

15. Мосты: назначение, алгоритмы работы, основные типы, рекомендации по использованию.

Мост (bridge) и его быстродействующий функциональный аналог - коммутатор (switch), делит общую среду передачи данных на логические сегменты. Логический сегмент образуется путем объединения нескольких физических сегментов с помощью одного или нескольких концентраторов. Каждый логический сегмент подключается к отдельному порту моста.

Мосты решают следующие задачи:

- Увеличивают размер сети.
- Увеличивают максимальное количество компьютеров в сети.
- Устраняют узкие места, появляющиеся в результате подключения избыточного числа компьютеров и возрастания трафика.
- Соединяют разнородные физические носители, такие, как витая пара и коаксиал.
- Соединяют разнородные сегменты сети, например Ethernet и Token Ring, и переносят между ними пакеты.

Мосты оперируют на канальном уровне OSI, от них не требуется проверки информации высших уровней -> они могут быстро продвигать трафик, представляющий любой протокол сетевого уровня. Мосты допускают использование в сети всех протоколов, не отличая при этом один протокол от другого.

Канальный уровень имеет два подуровня: Управления логической связью и Управления доступом к среде.

Мосты работают на подуровне Управления доступом к среде.

Мост можно запрограммировать так, чтобы он не пропускал все фреймы, посылаемые из определенной сети.

Мосты действуют как непреодолимая преграда для некоторых потенциально опасных для сети неисправностей.

Мосты — эффективное средство для расширения и сегментирования сети, поэтому они часто используются в больших сетях (отдаленные сегменты в таких сетях соединены телефонными линиями). Две сети, расположенные на значительном расстоянии друг от друга, можно объединить в одну.

С этой целью используют два удаленных моста, которые подключают через синхронные модемы к выделенной телефонной линии.

Разница между мостом и коммутатором:

мост в каждый момент времени может осуществлять передачу кадров только между одной парой портов, а коммутатор одновременно поддерживает потоки данных между всеми своими портами.

Алгоритм работы моста

Мост подключается к сети как любой узел.

Но порт моста не имеет MAC адреса, и мост обрабатывает все пакеты, поступающие на его порт.

- 1) Мост принимает любой пакет, имеющийся на входе порта и помещает его в буферную память.
- 2) Мост проверяет адрес назначения пакета и сравнивает его с адресами узлов, закрепленных за портами.
- 3) Если адресат находится в том же сегменте(на том же порту), что и отправитель, то буфер очищается.
- 4) Если адресат находится в другом сегменте(на др. порту), то мост на общих основаниях, как и любой узел, осуществляет процедуру доступа к передающей среде и передает кадр в этой среде.

Мост разрезает сеть на два домена коллизии. Коллизии правой и левой части независимы друг от друга, и это позволяет строить сети любого размера.

Для определения мостом принадлежности узлов к сегментам используют 2 метода:

- 1) Ручное формирование таблицы адресов
- 2) Формирование таблиц адресов происходит в режиме самообучения.

Определение принадлежности узлов сегментам возможно за счет наличия в кадре не только адреса назначения, но и адреса источника пакета. Используя адрес источника, мост устанавливает соответствие между номерами своих портов и адресами узлов.

В процессе изучения сети мост просто передает кадры, работая некоторое время репитером. После составления таблицы, он начинает передавать кадры между портами только в случае межсегментной передачи. Если после завершения обучения на входе моста появится кадр с неизвестным адресом назначения, то этот кадр будет повторен на всех портах. Если мост знает где узел -адресат, он передает пакет ему. Если адресат неизвестен, мост транслирует пакет во все сегменты.

Благодаря таблице маршрутизации мост способен сегментировать график.

Такой механизм порождает проблему петель (между двумя сегментами есть более чем один путь). Поэтому в сетях, где есть мосты, запрещено создавать топологии с кольцами (петлями). Для автоматического распознавания петель в конфигурации сети разработан алгоритм покрывающего дерева (Spanning Tree Algorithm, STA).

Типы мостов:

- 1) Прозрачные мосты.

Оба порта работают по одной сетевой технологии. Это мосты MAC подуровня.

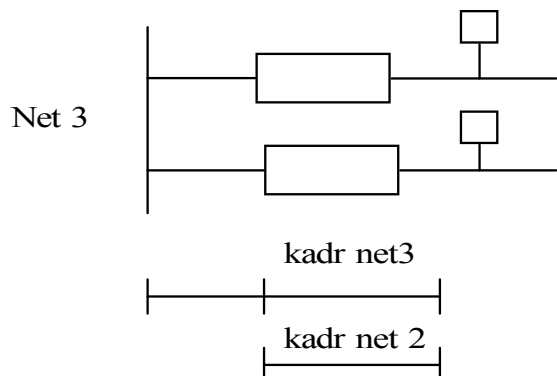
2) Транслирующие мосты.

Порты поддерживают разные сетевые технологии. Мосты работают на LLC подуровне, потому что извлекают данные из

кадра одной сети и записывают в кадры второй сети.

Такая трансляция никогда не бывает безупречной, т.к. всегда имеется вероятность, что одна сеть поддержит определенный фрейм, который не поддерживается другой сетью.

3. Инкапсулирующие мосты.



Передачи пакетов через сеть 3 идет без трансляции.

Также можно разделить мосты на локальные и дистанционные.

Дистанционное мостовое соединение представляет ряд уникальных трудностей объединения сетей. Одна из них - разница между скоростями LAN и WAN. Скорости LAN на порядок выше скоростей WAN.

Дистанционные мосты не могут увеличить скорость WAN, но они могут компенсировать этот баг путем использования достаточных буферных мощностей. Но нормальная работа таких мостов осуществима только для коротких пакетов информации, которые не переполняют буферные мощности моста.

Админы часто юзают мосты, потому что они:

- просты в установке и незаметны юзерам;
- обладают высокой гибкостью и адаптируемостью
- относительно дешевы

Если для расширения сети принято решение использовать мосты, необходимо учитывать следующие факты:

- мосты обладают всеми возможностями репитеров;
- соединяют два сегмента и восстанавливают сигналы на уровне пакетов;
- функционируют на Канальном уровне модели OSI;
- не подходят для распределенных сетей со скоростями передачи менее 56 Кбит /с
- не могут одновременно использовать несколько маршрутов
- пропускают все широковещательные сообщения, допуская перегрузку сети
- считывают адреса источника и получателя каждого пакета
- пропускают пакеты с неизвестным адресом получателя

Основное назначение мостов:

- соединить два сегмента для увеличения длины сети или количества узлов в ней
- уменьшить трафик за счет сегментации сети
- соединить разнородные сети