

12. Model catalog on OMNET++ – community cite. Available at: <http://omnetpp.org/index.php/models/catalog> (accessed 01.03.2015).

Received 01.12.2014

УДК 004.312.24

АЛГОРИТМ БАЛАНСИРОВКИ ТРАФИКА В КОЛЬЦЕВЫХ СЕТЯХ ETHERNET

Лихтциндер Б.Я., Рыжих С.В.

Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ

E-mail: rrsereg@mail.ru

В статье рассматриваются существующие методы защиты от «петель» в сетях Ethernet на уровне коммутации, приведено их описание на основе алгоритма балансировки трафика.

Ключевые слова: MSTP, RSTP, MSTP, BPDU, главный коммутатор, основной и резервный порты, кольцевые топологии.

Введение

Существует целый ряд способов регулирования загрузки сетей передачи данных начиная с кабельных узлов, обеспечивающих физическое подключение рабочих мест пользователей, и заканчивая ядром системы [1]. Способ регулировки зависит от класса коммутатора. Сети передачи данных принято рассматривать на трех иерархических уровнях: доступа, распределения и ядра системы. Коммутаторы также можно классифицировать в соответствии с уровнями модели OSI, на которых они передают, фильтруют и коммутируют кадры.

Различают коммутаторы уровня 2: Layer 2 (L2) Switch и коммутаторы уровня 3: Layer 3 (L3) Switch. Коммутаторы уровня L3 используются в основном для задач уровня агрегации и уровня ядра. Коммутаторы уровня L2 анализируют входящие кадры, принимают решение об их дальнейшей передаче и передают их пунктам назначения на основе MAC-адресов канального уровня модели OSI. Данные коммутаторы используются в основном для задач иерархии доступа. Так как данный уровень обеспечивает доступ конечных пользователей, он самый масштабный из вышеперечисленных.

Каждый уровень характеризуется своими наборами протоколов, алгоритмов и правил для передачи информации. Так, при работе с пакетами в L3 есть возможность использования двух и более путей достижения получателя пакета, что позволяет балансировать трафик, а при использовании механизмов по проверке задержки, потерь и джиттера еще и выбирать наилучший путь. Но это только при работе с IP пакетами. При работе с фреймами L2 данный подход невозможен, так как наличие двух и более путей вызовет закольцовывание трафика и, как следствие, «шторм».

Метод выделения связующего дерева

Для предотвращения закольцовывания трафика на сетях Ethernet используются алгоритмы защиты от петель. Логику их работы можно описать как «метод выделения связующего дерева» [2]. Указанный метод основан на последовательном исключении из цепи определенных звеньев. Для передачи трафика всегда используется только один путь достижения узла получателя. Следовательно, необходим механизм выбора канала для передачи с учетом его параметров. Одним из указанных параметров является перегрузка канала связи.

Особенность коммутаторов Ethernet состоит в том, что для временного хранения фреймов на каждом порту коммутатора имеется внутренняя буферная память. Если канал связи свободен, то фрейм сразу же передается по назначению. Если такой возможности нет, то коммутатор помещает фрейм на временное хранение в буферную память, образуя очередь. В случае же переполнения очереди фреймы удаляются в соответствии с используемым алгоритмом обслуживания очередей.

Распределение трафика достигается с помощью протоколов построения связующих деревьев и защиты от кольцевания STP, RSTP и MSTP. Наиболее распространенными являются сети кольцевой топологии, использующие построение множественных деревьев по протоколу MSTP[3].

При стандартной работе протоколов STP, RSTP, MSTP связующие деревья в каждом из портов коммутаторов на уровне L2 создаются в начале работы и изменяются только в случае изменения топологии или выхода из строя одного из элементов. Связующие деревья строятся на основании определения главного коммутатора и вычисления стоимости пути до него. Эти параметры передаются между коммутаторами с помощью служебных фреймов через определенные интервалы времени. Главные коммутаторы выбира-

ются на основании их приоритетов и могут быть различными в разных связующих деревьях. Стоимости путей вычисляются на основании полосы пропускания физических портов коммутаторов, через которые эти пути проходят. Путь через порт коммутатора с наименьшей стоимостью выбирается как основной, а остальные пути – как резервные.

