7_Объединение сетей

Необходимость в объединении сетей связана с ограничениями протяженности сети и числа узлов ней, а также с необходимостью снижения нагрузки на сетей.

Совокупность соединенных сетей - объединённая сеть.

Требуются специальные устрйства: мосты и маршрузиаторы.

Объединение сетей с помощью мостов:

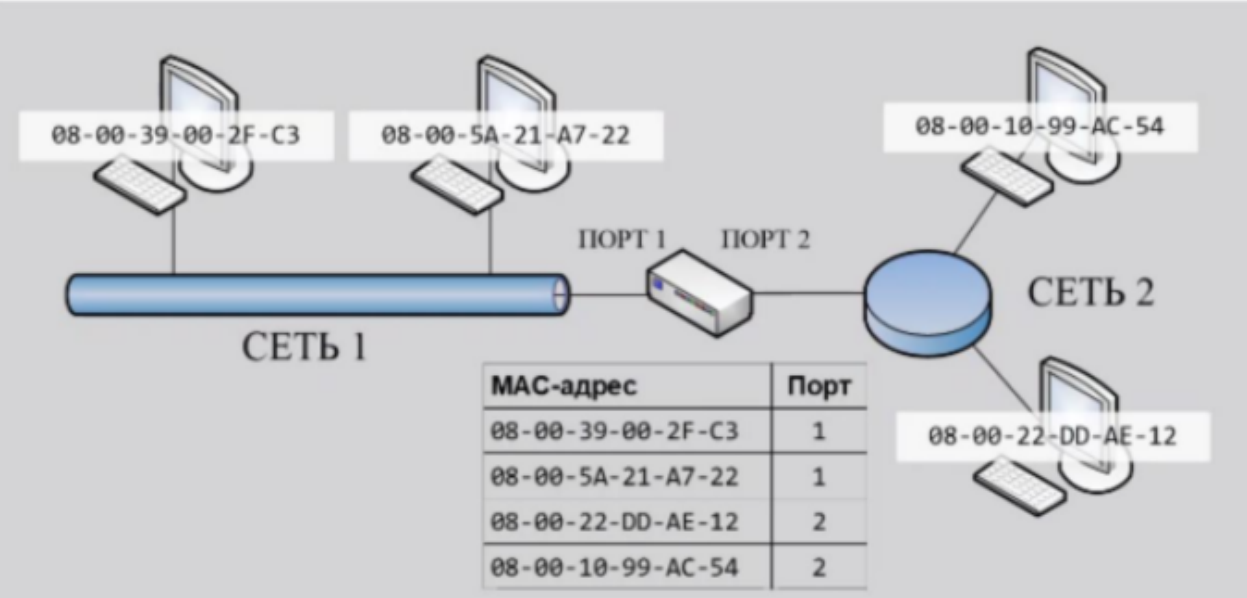
Мост может объединять сети на канальном уровне, разных сетевых технологий.

Существует несколько технологий объдения с помощью мостов.

- В сетях Ethernet, в основном применяется прозрачное мостовое соединение (transparent bridging).
- В сетях Token Ring, применяется мостовое соединение с маршрутизацией от источника (source-route bridging).
- Для объединения сетей, использующих различные сетевые технологии (обычно Ethernet и Token Ring) - применяется трансляционное мостовое соединение (translational bridging)

Прозрачное соединение мостов

Мостовое соединение прозрачное - анализирует проходящие через него кадры и изучает состав сети. Для каждого из них сеть представляется таблицей MAC адресов связанных с определенным портом этого моста.



