

УДК 004.724: 004.728

М. К. Б о й ч е н к о, И. П. И в а н о в

МОНИТОРИНГ РЕСУРСОВ УЗЛОВ КОРПОРАТИВНОЙ СЕТИ

Рассмотрены ресурсы корпоративных сетей, определена возможность их мониторинга при функционировании, исследована целесообразность и эффективность централизованного мониторинга ресурсов транзитных и оконечных узлов сетей.

E-mail: ivanov@bmstu.ru

Ключевые слова: сеть, буфер, процессор, нагрузка, трафик, пропускная способность, ресурс.

Наиболее дефицитными ресурсами в корпоративных сетях принято считать [1, 2] пропускную способность сегментов сети, буферное пространство коммуникационных и оконечных узлов, время центрального процессора этих узлов.

Для каждого сегмента корпоративной сети существует максимальная пропускная способность, соответствующая пропускной способности интерфейсов. Так, в коммутируемых сетях технологии Ethernet это может быть скорость 10; 100; 1000; 10000 Мбит/с, в беспроводных сетях стандарта IEEE 802.11 таковыми являются скорости 11 и 54 Мбит/с. Однако в сегментах ядра корпоративной сети или в сегментах уровня распределения такая пропускная способность делится между случайными в данный интервал времени числами источников трафика. Число хостов в корпоративных сетях колеблется от нескольких десятков до нескольких десятков тысяч. При поддержке алгоритмов построения покрывающих деревьев Multiple Spanning Tree Protocol (MSTP — протокол множественных связующих деревьев) [3] в одной корпоративной сети может существовать до 4096 виртуальных сетей, топологии которых могут перекрываться на одних и тех же физических сегментах уровня магистральной (ядра) или распределительного уровня (возможно перекрытие и на уровне доступа, если один хост входит в несколько виртуальных сетей). Поэтому заранее предсказать необходимые (максимально возможные) пропускные способности физических сегментов невозможно. Кроме того, ориентироваться на максимальные возможности существующих технологий передачи информации при проектировании физической топологии корпоративной сети слишком дорого. Невозможность предсказать потребную пропускную способность предопределяет закладывание некоторой избыточности при

выборе технологий межинтерфейсного обмена информацией. Реальное значение прошедшей через интерфейс информации может быть измерено только в течение заданного интервала времени, т.е. является усредненным на некотором интервале значением, не дающим информации о всплесках нагрузки на интерфейс внутри этого интервала наблюдения. Косвенно о всплеске интенсивности трафика можно судить по доле потери пакетов (хотя, следует отметить, что потери могут быть и при искажении кадров). Измерение скорости передачи информации возможно в различных системах управления (например, агентами Simple Network Management Protocol (SNMP) — протокола управления значениями параметров узла в сети [1]) и операционных системах конечных и транзитных узлов сети (например, Internetwork Operating System (IOS) — межсетевая операционная система коммутаторов Cisco Catalyst). К сожалению, даже при очень малых интервалах наблюдения, позволяющих установить истинные значения скорости передачи информации через интерфейсы и, следовательно, процент использования пропускной способности сегмента (его загрузку), нельзя предсказать поведение трафика, так как в следующий момент его загрузка может упасть до нуля либо достичь своего предельного значения.

Буферное пространство любого узла сети (так же, как и ее физические сегменты) является разделяемым ресурсом, предоставляемым информационным потокам. Любой пакет, прибывающий в узел сети (и ему предназначенный), сначала помещается в аппаратный буфер сетевого адаптера (либо в аппаратный буфер коммутатора или маршрутизатора). Затем пакет копируется либо в буфер оперативной памяти конечного узла сети (хоста), либо в буфер оперативной памяти маршрутизатора, либо в буфер выходного порта коммутатора через его коммутационную матрицу и т.д. Если буферное пространство, в которое необходимо передать пакет, занято, то либо происходит задержка (образование очереди обслуживания), либо пакет выбрасывается (что допустимо для очень малого числа приложений). Для обеспечения хорошего качества обслуживания буферное пространство приходится резервировать. Политика резервирования зависит от принятой политики обслуживания очередей в данном узле сети. Можно, например, все прибывающие на конкретный порт коммутатора кадры помещать в одну очередь на передачу в порты назначения. Можно организовать в оперативной памяти порта коммутатора несколько очередей в зависимости от принятых в сети приоритетов (стандарт IEEE 802.1Q предусматривает восемь уровней приоритетов для кадров виртуальных локальных сетей). В любом случае может быть организован мониторинг буферного пространства, при котором можно четко охарактеризовать загрузку буферного пространства в данный момент времени на