Недостатком такого способа оценки стоимости пути является то, что способ учитывает лишь номинальные пропускные способности портов и не учитывает их реальную загрузку трафиком. В реальности один из портов может быть перегружен местным трафиком, и его реальная пропускная способность, в отношении транзитного трафика, проходящего по связующему дереву, может значительно отличаться от номинальной. В результате в коммутаторах могут возникнуть большие очереди и задержки трафика.

Под перегрузкой порта коммутатора понимаем наличие хотя бы одного фрейма в очереди данного порта. В неперегруженном состоянии очереди в портах отсутствуют. Наличие очередей проверяется с помощью стандартного механизма, заложенного в коммутаторе. Данный функционал используется на всех современных коммутаторах.

С целью устранения возможных перегрузок предлагаем учитывать реальную загрузку каждого из портов коммутаторов. И при перегрузке на уровне L2 весь (или часть) трафик, проходящий через перегруженный порт коммутатора, адаптивно переключать на резервный путь.

Алгоритм балансировки трафика

Реализация алгоритма предусматривает следующие действия. На рис. 1 представлена обычная кольцевая схема соединения коммутаторов.

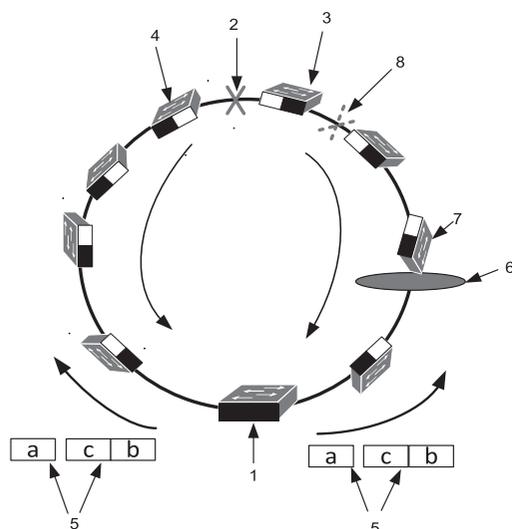


Рис. 1. Кольцевая схема сети Ethernet

В качестве главного выбираем коммутатор 1. Протокол STP обеспечивает логический разрыв кольца 2. Разрыв производится коммутатором 3 и фиксируется в его памяти. Соседний с ним коммутатор 4, находящийся по другую сторону разрыва, подобной информации не имеет. Наличие логического разрыва разделяет кольцо на две ветви, в которых информация в сторону главного коммутатора -1 перемещается по направлению и против направления часовой стрелки соответственно (стрелки вверх). Порты коммутаторов, направленные в сторону главного коммутатора, считаются активными (зачернены), а другие порты кольца – резервными. В обычном режиме информация о стоимости путей передается по кольцу в полях «Стоимость пути» (поле «а») служебных фреймов BPDU 5, перемещающихся от главного коммутатора по кольцу в противоположных направлениях (стрелки вниз) [4]. Логический разрыв не препятствует прохождению BPDU по кольцу. Каждый из них совершает полный цикл и изымается главным коммутатором 1.

Алгоритм учитывает возникающие перегрузки в выходных портах коммутаторов и предусматривает возможность уменьшения перегрузок за счет последовательного переноса точки 2 логического разрыва кольца в направлении перегруженного коммутатора. Так, например, при возникновении перегрузки 6 на выходном порту коммутатора 7 разрыв переместится в положение 8. При этом трафик коммутатора 3 будет передаваться в сторону главного коммутатора по резервному пути (против часовой стрелки), что приведет к некоторой разгрузке коммутатора 7. Если перегрузка не ликвидирована, а на левой ветви не возникло дополнительной перегрузки, то в следующем цикле прохождения BPDU точка разрыва 8 переместится еще на один шаг, и т.д. При исчезновении перегрузки точка разрыва будет перемещаться в противоположную сторону.

Реализация процесса переноса точки разрыва при перегрузках может осуществляться различными путями. Рассмотрим реализацию, основанную на использовании свободных разрядов некоторых полей BPDU.