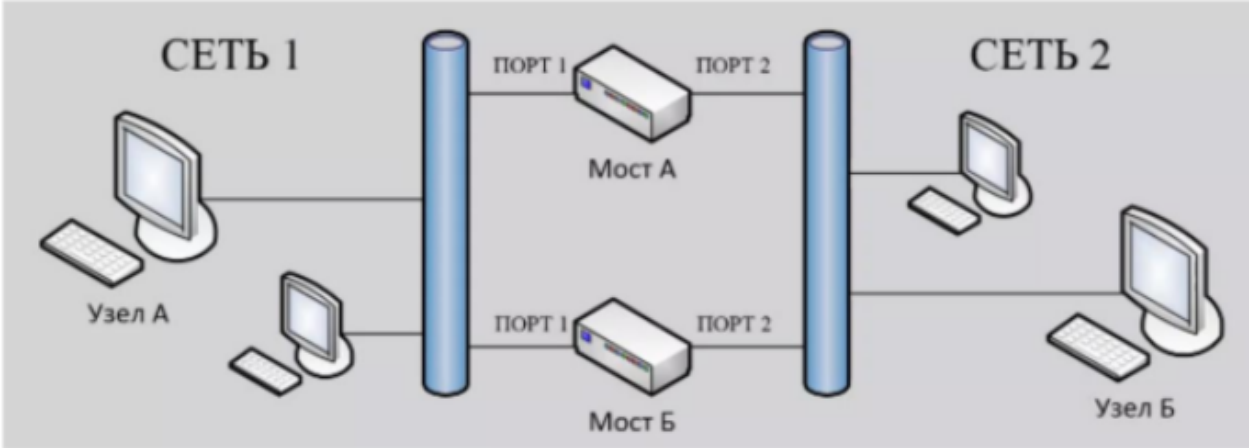
MAC-адрес	Порт
08-00-39-00-2F-C3	1
08-00-5A-21-A7-22	1
08-00-22-DD-AE-12	2
08-00-10-99-AC-54	2

Изначально мост ничего не знает о том, какой подсети принадлежат узлы. Затем постепенно мост заполняет свою таблицу. Записывая в нее адрес отправителя кадр и порт от которого кадр был получен.

Каждый раз ищет порт назначения в своей таблице. Если адрес находится то кадр передается через указанный порт, иначе во все активные порты, кром етого с которого кадр был получен.

☝ Петля - состояние сети, когда происходит бесконечная пересылка кадров между мостами подключенными в общую подсеть.

Пример для объединенной сети:



1. *Узел А отправляет кадр узлу Б. Мост А получает кадр и заносит в свою таблицу MAC-адресов связь с портом 1, Если моста А пока не знает порт узла Б, но выполняет рассылку.*
2. *Мост Б также получает кадр от А и выполняет те же действия. В итоге узел Б получает две копии от моста А и Б. , от узла А.*

Одновременно от с этим мост Б получает копию кадра от моста А и выполнит следующее действие : произведет изменение запись в таблице чтобы связать узел А с портом 2. И рассылает по всем активным портам. Аналогичная ситуация происходит когда мост А получает…

*При ответе от узла Б **оба моста проигнорируют** эти ответы. Т.к. будут считать, что их получатель находится в той же сети , что и отправитель.*

*Кроме того, если в объединенной сети есть петли и на какой либо узел отправили кадр с широковещательным адресом, то это приведет к ситуации **широковещательного шторма - когда всем узла сети будут бесконечно рассылаться копии кадра.***

> Однако наличие избыточных связей, которые и образуют петлю, может быть полезно для повышения надежности сетей, за счет образования резервных путей.

❓ **Проблема : постоянное обновление таблицы MAC-адресов.**

☝️ **Решение: протокол связующего дерева**

Протокол связующего дерева

Это протокол канального уровня, предназначенный для устранения петель в сетях Ethernet связанных с помощью мостов.

Объединяет определяет роли мостов и портов а затем блокирует часть мостов чтобы получилась топология дерево которая не будет иметь петель.

Чтобы определить какие порты блокировать мосты периодически обмениваются протокольными блоками данными (**BPDU** - сообщения протокольных блоков данных моста) используя при этом групповой MAC - адрес.

💡 **BPDU** - сообщения протокольных блоков данных моста

Выбор корневого моста

Для того, чтобы получить топологию дерево - в сети выбирается корневой мост. (это происходит не вручную).

Каждому корневому мосту назначается уникальный идентификатор (ID) - 8байтное число. Где младшие 6 байт содержат MAC-адрес моста, а 2 старших - приоритет моста.