любом интерфейсе сетевых узлов для любой из очередей, организованной любым устройством в сети на любом уровне и этапе обработки информации. Мониторинг буферного пространства широко используется при обработке информации. Например, метод Weighted random early detection (WRED — взвешенное случайное раннее обнаружение) [3] по степени загрузки буфера интерфейса путем удаления случайно отмеченных пакетов позволяет снизить вероятность возникновения перегрузки на рассматриваемом интерфейсе. Измерение размеров буферов проводится алгоритмами обслуживания очередей в вычислительных системах. Справедливая организация очередей Fair Queuing (FQ), справедливая организация очередей с побитовым циклом Bit-Round Fair Queuing (BRFQ), взвешенная справедливая организация очередей Weighted Fair Queuing (WFQ) и другие [4] базируются на измерении реальных или виртуальных размеров очередей относительно соответствующих ресурсов вычислительных систем.

Время центрального процессора, выделяемого каждому потоку в коммутаторах и маршрутизаторах, во многом определяет реальную пропускную способность транспортного узла. Если среднее время центрального процессора, выделяемое рассматриваемому потоку информации, обозначить T_p , а среднеквадратическое отклонение этой характеристики σ_p , то из теории очередей [4] следует, что среднее число запросов, ожидающих обслуживания, составит

$$\omega = \frac{1}{2} \frac{\rho^2}{1 - \rho} \left(1 + \frac{\sigma_p^2}{T_p^2} \right), \quad (1)$$

где $\rho = T_p/T_k$ — загрузка процессора; T_k — среднее время между поступающими в процессор заявками на обслуживание (в случае коммутатора или маршрутизатора $T_k = 1/\lambda$ — время между моментами поступления заявок на обслуживание пакетов; λ — средняя интенсивность трафика, измеряемая числом пакетов в секунду).

Выражение (1) справедливо для пуассоновского трафика [4]. Если к тому же время обслуживания распределено по показательному закону, так как для простоты анализа допускается экспоненциальное распределение длин прибывающих для коммутации или маршрутизации кадров, а время T_p линейно зависит от длины кадра, то $\sigma_p = T_p$ и $\omega = \rho^2(1 - \rho)$. Среднее число кадров, находящихся в очереди на обслуживание, вместе со средним числом обслуживаемых кадров (равным ρ) составляет:

$$r = \rho + \omega. \quad (2)$$

Следовательно, для пуассоновского входного потока $r = \rho/(1 - \rho)$. Среднее время ожидания пакетом обслуживания (т.е. нахождения его

в очереди на обслуживание) можно записать как

$$T_{\omega} = T_p \frac{\rho}{1 - \rho} \frac{1}{2} \left(1 + \frac{\sigma_p^2}{T_p^2} \right), \quad (3)$$

а среднее время прохождения пакета через процессор — следующим образом:

$$T_r = T_p + T_{\omega}. \quad (4)$$

Для пуассоновского входного потока $T_{\omega} = T_p \frac{\rho}{1 - \rho}$ и $T_r = \frac{T_p}{1 - \rho}$. Выражения (1)–(4) справедливы для дисциплины обслуживания First In First Out (FIFO — первым пришел, первым обслуживаешься). В современных процессорах реализуется подход разделения процессора Processor Sharing (PS) в случае отсутствия приоритетов или схема обобщенного разделения процессора Generalized Processor Sharing (GPS). При этом организуется несколько очередей на обслуживание с равными или различными долями выделения времени центрального процессора [4]. Анализ подобных схем в аналитическом виде в общем случае провести невозможно. Поэтому приходится прибегать к имитационному моделированию. Так или иначе состояния центрального процессора и длин очередей поступающих на обработку пакетов данных взаимосвязаны и могут быть доступны для мониторинга в современных транзитных узлах корпоративной сети.