Для информирования других коммутаторов о перегрузке, может, например, использоваться поле «Идентификатор порта в логическом дереве» [5] длиной 1 байт, в котором в настоящее время используется только 4 бита, и значение оставшихся 4 битов всегда равно 0. Реализация предлагаемого способа задействует старший разряд из этого поля. (Разряд «с» для информирования наличия перегрузки на пути достижения

главного коммутатора.) Для определения коммутатора 4, ближайшего к точке разрыва, используется поле «b» (последние 4 бита поля «Идентификатор порта в логическом дереве»). Указанное поле используется в стандартной реализации протокола MSTP для определения типа порта соседнего коммутатора.

В начальный момент, при выходе из главного коммутатора, разряды «с» в обоих BPDU 5 устанавливаются в нулевое состояние. При обнаружении перегрузки в активном порту коммутатор 7 переводит разряд «с» в единичное состояние и передает далее во все коммутаторы (против часовой стрелки) служебный фрейм с измененным разрядом «с». Другой BPDU, перемещающийся в обратном направлении, поступит в коммутатор 7 через пассивный порт и поэтому перегрузку не зафиксирует. Разряды «b», при выходе обоих BPDU из главного коммутатора, устанавливаются в состояние, определяющее тип порта, через который будет послан данный BPDU соседнему коммутатору. Так, на выходе из главного коммутатора состояние разряда «b» будет определять основной тип порта. При прохождении BPDU (против часовой стрелки) через активный порт коммутатора 3, который ранее произвел виртуальный разрыв 2, его разряд «b» устанавливается в состояние, определяющее заблокированный тип порта, которое передается и фиксируется соседним коммутатором 4.

После этого на коммутаторе 4 разряд «b» BPDU устанавливается в состояние, определяющее активный тип порта, и пересылается соседнему коммутатору. Разряд «b» другого BPDU (движущегося по часовой стрелке) поступает в граничный коммутатор 3 через пассивный порт и определяется как основной тип порта. Таким образом, после окончания цикла передачи только у граничных коммутаторов 3 и 4 копии разрядов «b» и «с» BPDU, перемещающихся против часовой стрелки, находятся в состоянии блокировки и перегрузки порта. Во всех остальных коммутаторах копии разрядов «b» имеют другие значения. Копии разрядов «b» и «с» BPDU, перемещающихся по часовой стрелке, также будут иметь другие значения.

Одновременное наличие блокировки и перегрузки порта сформирует единичный сигнал, который будет записан в старший разряд «a» поля «Стоимость пути» этого BPDU. Для приемлемой сходимости в каждом связующем дереве используется не более 20 коммутаторов. Поэтому существующее поле «Стоимость пути» из 4 байтов избыточно для такого количества коммутаторов

и старшие разряды указанного поля могут быть использованы для реализации алгоритма.

Изменение старшего разряда поля «Стоимость пути» гарантированно вызовет смену пути до главного коммутатора от коммутатора 3 и, следовательно, переместит точку разрыва кольца из положения 2 в положение 8. Такой способ изменения стоимости пути позволяет использовать механизмы, заложенные в стандартной реализации протокола MSTP.

При поступлении каждого очередного служебного фрейма BPDU, в случае прекращения перегрузки, переключение коммутаторов на резервный канал будет производиться поочередно начиная с коммутатора, ближайшего к точке разрыва кольца.

Коммутационный BPDU

В стандартной реализации изменение топологии выполняется путем полного удаления прежних таблиц коммутации, что приводит к длительному процессу перестройки связующего дерева [5]. В предлагаемом алгоритме переключение каналов на граничном коммутаторе -3 производится путем замены выходных интерфейсов в существующих таблицах коммутации по MAC-адресам. Для информирования других коммутаторов предлагается ввести новый тип BPDU – коммутационный.

На рис.2 представлена реализация коммутационного BPDU. Изменение состояния порта вследствие алгоритма балансировки трафика вызывает формирование коммутационного BPDU. Коммутационный BPDU содержит MAC-адреса 9, добавленные в таблицу коммутации MAC-адресов 8 через порты, не изменившие состояния пересылки трафика. Данный BPDU 10 будет послан через порт 2, перешедший в состояние «основной» из состояния «блокированный». Для оставшихся же MAC-адресов (MAC-адресов, доступных через порты, изменившие состояние пересылки трафика) в MAC-таблице данного логического коммутатора 3 будет заменен порт доступности на порт 2, изменивший состояние «блокированный» на состояние «основной». Таким образом, часть трафика от коммутатора 3 будет направлена в обход точки перегрузки.

