УДК 621.39

С.С. Батыр, Г.В. Ступак, А.В. Хорхордин
Донецкий национальный технический университет, г. Донецк,
кафедра автоматики и телекоммуникаций
E-mail: sbatyr@gmail.com, stupakgv@gmail.com

ИССЛЕДОВАНИЕ МЕТОДОВ ИЗМЕРЕНИЯ УРОВНЯ САМОПОДОБИЯ ДЛЯ ПОТОКА ДАННЫХ

Abstract
Batyr S.S., Stupak G.V., Khorkhordin A.V. Investigation of methods for measuring the level of self-flow data*. Two methods of getting statistical information about traffic flow were described. Self-similarity level of data flow is very important parameter. Model for logging traffic flows was developed. Self-similarity was calculated for both methods of logging. Results of analyses described that both methods can be used for self-similarity test.*
Keywords: *traffic, network, self-similarity, carrying capacity, port, model of OSI.*

Анотація
Батир С.С., Ступак Г.В., Хорходін О.В. Дослідження методів вимірювання рівня само подібності для потіка даних. *В статі описані два методи збору статистичної інформації потоку даних. Рівень самоподібності трафіку є дуже важливим параметром. Була розроблена модель збору трафіка. Був визначений рівень само подібності трафіка для обох випадків. Результати свідчать про те що обидва методи можуть бути використані для провадження визначення рівня самоподібності.*
Ключові слова*: трафік, мережа, самоподібність, пропускна спроможність, порт, модель OSI.*

Аннотация
Батыр С.С., Ступак Г.В., Хорхордин А.В. Исследование методов измерения уровня само подобия для потока данных. *В статье описаны два метода сбора статистической информации потока данных. Уровень само подобия трафика является очень важным параметром. Была разработана модель сбора трафика. Был определен уровень самоподобности трафика для обоих случаев. Результаты говорят о том, что оба метода могут быть использованы для определения уровня само подобия.*
Ключевые слова: *трафик, сеть, самоподобие, пропускная способность, порт, модель OSI.*

Общая постановка проблемы. Интенсивное развитие экономики связано с совершенствованием технологии и организации производства на базе широкого применения вычислительной техники и средств телекоммуникаций. Одной из форм повышения эффективности производства является использование компьютерных сетей для управления многопрофильными структурами, сложными энергетическими системами, гибкими производственными системами, телекоммуникационным оборудованием в сетях связи, для автоматизации измерений и в других целях. Успешная работа многих организаций и компаний сегодня напрямую зависит от средств телекоммуникаций. Большую роль в деловой жизни стали играть Internet и мультимедиа. В настоящее время из-за интенсивного роста числа пользователей и различных приложений в телекоммуникационных системах существуют десятки разновидностей трафика, что влияет на уровень загруженности каналов и прочих

элементов сети. Перегруженность каналов может порождать следующие проблемы функционирования компьютерных сетей:

- конечные пользователи не удовлетворены уровнем задержки при работе с сетевыми приложениями;
- дистанционно обслуживаемые клиенты выражают неудовлетворенность неустойчивым информационным обслуживанием;
- затруднено выявление причин появления случайно возникающих перегрузок сети и возможность их локализации.

Для решения проблем в сети необходимо с достаточным уровнем достоверности определить следующее [1]:

- какие аппаратные средства перегружены и требуют замены;
- какие аппаратные и/или программные средства следует выбрать для замены, чтобы устранить “узкие места сети”;
- какие изменения следует вносить в распределение функций и организацию работы серверов.

Для определения перегрузки, источников с повышенной нагрузкой и узких мест сети необходимо использование специализированных средств для сбора статистики. Таковыми, как правило, выступают либо аппаратные, либо программные продукты.

Постановка задачи исследования. Параметры сетевой нагрузки носят стохастический характер, и предсказать уровень загруженности каналов современными средствами достаточно сложно. В первую очередь это связано с ростом информационных потоков и нестационарностью сетевой нагрузки. Отсюда возникают проблемы с рациональным использованием пропускных способностей существующих каналов связи. Как было доказано в многочисленных работах, а также в работе [2], сетевой трафик имеет самоподобную структуру. Целью исследования является выяснение наиболее адекватного способа получения статистики работы сети в условиях самоподобного трафика.

Решение задачи и результаты исследований. На сегодняшний день существует два основных подхода к оценке загруженности сети — использование сниффера либо статистики сетевого интерфейса. Ключевое отличие состоит в алгоритмах роботы и получаемой информации.

Сниффер или анализатор трафика — сетевой анализатор трафика, программа или программно-аппаратное устройство, предназначенное для перехвата и последующего анализа, либо только анализа сетевого трафика, предназначенного для других узлов.