Мониторинг ресурсов узлов корпоративной сети принципиально возможен, однако для решения задачи оперативного управления интенсивностью трафика централизованное использование результатов мониторинга встречает серьезные трудности. Например, объем получаемой информации (интенсивность передачи информации, объем занятого буферного пространства, доля времени центрального процессора) по каждому интерфейсу в сети не только очень велик сам по себе, но и требует значительного времени для его передачи по самой сети в центр управления. При этом дополнительно используются те же ресурсы сети. Выработка управляющих воздействий по результатам мониторинга представляет собой задачу, для решения которой нужны значительные вычислительные мощности, и, наконец, передача управляющих команд к интерфейсам узлов сети, в свою очередь, требует ресурсных затрат все той же сети, кроме того, к моменту получения управляющего воздействия от центра управления ситуация на каждом интерфейсе с высокой вероятностью меняется и поступающая команда может оказаться бесполезной. Иными словами, централизованное управление интенсивностью трафика приводит к неоправданным накладным затратам, отличается низкой оперативностью, т.е. не эффективно. Поэтому для управления интенсивностью трафика более приемлемы концепция распределенного менеджмента, при которой мо-

ниторинг осуществляется лишь на используемом маршруте данного потока информации, и различные механизмы регулирования. Так, в стеке протоколов TCP/IP на транспортном уровне при использовании протокола Transmission Control Protocol (TCP — протокол управления передачей) осуществляется мониторинг времени прохождения в оба конца (Round-Trip Time, RTT), который не дает подробной информации о состоянии ресурсов на всех узлах маршрута в сети, однако по его значению можно судить о состоянии ресурсов и управлять размерами окон приема, передачи, перегрузки и порога, а также таймерами TCP-сущностей при использовании технологии скользящего окна [2]. Время RTT измеряется на самом источнике информации при прибытии пакета, подтверждающего правильный прием переданного кадра, что минимизирует накладные затраты (особенно при “попутной перевозке” квитанций на переданные пакеты). Явным недостатком метода скользящего окна в TCP является его достаточно высокая инерционность. Если кадр, по какой-либо причине теряется в начале маршрута от источника к приемнику, то в лучшем случае источник установит этот факт только по истечении времени, равного RTT. Учитывая, что срабатывание таймера повторной передачи Retransmission Time Out (RTO) происходит чуть позднее, приходится констатировать факт возможной бесполезной нагрузки сети информацией от источника в течение достаточно длительного периода времени (так как впоследствии, вероятнее всего, придется прибегать к повторной передаче), что задерживает ликвидацию заторов на маршруте, если именно они явились причиной отсутствия подтверждения.

При использовании User Datagram Protocol (UDP — протокол пользовательских дейтаграмм) нет явных средств мониторинга ресурсов на пути следования пакетов. Для приложений реального времени, которые применяют протокол Real Time Protocol (RTP — протокол передачи трафика реального времени), работающий поверх UDP, мониторинг ресурсов маршрута осуществляется нумерацией пакетов RTP, указанием идентификатора источника синхронизации вместе с отметкой времени. Это позволяет определить расхождение в интенсивностях передачи и приема пакетов, установить факт пропажи пакета и т.п. Для поддержания обратной связи между источником и приемником информации протокол RTP всегда используется в паре с протоколом Real-Time Transport Control Protocol (RTCP — протокол управления сетевыми компонентами). Постоянная обратная связь обеспечивает динамическую настройку на обеспечение наилучшего качества при текущих обстоятельствах для используемых приложений реального времени. Строго говоря, пары RTP и RTCP работают над протоколом UDP (их пакеты инкапсулируются в дейтаграммы UDP), поэтому многие исследователи относят их к прикладному уровню стека протоколов

TCP/IP [4]. Подобным же образом работают и методы мониторинга других потоковых протоколов.