Корневым выбирается мост с наименьшим ID. Можно влиять на выбор корневого моста, изменяя приоритет в ID моста. Т.к. находится в старших разрядах ID моста, его значение подавляет значение MAC-адреса.

Если всем мостам назначить одинаковый приоритет, будет выбран мост с наименьшим значением MAC-адреса.

После включения каждый мост считает себя корневым, но только до тех пор пока не выяснится обратное. Поэтому каждый мост рассылает сообщения, где в качестве ID корневого моста свой ID. Каждый мост сравнивает полученное значение корневого со значением хранящимся в

памяти, если оно меньше, то сохраняется в памяти в качестве ID-корневого моста.

Затем мост рассылает снова BPDU, но уже с новым ID корневого моста. В конце концов все мосты определяться, кто из них корневой мост.

Пример: дата и время рождения. Кто младше.

Корневой порт

Для каждого моста, который не является корневым, определяется корневой порт. Это порт, с которым порт будет обмениваться с корневым мост.

Корневым портом выбирается тот, у которого наименьшая суммарная стоимость пути к корневому мосту.

Когда получает сообщение, мост увеличивает стоимость порта. Если найдется несколько портов с минимальной стоимостью. Корневым выбирается порт с наименьшим идентификатором.

ID порта - 2байтное - старший - приоритет, младший - номер порта.

Выбор назначенных мостов

Для каждой подсети выбирается мост, который будет передавать кадры между узлами данной подсети и корневым мостом. Мост может быть назначен нескольким подсетям, но для подсети может быть назначен только один мост.

Назначенным может быть мост, который непосредственно подключен к подсети не через корневой порт. Среди подходящих мостов, назначенным выбирается мост имеющий наименьшую стоимость пути к корневому мосту.

Если таких мостов несколько, то назначенным с наименьшим ID.

Последний этап: определение **назначенных** портов (?уточнить)

Для всех мостов определяется назначенный порт. Через этот порт мост обмен данными с подключеннными узлами подсети. У корневого моста как правило все порты являются назначенными.

Если мост подключен к подсети с помощью нескольких портов, назначенным выбирается порт с наименьшим ID.

Все порты, которые не являются корневыми или назначенными блокируются.

Первые три поля - содержат нулевые значения.

Флаги - используются только 2 старших бита. TCN - уведомление об изменение топологии, TCA - подтверждение изменения топологии.

Уведомление об изменении - изменение состояния порта, разрыв моста и что-то еще? Тогда перестраивается дерево.

ID корневого моста - 8 байт Стоимость пути до корня.

ID - моста - 8 байт. ID-порта - 4 байта. Из них используется только 2.

Возраст сообщения - указывает время, прошедшее с тех пор, как корневой мост отправил сообщение, ставшее причиной этого сообщения .

Макс возраст - как долго мост может блокировать порты.

Время приветствия - указывает временной интервал как часто BPDU посылаются корневым мостом.

Задержка перехода - указывает время до начала изменения топологии?

Объединение сетей с помощью маршрутизатора

Посредством передачи пакетов из одной сети в другую.

При передаче пакетов маршрутизатор решает задачу выбора маршрута - маршрутизация.

Выполняется на основании таблицы маршрутизации . Она у этого устройства.

Таблица - это база данных, расположенная в памяти маршрутизатора.

Каждая запись таблицы маршрутизации содержит адрес сети или узла, адрес слдеующего маршрутизатора,пункты назначния, метрика задающая предпочтительность маршрута.

Когда маршрутизатор получает пакет, он извлекает адрес назначения из заголовка пакета. А затем используя этот адрес, просматривает таблицу маршрутизации в поисках нужной записи.

При наличии нескольких подходящих записей к.п. выбирается запись с наименьшим значением метрики. Если поиск не дал результата полученный пакет отбрасывается.

Таблица может заполняться вручную администратором сети или автоматически с помощью протокола маршрутизации. (первая - статическая, вторая - динамическая).