Как известно [5], перехват трафика может осуществляться:

- обычным «прослушиванием» сетевого интерфейса (метод эффективен при использовании в сегменте концентраторов (хабов) вместо коммутаторов (свитчей), в противном случае метод малоэффективен, поскольку на сниффер попадают лишь отдельные фреймы);
- подключением сниффера в разрыв канала;
- ответвлением (программным или аппаратным) трафика и направлением его копии на сниффер;
- через анализ побочных электромагнитных излучений и восстановление таким образом прослушиваемого трафика;
- через атаку на канальном (MAC-spoofing) или сетевом уровне (IP-spoofing), приводящую к перенаправлению трафика жертвы или всего трафика сегмента на сниффер с последующим возвращением трафика в надлежащий адрес.

Анализ прошедшего через сниффер трафика позволяет:

- обнаружить паразитирующий, вирусный и закольцованный трафик, наличие которого увеличивает загрузку сетевого оборудования и каналов связи (снифферы

здесь малоэффективны; как правило, для этих целей используют сбор разнообразной статистики серверами и активным сетевым оборудованием и её последующий анализ).

- выявить в сети вредоносное и несанкционированное ПО, например, сетевые сканеры, флудеры, троянские программы, клиенты пиринговых сетей и другие (это обычно делают при помощи специализированных снифферов — мониторов сетевой активности).
- перехватить любой незашифрованный (а порой и зашифрованный) пользовательский трафик с целью получения паролей и другой информации.
- локализовать неисправность сети или ошибку конфигурации сетевых агентов (для этой цели снифферы часто применяются системными администраторами).

Поскольку в «классическом» сниффере анализ трафика происходит вручную, с применением лишь простейших средств автоматизации (анализ протоколов, восстановление TCP-потока), то он подходит для анализа лишь небольших его объёмов (небольших локальных сетей).

Другим средством [5] получения статистики является снятие информации с сетевого интерфейса. В общем случае получить данную статистику ранее позволяли утилиты, которые шли в комплекте с сетевыми интерфейсами. На сегодняшний день получение наиболее полных результатов возможно в операционной среде Linux при использовании утилиты *ifconfig*. Команда *ifconfig* используется для конфигурирования сетевых интерфейсов ядра системы, а также при отладке или настройке производительности системы. При помощи *ifconfig* возможен контроль следующих уровней модели OSI и соответствующих параметров:

1-й уровень OSI(физический):
- наличие несущей в проводе, подведённом к интерфейсу;
- характеристики Ethernet;

2-й уровень OSI (канальный):
- MAC-адрес (он же hardware address);
- флаги интерфейса;

3-й уровень OSI (сетевой):
- MTU — Maximum Transfer Unit — максимальный размер пакета, который можно передать/принять через данный интерфейс. Пакет имеется ввиду на сетевом уровне модели OSI, порция информации на канальном уровне называется «кадр»;
- IP-адрес;
- IPv6-адрес;
- маска подсети;
- широковещательный адрес.

Ниже приведена схема, (рисунок 1), иллюстрирующая деление средств анализа сетей, построенных на описанных выше способах с кратким описанием их функций.

На рисунке 2 приведена схема исследования, построенная на базе ПК с ОС Linux. Использованы стандартные утилиты для сбора и автоматизации сбора данных.

Для анализа сетевого трафика использовался персональный компьютер с операционной системой Linux. В качестве средств мониторинга сетевого трафика выступали стандартные утилиты — сниффер *tcpdump*, и команда *ifconfig*, для снятия статистики с сетевого интерфейса. Наблюдения проводились в рамках сети кафедры "Автоматики и телекоммуникаций" Донецкого национального технического университета в течении двух рабочих недель.

Приведем выдержку из файлов отчетов с результатами выполнения команд, а также дадим их расшифровку.