Для мониторинга состояния ресурсов сети при их управлении возможно использование утилит протокола межсетевых управляющих сообщений Internet Control Message Protocol (ICMP), в основе которых так же, как и в протоколе TCP, лежит измерение RTT. Так, утилита *ping* устанавливает факт наличия связи между узлами сети и измеряет RTT между запрашивающим и запрашиваемым узлами. Утилита *tracert* (или *tracert* в OS Windows) позволяет многократным использованием утилиты *ping* установить маршрут и данные по RTT для всех узлов сети на маршруте от запрашивающего узла сети до запрашиваемого при условии, что эти узлы обладают собственными IP-адресами. К недоработкам утилиты *ping* (а следовательно, и утилиты *tracert*) следует отнести разрешающие способности, т.е. точность измерения RTT (микросекунды). Для глобальных сетей это оказывается достаточным, однако в корпоративных локальных сетях RTT даже для нескольких сегментов с высокой пропускной способностью (технологии Fast Ethernet, Gigabit Ethernet, 10 Gigabit Ethernet) не превышает погрешности измерений. Именно поэтому для исследования времени задержки трафика коммутаторами приходится модифицировать процедуру *ping*, привлекая для измерения наиболее быстродействующие элементы компьютеров (тактовые генераторы процессоров) [5]. Другим, не столь явным недостатком процедуры *ping* является использование протокола ICMP, о чем в соответствующем поле IP-заголовка пакета уведомляет код типа протокола. Для узлов сети, поддерживающих метод BRFQ и WFQ, пакетам, содержащим эхо-запросы и эхо-ответы, будет отдан приоритет при обслуживании, в первом случае из-за короткой длины пакета (при технологии BRFQ), а во втором (WFQ) — еще и из-за приоритетности управляющих сообщений в сети по отношению к пользовательскому трафику. Поэтому использовать утилиту *ping* для мониторинга UDP- и TCP-трафиков для корпоративных сетей, построенных и функционирующих на основе современных сетевых технологий, следует весьма корректно. В общем случае при мониторинге маршрутов в корпоративных сетях необходимо программное обеспечение, осуществляющее зондирование ресурсов с приоритетом того же уровня, к которому относится трафик штатного информационного потока на данном маршруте.

СПИСОК ЛИТЕРАТУРЫ

1. О л и ф е р В. Г., О л и ф е р Н. А. Компьютерные сети. – СПб.: Питер, 2007.
2. Т а н е н б а у м Э. Компьютерные сети. – СПб.: Питер, 2009.

3. Ф и л и м о н о в А. Построение мультисервисных сетей Ethernet. – СПб.: БХВ, 2007.
4. С т о л л и н г с В. Современные компьютерные сети. – СПб.: Питер, 2003.
5. Р а з р а б о т к а методов и алгоритмов управления интенсивностью трафика в коммутируемых сегментах корпоративной сети / М.К. Бойченко и др. // Отчет о НИР 1.10.09 – М.: МГТУ, 2009.

Статья поступила в редакцию 26.04.2010

Максим Константинович Бойченко родился в 1978 г., окончил МГТУ им. Н.Э. Баумана в 2001 г. Ведущий программист лаборатории ИТ Управления информатизации МГТУ им. Н.Э. Баумана. Специализируется в области информационных и телекоммуникационных технологий.

M.K. Boichenko (b. 1978) graduated from the Bauman Moscow State Technical University in 2001. Leading programmer of “Information Technologies” laboratory of Administration on Informatization of the Bauman Moscow State Technical University. Specializes in the field of information and telecommunication technologies.

Игорь Потапович Иванов родился в 1955 г., окончил в 1979 г. МВТУ им. Н.Э. Баумана. Проректор по информатизации, заведующий кафедрой “Теоретическая информатика и компьютерные технологии” МГТУ им. Н.Э. Баумана. Канд. техн. наук, доцент. Автор более 30 научных работ в области информационно-коммуникационных технологий.

I.P. Ivanov (b. 1955) graduated from the Bauman Moscow Higher Technical School in 1979. Ph. D. (Eng.), vice-rector on informatization, head of “Theoretical Bases of Information and Computer Technologies” department of the Bauman Moscow State Technical University. Author of more than 30 publications in the field of data and communication technologies.