Логические коммутаторы, получившие коммутационный BPDU 11, заменяют порты доступности MAC-адресов 12 содержащихся в коммутационном BPDU, на порт получения данного BPDU в своих MAC-таблицах 13. После последовательного прохождения всех коммутаторов данный BPDU будет удален создавшим его ком-

мутатором. Это существенно снизит объем служебного трафика, необходимого на пересоздание таблиц коммутации.

Если вследствие переключения трафика перегрузка образуется на «резервном» пути, то коммутатор, зафиксировавший перегрузку, будет действовать по описанному алгоритму в обратном направлении. Это приведет к возврату на «основной» путь и не вызовет потери трафика, передаваемого через другие каналы. В дальнейшем при исчезновении перегрузки (получении BPDU с нулевым значением бита «с») будет происходить последовательное (каждым циклом BPDU) обратное переключение трафика на «основные» каналы. Коммутатор, участвовавший в переключении трафика на «резервный» канал, следуя описанному алгоритму, получив бит «с» с нулевым значением, будет последовательно удалять значения старшего бита в поле «Стоимость пути» начиная с наибольшего номера логического кольца.

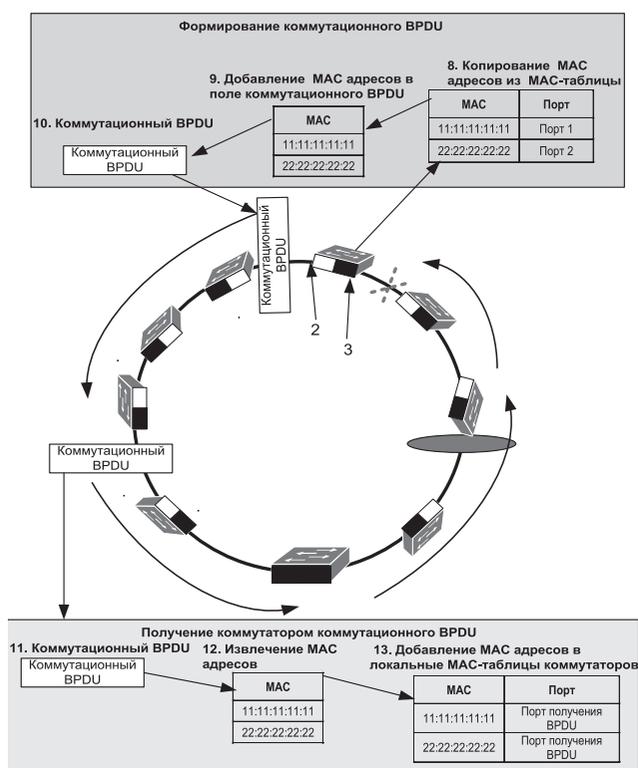


Рис.2. Реализация коммутационного BPDU

Таким образом, изменение старшего разряда поля «Стоимость пути» гарантированно вызовет смену пути до главного коммутатора от коммутатора 3 и, следовательно, переместит точку разрыва кольца в положение 2. После этого, согласно алгоритму, будет сфор-

мирован коммутационный BPDU и отправлен через порт, изменивший состояние «блокированный» на «основной». Таким образом, будет восстановлено первоначальное распределение трафика логических колец. За один цикл BPDU каждый физический коммутатор переключит только одно логическое кольцо. Следовательно, за один цикл BPDU в сети передачи данных будет переключено не более двух логических деревьев. (Первое логическое кольцо – при смещении точки разрыва части логических колец до точки перегрузки и второе – при смещении точки разрыва следующего логического кольца по направлению к точке перегрузки.)

Таким образом, после исчезновения перегрузки через несколько циклов BPDU все логические кольца будут переведены в первоначальный режим работы, обеспечив целостность системы.

Выводы

Разработанный алгоритм предназначен повысить эффективность использования сетей передачи данных и улучшить показатели обработки трафика, такие как величина задержки и потеря пакетов.

Данный алгоритм не вносит изменений в протоколы связующих деревьев, а лишь использует доступные механизмы данных протоколов. Это позволяет применить данный алгоритм в существующих сетях передачи данных, в которых оборудование, не поддерживающее данный алгоритм, будет выступать в роли транзитных участков.