У первой важное преимущество - это контроль. В отношении маршрутов администратор может определяет точную конфигурацию, которая не будет подвергаться изменениям. Недостатки очевидны в крупной сети. Это трудоемкость процесса ввода записей для каждого маршрутизатора, а еще потому что при каждом изменении модификации сети требуется вносить изменения.

Для второй - все наоборот. Минус - остуствтие контроля. Плюс - не требуется при изменении сети что-то постоянно менять вручную.

Алгоритмы маршрутизации

Первый дистанционно векторный

Каждый маршрутизатор периодически рассылает своим соседним маршрутизаторам вектор расстояни(дистанции) до известных ему сетей.

> ☝️ Рассылает только соседним(!)

Информация обо всех известных ему сетям. Указывает расстояние в количестве переходов (прыжков), то что необходимо пройти до цели назначения. Получив такой вектор маршрутизатор обновляет соотвтетствующие записи в своей таблице маршрутизации. При этом маршрутизатор получает информацию о том , от кого получит вектор.

В итоге каждый маршрутизатор будет влядеть информацией обо всех сетяхи расстояних до них.

Если записи не обровляются долго, то их удалить из таблицы

По каналу

Поддерживает базу данных состояний канала. В ней содержится информация обо всех маршрутизаторах обьединой сети.на сонов этой базе маршрутизатор строит дерево кратчайших путей

/*в блокноте*/

Т.о. у всех маршрутизаторов будет идентичная база состояний.

Маршрутизатор посылает сообщение, получив которое маршрутизаторы станут обновлять свои таблицы. Называется LSA.

Сходимость маршрутов - процесс раскрытия новой маршрутной ситуации(?)

В идеале сеть в состоянии конвергенции к

/**/

Для предотвращения петель в протоколе RIP

Разделение горизонта - нецелесообразно отправлять информацию о маршруте с в том же направлении с которого она была получена.

Негативный отклик заключается в рассылке уведомлений о недоступности того или иного маршрута.

RIPv1 - классовая адресация, RIPv2 - безклассовая адресация. Обе версии используют транспортный протокол UDP и порт с номером 520.

Одно сообщение RIP может содержать до 25 записей. При этом максимальный размер сообщения не превышает 512 байт.

Для передачи сообщений сообщений адрес: RIPv1 255.255.255.255

RIPv2 224.0.0.9

Формат кадра:

Команда - 1/2 запрос/ответ

Версия - содержит версию протокола

Зарезервировано

Запись 1(20 байт)

...Запись N(20 байт)

Формат записи в сообщении RIPv1:

ID семейства адресов - определяет протокол сетевого уровня, используемый для адресации. Для IPv4 это 2.

Резерв

Сетевой адрес - 4 байта - адрес сети или узла назначения.

Резерв

Метрика - определяет число переходов , количество маршрутизаторов через которые пакет может пройти по маршруту. В запросах данное поле не используется. Маршруты сетей непосредственно подключенных к маршрутизатором имеют метрику 0, недоступный - 16.

Небольшой диапазон делает этот протокол непригодным для больших сетей.

При векторном подходе сходимость достаточно медленная(?)

Формат для RIPv2:

ID семейства адресов - ||-

Признак маршрута - для отличия внутренние маршруты от маршрутов внешних (от внешних шлюзевых протоколов - не нулевое. Например номер автономной системы)

Сетевой адрес

Маска подсети - маску подсети для поля сетевой адрес. Если узел то маска 255.255.255.255 .

Следующий маршутизатор - адрес следующего маршрутизатора на пути к пункту назначения. Если поле содержит адрес 0000, то следующий маршрутизатор - отправитель сообщения.

Метрика

При необходимости можно использовать первую и только первую запись сообщения для аунтефикации, тогда на маршрутные данные остается только 24 записи. Формат первой записи:

FFFF (все 15 единиц)

Тип аутентификации - использование нешифрованного пароля. Значение 2. (?)

Информация аунтефикации - содержит пароль. Если короче 16, то остальные байты заполняются 0.