Рисунок 1 — Структура элементов анализа и мониторинга сетей

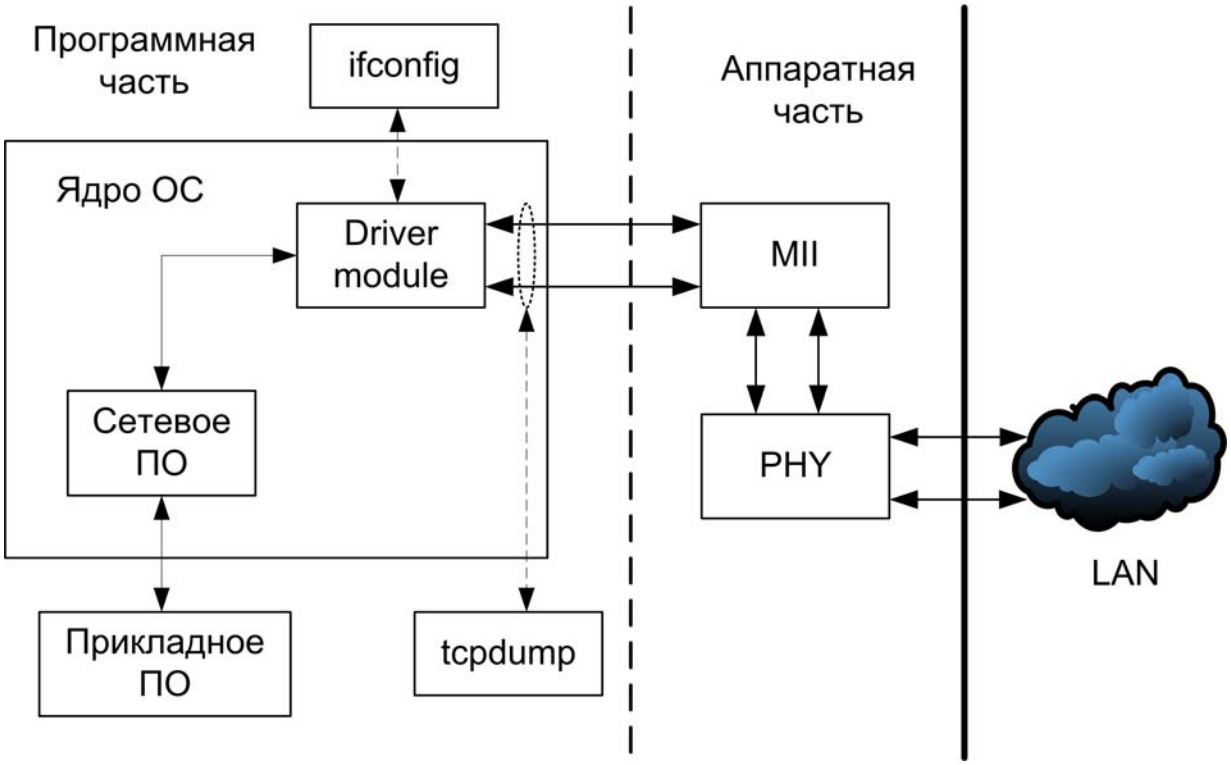

Рисунок 2 — Аппаратно-программная модель схемы сбора данных

Пример результатов работы сниффера *tcpdump*:

```
tcpdump -n -vvv -tttt -s 96 -i eth0
```

Фрагмент лог-файла:

```
02/05/2009 09:39:24.855341 10.0.0.138.4340 > 10.0.0.197.3128: S [tcp sum ok]
3298328004:3298328004(0) win 65535 <mss 1460,nop,nop,sackOK> (DF) (ttl 128, id 43021, len 48)
```

02/05/2009 09:39:24.855396 10.0.0.197.3128 > 10.0.0.138.4340: S [tcp sum ok]
1775499054:1775499054(0) ack 3298328005 win 5840 <mss 1460,nop,nop,sackOK> (DF) (ttl 64, id 0, len 48)
02/05/2009 09:39:24.855567 10.0.0.138.4340 > 10.0.0.197.3128: . [tcp sum ok] 1:1(0) ack 1 win 65535 (DF) (ttl 128, id 43023, len 40)
02/05/2009 09:39:24.885077 10.0.0.138.4340 > 10.0.0.197.3128: P 1:667(666) ack 1 win 65535 (DF) (ttl 128, id 43024, len 706)
02/05/2009 09:39:24.885117 10.0.0.197.3128 > 10.0.0.138.4340: . [tcp sum ok] 1:1(0) ack 667 win 6660 (DF) (ttl 64, id 39666, len 40)

В отчете содержится полное время перехвата пакета с точностью до микросекунд, адрес источника и приемника с портами, полная информация о данных в пакете, включая флаги протоколов IP и TCP.

Пример результатов работы комады *ifconfig*:

```
date +%s >> /home/ncftp/interface-`date +%F`
ifconfig eth0 |grep X >> /home/ncftp/interface-`date +%F`
```

Фрагмент лог-файла:

```
1233661201
      RX packets:171633396 errors:0 dropped:0 overruns:0 frame:0
      TX packets:172820775 errors:0 dropped:0 overruns:0 carrier:0
      RX bytes:135169609096 (128907.7 Mb)  TX bytes:140363140486 (133860.7 Mb)
1233661261
      RX packets:171634642 errors:0 dropped:0 overruns:0 frame:0
      TX packets:172822027 errors:0 dropped:0 overruns:0 carrier:0
      RX bytes:135170165227 (128908.3 Mb)  TX bytes:140363695515 (133861.2 Mb)
```

Команды запускаются каждую минуту с помощью демона cron. Первая строка содержит время сбора статистики в формате Unix: в секундах от начала эпохи (00:00 1 января 1970 г.). Следующие три строки содержат информацию о принятых и переданных пакетах от включения системы и объеме принятых и переданных байт.

