250 OK
12) ->QUIT закрывает передачу всего сообщения

Протокол POP3(Post Office Protocol)

Протокол прикладного уровня,работает по 110 порту.
Не имеет кодовых 250 и т.п.
Есть +OK
-ERROR

1)Автоизация
->USER
->PASS
->QUIT(переходи в 3)
2)Получение почты
->STAT(количество сообщений и их объем)
->LIST(если ввели номер сообщения,то выводится информация о данных сообщения)
->RETR(передает содержание(объем) сообщения)
->DELE(помечает указанное сообщение на удаление,но не удаляет его)
->RESET(если какое-то сообщение было помечено,как удаленное,то эти пометки снимаются)
3)Завершение
В этом режиме сервер удаляет все помеченное. Закрывается TCP-соединение.

Протокол IMAP

Протокол прикладного уровня,работает поверх TCP на 143 порту.
Предназначен для получения почты, для корпоративных клиентов, хранит письма на сервере(если сообщение было изменено, то другие пользователи узнают об этом)
Поиск писем на сервере. Возможность создавать папки. Есть INBOX,куда попадают письма по умолчанию.

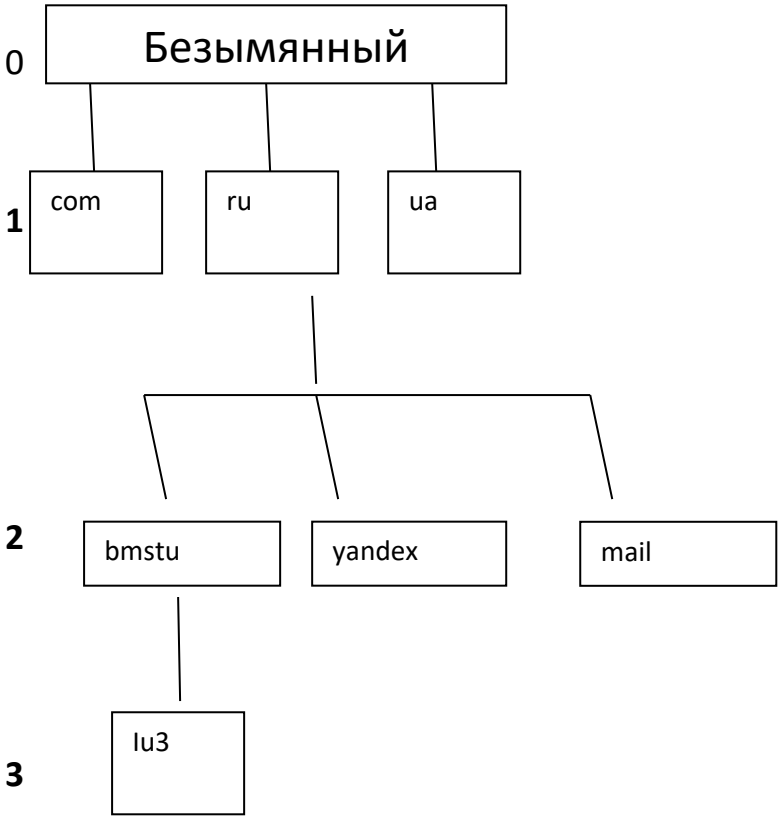
Лекция 15.

<u>Доменная структура Internet</u>

URL- Uniform Resource Locator(универсальный идентификатор) «один домен-одна страна.»

-имя записывается с нижнего уровня

-безымянный с 0 уровня не пишется.



- Max 255(включая точки)
- max число символов на домен 64 символа
- ICANN(Internet Corporation Assigned Name and Number)

  <u>DNS(Domain Name System)</u>

  Система доменных имен.