Нешифрованное - для увеличения скорости передачи данных. Пароль лучше хэширование. Для внутренний сети если она безопасна.

NG RIP для IPv6

Поле версия - 1. Транспортный протокол UDP, порт 521. Групповой адрес FF02::9.

Формат сообщения тот же .

Первое поле содержит сетевой адрес размером 16 байт. Здесь адрес сети или узла назначения.

Признак маршрута то же, Префикс - содержит префикс связанный с адресом сети. Метрика короче.

/Запись назначенного маршрута/

Если нужно указать какой маршрутизатор следующий использовать запись назначенного маршрута (сюда записать адрес). Все записи для которых адресом является вышеуказанный группируются после записи этого маршрута. Другие записи если они есть указываются поле записи назначенного маршрута. Последний байт записи FF. Если 128 нулей то следующий это отправитель.

Протокол "По самому короткому пути"

Сетевой. На алгоритме "По состоянию канала".

OSPF - внутренний шлюзовой протокол. **А RIP был прикладным.**

Используются в крупных сетях, поскольку ограничений как в протоколе RIP у него нет. В небольших сетях не применяется из-за сложности настройки маршрутизации.

OSPFv2 - Idля IPv4

OSPFv3 - для IPv6

Протокол позволяет объединять смежные сети в области. При этом внутренние маршрутизаторы не будут знать ничего о топологии за пределами совей области. Это позволяет снизить передаваемый объем служебной информации. При использовании областей допущение об идентичности бд состояний канала перестает быть верным.

Каждый маршрутизатор имеет бд состояний канала только для своей области. Маршрутизаторы связывающие несколько областей граничные области.

> 💡 Магистраль - специальная область, с которой соединены все остальные области. Отвечает за распространие маршрутной информации между остальными областями.

Чтобы уменьшить число сообщений LSA передаваемых между маршрутизаторами, применяется принцип назначенного маршрутизора. На каждую область предполагается один назначенный. В зависимости от его приоритета от 0 до 255. Где большие числа являются приемлемыми. Втором по значимости является резервным.

Основная функция назначенного маршрутизатора состоит в рассылке LSA сообщений между всеми маршрутизаторами области. Резервный назначенный маршрутизатор обеспечит быстрое восстановление назначенного маршрутизатора в случае если тот выйдет из строя.

OSPF работает на базе IP, для идентификации определяется по номеру 89.

Все пакеты используют однотипные заголовки.

Версия - 2,3

Тип - 1 байт - определеяет назначение пакета

1. Приветственный пакет - для определения соседних маршрутизаторов
2. Описание баз данных - распроснет информации о состоянии канала
3. Запрос состония канала - для запросов информации о базе днны
4. Обновление состояния канала- рспростния LSA

5. Подвтержени состояния канала - для приема LSA

Длина пакета - в байта + длина заголовra

Идентификатор маршрутизоатора - отправитель

Id области -

Контрольная сумма - 2 байта. Результат для всего пакета начиная с заголовка, но без учета поля аунтефикации.

Тип аунтефикации - 0 не используется 1 - простая парольная 2 - криптографическая аунтефикация. Не шифрование пароля

Аунтефикации - 8 байт. Содержит данные аутефикации.

В OSPF3 были удалены аунтефикации. Вместо них используются средства IPv6

Все поля имеют тот же смысл что и верхнее. Но! Во первых короче, а во -вторых

Идентификатор экземпляра - для управления взаимодействием маршрутизаторов одной области. Маршрутизаторы могут стать соседями, если могут иметь одинаковый ID экземпляра.

Пакеты передаются с использованием групповым адресом (1 строка) между всеми.

Для передачи назначенным маршрузиторам - заканчивается на 6.

Внешний шлюзовый протокол

Обмен информацией о доступности между граничными маршрутизаторами автономной системы.

Оба протокола - прикладные стека TCP/IP.

ИКС и сети 07.11.docx

https://vk.com/doc119489364_575605661?hash=e398803dcdaabed0a6&dl=1dba025af4aa254ffa