Литература

1. Афанасьев Э. Cisco QoS для начинающих. URL: http://network.xsp.ru/3_11.php (20.05.2014).
2. Лапухов П. Введение в MSTP. URL: <http://blog.ine.com/2010/02/22/understanding-mstp/> (05.05.2014).
3. Малоизвестный MST. URL: <http://habra-habr.ru/post/128172/> (05.05.2014).
4. IEEE Std 802.1aq™-2012. Vol. 4, p. 145.
5. Компания Cisco. Документ № 24248. URL: <http://www.cisco.com/c/en/us/support/docs/lan-switching/spanning-tree-protocol/24248-147.html> (05.05.2014).

Получено 27.11.2014

Лихтциндер Борис Яковлевич, д.т.н., профессор Кафедры мультисервисных сетей и информационной безопасности (МСИБ) Поволжского государственного университета телекоммуникаций и информатики (ПГУТИ). Тел. (8-846)333-47-69. E-mail: lixt@samtel.ru

Рыжих Сергей Вячеславович, аспирант Кафедры МСИБ ПГУТИ. Тел. 8-903-30-44-666. E-mail: rrserg@mail.ru

ALGORITHM FOR TRAFFIC BALANCING IN ETHERNET RING NETWORKS

Lihtzinder B.Y., Ryzhikh S.V.

For today the majority data networks are working on technology Ethernet, and used in a ring topology, thereby allowing optimal network reliability. Most of the equipment of such networks is access switches having from 8 to 48 ports for connecting subscribers. Such switches usually only work with frames, that is level 2 model OSI. But such a construction, in addition to simplicity, reliability and low cost, has its drawbacks, such as the possibility of Ethernet storms that can bring the entire network. The process of storms underlies at technology of Ethernet, namely in the absence of the removal mechanism of broadcast frames from the exclusive network. Consequently, this technology does not allow the existence of more than one communication switching channel between the two points exchanging information. For protection from storms special protocols to block redundant switching channels between the communications equipment are used. There are quite a lot of such protocols today, such as STP, PVSTP, RSTP, but the most common is MSTP. But they all have the same function, based on the finding of redundant channels of communication between the two switches. In finding happens, block all but one, selected on the basis of comparison of parameters based on the coefficients dependent on the bandwidth of the channel. Thereby, there is prohibition of the transmission of traffic on all channels except the selected one. Based on the basic principle these protocols impose restrictions on the use of all possible communication channels. This approach results in an inefficient allocation of all available bandwidth in the network. To solve this problem, we developed an algorithm that gives an opportunity to use all available network band-width, if it needs, without affecting to the quality of service for existing subscribers. Our algorithm uses the builtin mechanisms in existing protocols, thereby it allows at low changes improving the efficiency of the network to be almost doubled.

Keywords: *MSTP, RSTP, BPDUs, root switch, root port and block port, network, ring topology.*

Likhtzinder Boris Yakovlevich, Doctor of Technical Science, Professor of Department of Multiservice Networks and Information Security, Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation. Tel. +7 846 333 47-69. E-mail: lixt@samtel.ru

Ryzhikh Sergei Vyacheslavovich, Postgraduate Student of Department of Multiservice Networks and Information Security, Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation. Tel. +7 903 30 44 666. E-mail: rrserg@mail.ru

References

1. Afanas'ev Ye. *Cisco QoS dlya nachinayushchih* [Cisco QoS, entry-level]. Available at: http://network.xsp.ru/3_11.php (accessed 20.05.2014).
2. Lapuhov P. *Understanding MSTP*. Available at: <http://blog.ine.com/2010/02/22/understanding-mstp/> (accessed 05.05.2014).
3. Stycenko A. *Maloizvestnyj MST. Multi-region implementation considerations* Available at: <http://habrahabr.ru/post/128172/> (accessed 05.05.2014).
4. IEEE Std 802.1aq™, 2012, vol. 4, p. 145.
5. Understanding Multiple Spanning Tree Protocol (802.1s). Cisco Systems, Inc. Available at: <http://www.cisco.com/c/en/us/support/docs/lan-switching/spanning-tree-protocol/24248-147.html> (accessed 05.05.2014).

Received 27.11.2014