Оценку уровня самоподобия для каждого из измерений будем производить в соответствии с методикой, изложенной ниже.

Дадим определения [4] строго и асимптотически самоподобных в широком смысле случайных процессов дискретного аргумента и укажем их связь с процессами, самоподобными в узком смысле, и с процессами с медленно убывающей зависимостью. Следует заметить, что теория само подобного телетрафика проходит относительно раннюю стадию своего развития, по этой причине существуют некоторые различия в терминологии и даже в определениях.

Обозначения. Пусть $X=(X1, X2,...)$ — полубесконечный отрезок стационарного в широком смысле случайного процесса дискретного аргумента (времени) $t \in N^{\Delta} = \{1,2,....\}$. Обозначим через $\mu < \infty$ и $\sigma^2 < \infty$ среднее и дисперсию процесса X соответственно, а через

$$r(k)^{\Delta} = \frac{\overline{(X_{t+K} - \mu)\cdot(X_t - \mu)}}{\sigma^2}$$

$$b(k)^{\Delta} = \sigma^2 \cdot r(k), \qquad k \in Z_+^{\Delta} = \{0,1,2....\}$$

автокорреляционную функцию и автоковариацию процесса X. Так как процесс X . стационарный в широком смысле, среднее $M[X]=\mu$, дисперсия $D[X] = \sigma 2 \equiv b(0$, коэффициент корреляции $r(k)$ и автоковариация $b(k)$ не зависят от времени t и $r(k)=r(–k)$, $b(k)=b(–k)$.

Допустим, процесс X имеет автокорреляционную функцию следующего вида:

$$r(k) \sim k - \beta L1(k),\ k \to \infty, \qquad (1)$$

где $0<\beta<1$ и L_1 медленно меняющаяся на бесконечности функция, то есть $\lim_{t\to\infty}\frac{L_1(tx)}{L_1(t)}=1$ для всех $x > 0$ (примерами медленно меняющейся функции могут служить $L1(t)=const$, $L1(t)=log(t)$).

Обозначим через $X^{(m)}=(X_1^{(m)}, X_1^{(m)},...)$ усредненный по блокам длины m процесс X, компоненты которого определяются равенством

$$X_t^{(m)^\Delta}=\frac{1}{m}\left(X_{tm-m+1}+...+X_{tm}\right),\quad m,t\in N \tag{2}$$

В дальнейшем изложении, будем называть такой ряд агрегированным [4]. Обозначим через $r_m(k)$, $b_m(k)$ и $V_m=b_m(0)$ коэффициент корреляции, автоковариацию и дисперсию процесса $X^{(m)}$ соответственно. Приведем ниже определение строго самоподобного в широком смысле процесса.

Определение. Процесс X называется строго самоподобным в широком смысле [ССШС] (exactly second-order self-similar) с параметром $H=1-(\beta/2)$, $0<\beta<1$, если

$$r_m(k)=r(k),\ k\in Z+,\ m\in\{2,3...\}, \tag{3}$$

т.е. ССШС процесс не меняет свой коэффициент корреляции после усреднения по блокам длины m. Другими словами, X является ССШС, если агрегированный процесс $X^{(m)}$ неотличим от исходного процесса X, как минимум в отношении статистических характеристик второго порядка.

Параметр H [4], называемый коэффициент Хэрста [Hurst parameter], имеет принципиальное значение в теории самоподобных процессов. Он является индикатором степени самоподобия процесса, а также свидетельствует о наличии у него таких свойств как персистентность/антиперсистентность и продолжительная память.

Определение. Процесс X называется асимптотически самоподобным в широком смысле (АСШС) [second-order asymptotical selfsimilarity] с параметром $H=1-(\beta/2)$, $0<\beta<1$, если

$$\lim_{m\to\infty} r_m(k)=g(k),\quad k\in N \tag{4}$$

где $g(k)=[(k-1)^{2-\beta}-2k^{2-\beta}+(k-1)^{2-\beta}]/2$ коэффициент корреляции ССШС.

Смысл этого определения состоит в том, что X является АСШС процессом, если после усреднения по блокам длины m и при $m\to\infty$ он сходится к ССШС процессу.