Компьютерная разделенная система для получения информации о доменах. Обычно используется для получения неизвестного IP-адреса по известному доменному имени узла Internet.
Распределенная база данных DNS поддерживается с помощью иерархии DNS-серверов. На нулевом уровне работают 13, так называемых, корневых DNS-серверов, которые в обязательном порядке содержат информацию обо всех IP-адресах DNS-серверов 1ого уровня. На первом уровне должно функционировать как минимум 2 DNS-сервера(первичный и вторичный), которые в обязательном порядке содержат информацию обо всех IP-адресах DNS-сервера второго уровня и т д.

В каждом домене существует организация, в которой делегированы ICANN права по регистрированию домена.
DNS-протокол работает поверх UDP на 53 порту.

Ключевые характеристики DNS.

1) Распределенность администрирования
   Ответственность за разные части иерархической структуры несут разные люди/организации.
2) Распределенность хранения информации.
   Каждый узел сети в обязательном порядке должен хранить только те данные, которые входят в зону его ответственности и возможно адреса ключевых DNS-сервефров.
3) Кэширование информации
   Узел может хранить некоторое количество данных не из своей зона ответственности для уменьшения нагрузки на сеть.
4) Иерархическая структура
   В ней все узлы объеденены в дерево и каждый узел может или самостоятельно определять работу нижестоящих узлов или передавать их узлам.
5) Резервирование
   За хранение и обслуживание своих узлов отвечают обычно несколько серверов разделенных как физически,так и логически, что обеспечивает сохранность данных и продолжения работы даже в случае сбоя одного из узлов.

Терминология и принципы работы.

1) Домен(Domain)
   Узел в дереве имен вместе со всеми подчиненными ему узлами(если таковые имеются), т е именованная ветвь или поддерево в дереве имен.
   Структура доменного имени отражает порядок следования узлов в иерархии. Доменное имя читается слева направо от младших доменов к доменам высшего уровня.
2) Под домен (Subdomain)
   Подчиненный домен. Теоретически такое деление может достигать длины 127 уровней. На практике не превышает 4-5.
3) Ресурсная запись
   Единица хранения и передачи информации в DNS. Каждая ресурсная запись имеет имя(т е привязка к определенному доменному имени), тип и поле данные, формат и содержание, которое зависит от типа.
4) Зона
   Часть дерева доменных имен(включая ресурсные записи) размещаемое как единое целое на некотором сервере доменных имен, а чаще одновременно на нескольких серверах. Целью выделения части дерева в отдельную зону является передача ответственности за соответствующий домен другому типу или организации.
5) Делегирование
   Операция передачи ответственности за часть дерева доменных имен другому лицу или организации.
6) DNS-сервер
   Специализированное ПО для обслуживания DNS а также компьютер, на котором это ПО выполняется.
7) DNS-клиент
   Специализированная библиотека/программа для работы с DNS. В ряде случаев DNS-сервер выступает в роли DNS-клиента.
8) Авторитетность
   Признак размещения зоны на DNS-севере. Ответы DNS-сервера могут быть двух типов.
   -авторитетные(когда сервер заявляет, что сам отвечает за зону)

-не авторитетная (когда сервер обрабатывает запрос и возвращает ответ других серверов.)
Система DNSсодержит иерархию DNS-серверов, соответствующую иерархии зон. Каждая зона поддерживается как минимум одним авторитетным сервером DNS.

Рекурсия.

(в DNS) обозначает алгоритм поведения DNS-сервера, при котором сервер выполняет от имени клиента полный поиск нужной информации во всей системе DNS, при необходимости обращаясь к другим DNS серверам.

DNS-запрос может быть рекурсивным, требующим только поиска, и не рекурсивным, не умеющим выполнять полный поиск.

При ответе на не рекурсивный запрос, также при запрете выполнять рекурсивные запросы DNS-сервер либо возвращает данные о зоне, за которые он отвечает, либо возвращает адреса серверов, которые обладают большим объемом информации о запрошенной зоне.
В случае рекурсивного запроса DNS-сервер опрашивает другие DNS-сервера(в порядке убывания уровня зон в имени) пока не найдет ответ или не обнаружит, что домен не существует.
При рекурсивной обработке запроса все ответы проходят через DNS-сервер и он получает возможность кэшировать их(временное хранение). Повторный запрос на эти имена в этом случае не идет дальше КЭШа сервера. В КЭШ существует TTL.

Записи в DNS.

(Recourse Records)

Записи DNS-единица хранения и передачи в DNS.

Каждая ресурсная запись состоит из:

1) NAME(имя)

Доменное имя, к которому привязано или к которому принадлежит данная ресурсная запись.

2) TTL(время жизни)
   Допустимое время хранения данной ресурсной записи в КЭШе неответственного DNS-сервера.
3) TYPE(тип)
   Определяет формат и назначение данной ресурсной записи.
4) CLASS(класс)
   Теоретически считается, что DNS может использоваться не только в TCP/IP сетях. Это поле определяет тип используемых протоколов.
5) LDLEN(длина поля данные)
   В байтах.
6) RDATA(данные)
   Формат и содержание зависят от типа записи.
   Наиболее важные типы DNS-записи.

1) A address record(IPv4)
   Запись адреса хоста и его IP-адреса.
2) AAAA address record(IPv6)
3) CNAME canonical name record
   Каноническая запись имени(псевдоним). Используется для перенаправления на другое имя.
4) MX mail exchange
   (Почтовый обменник)
5) NS name server
   Указывает на DNS-сервер данного домена.

Лекция(семинар) 16.

HTTP

Hyper Text Transfer Protocol.
-протокол прикладного уровня,обеспечивает высокопроизвоительный процесс тирражирования информации в интернет.

Этот протокол позволяет получать доступ к ресурсам и сервисам web-серверов. www(world wire web)

Для унификации доступа к многофункциональеым ресурасм сети web-сереров поддерживают комплекс интерфейсов, позволяющих структурировать уровни и методы работы с различными ресурсами сети.

Каждый из интерфейсов представляет собой объект сети со своими методами и своей структурой.

Индентификация ресурсов сети.
Осуществляется с помощью :
-URI
-URN
-URL
Uniform Rresource Identifier
Uniform Rresource Name
Uniform Rresource Locator
Разные имена одного и того же сервиса, который предназначен для идентификации типов,методов работы и компьютеров,на которых находятся определенные ресурсы,доступные через Internet.

Этот сервис сотоит из *трех частей*:
1)схема
идентифицирует тип сервиса,через который можно получить доступ к сервису(FTP,WEB)
2)Адрес
идентифицирует адрес(узел/host) ресурса.(www://,http://,https://,ftp://)
3)Имя(путь доступа)
Идентифицирует полный путь к ресурсу на выбранном host'е, который мы хотим использовать для доступа к ресурсу.

http://www.bmstu.ru/home/image/ping1.jpg
URL ресурса может содержать не только имя,но и параметры,необходимые для его работы.Параметры отделяются «?» и строка параметров состоит из лексем, которые разделяются «&».

Принципы постороения HTTP-соединения.
ПРотокол HTTP-соеднинения предельно прост.
Устанавливается TCP-соединение порт 80 по инициативе клинента.клиент же отправляет HTTP-запрос. Сервер отвечает HTTP-ответом и TCP-соединение разрывается.
В более сложном случае принимают участие несколько промежуточных объектов.

1)proxy

Представляет собой промежуточный объект,который принмает запросы клиента и в зависимости от своих настроек изменяет часть или все сообщение запроса и передает переформатированный запрос далее по цепочке. В момент принятия запроса proxy может работать как сервер и сам отвечать на запррос. При передаче запроса, он работает как клиент.

Proxy фактически является «главными воротами» входа пользователя в INTERNET.

2)Gateway(шлюз)

ПРедставляет собой промежуточный сервер и работает прозрачно для клиента(т е запросы ретранслируются). Шлюз просматривает входящий трафик и является “главными воротами” входа пользователя Internet во внутреннюю сеть.