Вместе с понятием ССШС существует понятие просто само подобного процесса, которое для большего терминологического различия мы будем называть самоподобным в узком смысле процессом (СУС).

Определение. Процесс X называется самоподобным в узком смысле (СУС) [strictly self-similarity] с параметром $H=1-(\beta/2)$, $0<\beta<1$, если справедливо выражение

$$m^{1-H}X^{(m)}\overset{\bullet}{=}X,\quad m\in N \tag{5}$$

которое понимается в смысле равенства распределений. Связь междупроцессами ССШС и СУС аналогична связи между процессами, стационарными в широком и узком смыслах.

В работе [3] исследователи отметили одинаковость распределений исходного и агрегированного процессов (при значительном интервале изменения m), измерили параметр H (параметр Хэрста) и обнаружили, что последний для сетевого трафика находится в интервале (0.5, 1). На качественном уровне такой самоподобный трафик имеет постоянный, взрывной. характер [burstiness], то есть обладает высокой пачечностью на многих масштабах временной оси. Напомним, что коэффициент пачечности (пачечность) для заданного процесса соответствует отношению пиковой интенсивности процесса поступления заявок на обслуживание к его среднему значению[4].

Произведем анализ перехваченных данных. Вычислим среднее количество принятых и переданных пакетов, байт за сутки при различных уровнях агрегации. Интервал снятия статистики с сетевого интерфейса составляет 10 с.

При агрегации среднее значение и девиация увеличивают свои значения, поэтому проведем оценку уровня самоподобия анализируемого потока данных.

Таблица 1 — Среднее количество и девиация принятых/переданных пакетов и байт

		исходный		6 точек		36 точек		180 точек	
		M	D, 1,0e+09	M	D, 1,0e+09	M	D, 1,0e+09	M	D, 1,0e+09
Статистика интерфейса	Пр. пак.	19339	21.0157	22565	28.2183	26301	38.3170	31312	55.6226
	Пер. пак.	27879	0.4311	32525	0.5810	37891	0.7910	45059	1.1494
Сниффер	Пр. байт	3264445	62.9259	3812169	82.0119	445438	108.0928	5354437	155.5291
	Пер. байт	3303956	0.5599	3853543	0.7562	4487823	1.0299	5336604	1.5013

Таблица 2 — Уровни самоподобия при различной степени агрегации

	H	6 точек	36 точек	180 точек
Статистика интерфейса	Пр. пак.	0.9178	0.9162	0.9063
	Пер. пак.	0.9167	0.9153	0.9056
Сниффер	Пр. байт	0.9261	0.9245	0.9129
	Пер. байт	0.9161	0.9150	0.9050

Значения параметра Херста превышают 0.5, что указывает на высокий уровень самоподобности исследуемого процесса.

Выводы. Сравнительный анализ двух способов сбора данных показал, что статистику сетевого интерфейса можно использовать на ряду с логами сетевого сниффера. Логирование потока с помощью сниффера дает более детальные характеристики потока, но повышает нагрузку центрального процессора системы, что ограничивает возможность применение такого метода на маршрутизаторах, обслуживающих высокоскоростные линии связи с высоким уровнем загрузки. Статистика сетевого интерфейса предоставляет менее детальные данных о потоке, однако ее учет ведется драйвером на уровне ядра, что значительно проще. При этом собранных данных достаточно для идентификации уровня самоподобия потока данных, проходящего через сетевой интерфейс.

Литература

1. Л.И. Абросимов. Измерение характеристик корпоративной вычислительной сети банка / Абросимов Л.И., Беззубченко И.Л., Невзоров Ю.Ю., Горбунов М.Е., Электронный журнал ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ. Теория и практика, 2005, №1 (6), раздел 10, статья 1.
2. Б.С. Цыбаков. Модель телетрафика на основе самоподобного случайного процесса. / Б.С. Цыбаков //Радиотехника, 1999. — № 5. — С. 24–31
3. В.И. Нейман. Новое направление в теории телетрафика / В.И. Нейман. // Электросвязь. .1998. — № 7. — С. 27–30.
4. W.E. Leland. On the self-similar nature of ethernet traffic/ Leland W.E., Taqqu M.S., Willinger W., and Wilson D.V.// IEEE/ACM Transactions of Networking, 2(1), 1994. — P. 1–15.
5. Столлингс В. Современные компьютерные сети. 2-е издание. — С.Пб.: Питер, 2003. — 783 с.

Здано в редакцію:
18.02.2009р.

Рекомендовано до друку:
д.т.н, проф. Скобцов Ю.О.