Proxy+gateway=firewall(межсетевой экран)- используют КЭШирование.

3)tunnel

Представляет собой программу-посредник между двумя соединениями. Используется,когда необходимо организовать поток данных через какой-либо промежуточный объект,который не сможет интерпретировать структуру потока данных.

Каждый из объектов участников соединения за исключением tunnel может поддерживать внутренний КЭШ запросов и ответов.

КЭШ представляет собой локальную БД сообщений-ответов и систему управления этой базой. КЭШ хранит ответы серверов и возврщает их по запросц клиента, не передавая запрос следующему объекту цепочки соединения.Эффект использования КЭШ в том,что он уменьшает длину цепочки соединения и соответственно время соединения.

Опписание протокола HTTP.

Запросы клиента и ответы сервера используют стандарт MIME&

Параметры и методы запроса.

METHOD<SP>request-VRL<SP>HTTP-Version

[General header](может использоваться в запросе/ответе. Только тогда, когда передается тело сообщения)

Date: (время постоения сообщения)

Pragma: (устанавливает специальные диррективы.)

[Request-header] (в запросах клиента)

Authorization: (для аутентификации. Ответы не КЭШируются)

From: (интернет-адрес пользователя)

If-Modified-Since: (используется при работе методом GET. Если запрашиваемый ресурс не изменился с момента указанного в этом параметре,данный ресурс не возвращается полностью,а возвращается только заголовок.)

Referer: (содержит URL ресурса)

User-agent: (информация о ПО клиента)

[Entity-Header] (заголовок передаваемого сообщения. Он содержит информацию о структуре и формате тела запроса, а сели тело отсуттвует, то информацию о запрашиваемом ресурсе.Использкется как в запросе6 так и в отсете.)

Allow: (содержит список методов,поддерживаемых ресурсом)

Content-Encoding: (содержит идентификатор типа дополнительной кодировки ресурса)

Content-length: (длина тела сообщения в байтах)
Content-Type: (содержит тип ресурса и таблицу кодировки)
Expiress: (содержит дату окончания срока действия ресурса)
Last-Modified: (содержит дату и время последнего измения ресурса)
/*line*/
[Entity-Body]

METHOD
-GET(запрашивает информацию о ресурсе,расположенному по заданному URL)
-POST(для передачи клиентом на сервер данных,которые должны быть обработаны ресурсом,указанным в URL или в теле запроса:Entity-Body)
Ответы на запросы методом POST не КЭШируются.
-HEAD(аналогично GET, но возвращает только заголовок)
-DELETE(для удаления ресурса,определяет URL)
-PUT(используется,когда клиент желает сохранить передаваемый на сервер ресурс с опрделенным URL)
Request URL- URLресурса.он может быть представлен как в абсолютном,так и в относительном формате.
Абсолютный-полный URL
Относительный-путь к ресурсу на текущем сервере HTTP.
HTTP-version-версия HTTP.

Если запрос содержит тело,а не один заголовок, то оно располагается после пустой строки,которая отделяет заголовок от тела запроса.
Формат и длина тела должны удовлетворять стандарту MIME.

Структура ответа.
HTTP-Version<SP>STATUSCODE<SP>Reason-Phrase
[General-Header]
[Response-Header]
Location: (содержит полный URL ресурса,который отвечает на отпрвленный запрос. Если ресурс перемещен на другой сервер, то запрашиваемый сервер отвечает Location с URL нового ресурса.И запрос клиента автоматически перенаправляется по URL указанным сервером.)
Server: (содержит спецификацию ПО web-сервра,отвечающего на запрос)
www-authenticle: (одержат параметры аутентификации)
[Entity-Header]
/*line*/
[Entity-Body]

STATUSCODE(трехзначный числовой код.ПО клиента)
Reason-Phrase(короткая строка описания статуса,которая предназначена для текстового анализа результата обработки запроса)