

ДИАГНОСТИКА p2p-АКТИВНОСТИ НА ОСНОВЕ АНАЛИЗА ПОТОКОВ NetFlow

С. В. Бредихин, Н. Г. Щербакова

Институт вычислительной математики и математической геофизики СО РАН,
630090, Новосибирск, Россия

УДК 004.415.532

Рассматривается задача идентификации p2p-трафика. Приведен краткий обзор методов выявления такого трафика и представлен метод идентификации потоков трафика путем анализа информации транспортного уровня. В основе метода лежат эмпирические предположения о поведении участников p2p-обмена. Исходными данными являются потоки трафика, сформированные сетевой операционной системой. Разработан и реализован алгоритм деления потоков на два класса в пассивном режиме. Проведена апробация алгоритма на реальных данных и приведена оценка его эффективности.

Ключевые слова: p2p-трафик, TCP/UDP-порты, NetFlow, эффективность алгоритма, Snort.

The problem of identifying p2p traffic is considered. Some of the currently proposed p2p traffic classification methods are observed and the transport layer identification method is introduced. The method is based on the heuristics about the behavior of p2p pairs during data transfer. Netflows formed by the net ios are examined as traffic sources. The algorithm is designed for analysis of passive traffic traces that classifies netflows accorditg two categories. It was approbated on real network traffic, its accuracy was evaluated.

Key words: p2p traffic, TCP/UDP ports, NetFlow, algorithm efficiency, Snort.

Введение. В настоящей работе рассматриваются методы анализа информационных потоков IP-сетей, позволяющие находить в их составе трафик, характерный для p2p-приложений. В общем случае задача идентификации p2p-приложений является частью проблемы классификации интернет-трафика. С развитием файлообменных сетей стало сложно идентифицировать p2p-трафик с помощью номеров портов [1–3]. Более того, возможны случаи использования p2p-портов другими приложениями. Например, при увеличении номера клиентского порта при обращении к проху-серверу номер может совпасть с номером, используемым p2p-приложением. В работе [4] приведены результаты, показывающие, что зачастую на основе номеров портов можно классифицировать только 30% p2p-трафика. Возникла необходимость исследовать трафик на основании поведения участников обмена. Цель исследования состоит в том, чтобы с достаточной степенью точности определить, сгенерирован ли изучаемый IP-трафик каким-либо p2p-приложением.

Сложность классификации обусловлена прежде всего разнообразием сетевых приложений, недостаточно полным их описанием и необходимостью собирать данные о трафике на загруженных высокоскоростных линиях связи. В то же время современные маршрутизаторы имеют возможность аккумулировать характеристики трафика, проходящего через интерфейсы, и экспортировать эту информацию в виде потоков. Однако заметим, что в процессе аккумуляции часть информации о трафике может быть утеряна. Например, для

применения алгоритмов анализа временных интервалов между пакетами такой способ сбора данных неприемлем.

В настоящей работе в качестве исходных данных о трафике исследуемого сетевого фрагмента будем рассматривать только ту информацию, которая представлена в потоках NetFlow [5], агрегированных операционной системой маршрутизатора. В связи с этим более подробно рассмотрим определение структуры потока. В работе [6] предложена методика параметризации потоков трафика. В настоящей работе предлагается отказаться от широко используемого определения потока как трафика, соответствующего ТСР-сессии и сформулировать такое определение потока и его метрик, которое было бы пригодно для формальной постановки достаточно широкого диапазона задач исследования трафика IP-сетей.

В [6] выделены четыре аспекта, определяющие структуру потока. Во-первых, поток однонаправленный либо двунаправленный. В общем случае предпочтительно рассматривать однонаправленный трафик, так как даже в случае двунаправленных ТСР-сессий направления имеют асимметричные характеристики.

Во-вторых, большое значение имеет способ выполнения агрегации потока — по одной либо по двум оконечным точкам. Различаются потоки с одной и с двумя оконечными точками. В [6] рассматриваются следующие подходы: объединение в одном потоке всего трафика, направленного на оконечную точку; объединение всего трафика, исходящего из одной точки; объединение в поток трафика между двумя точками.

В-третьих, важной характеристикой является глубина (степень) детализации каждой конечной точки по следующим параметрам трафика: приложениям, оконечным пользователям, хостам, административным областям, провайдерам, точкам внешней коннективности, узлам или участкам опорной сети. Степень детализации зависит от задачи. (В настоящей работе рассматривается детализация по сетям и хостам, а также детализация по паре IP-адрес/порт.)

В-четвертых, потоку могут соответствовать различные функциональные уровни. Можно определять поток на прикладном уровне или рассматривать транспортный уровень и выявлять, например, начало и конец ТСР-соединения. В работе [6] поток определяется только на основе активности передачи пакетов между оконечными точками сетевого уровня, так как это наиболее общий подход, соответствующий сути интернет-среды.

Время формирования потока определяется параметром timeout. Согласно выбранной структуре (профилю) пакеты собираются в поток, пока интервал между соседними пакетами меньше определенного значения. Стандартным значением принято считать значение, равное 64 с.

1. Задача и обзор работ. В настоящей работе рассматривается задача идентификации р2р-трафика в потоках, сформированных на оборудовании фирмы Cisco Inc операционной системой IOS. Эта задача является частью проблемы детерминированной классификации и формулируется следующим образом. Дано множество исследуемых объектов $X = \{x_1, \dots, x_n\}$. Имеется набор классов $C = \{c_1, \dots, c_k\}$. Требуется найти функцию отображения $f : X \rightarrow C$, так чтобы каждый объект x_i принадлежал только одному классу. Необходимо построить алгоритм, реализующий требуемое отображение, и интерпретировать полученные результаты.

Существует несколько альтернативных подходов к выявлению р2р-трафика. Достаточно хорошие результаты можно получить с помощью алгоритмов, основанных на изучении содержимого ТСР/UDP-пакетов и выявлении шаблонов прикладного уровня, соответствующих р2р-приложениям. В [7] приведен обзор работ, в которых применяется данный метод,

характеризующийся высокой ресурсоемкостью. Возникает проблема использования высокоэффективных алгоритмов распознавания контекста и определения количества пакетов сессии, которые необходимо проверить, прежде чем вынести суждение. В работе [8] подробно обсуждаются проблемы, возникающие при создании эффективной системы выявления наиболее распространенных р2р-приложений, устойчивой к потере пакетов и несимметричности маршрутизации. Поскольку р2р-приложения недостаточно документированы, поиск адекватных шаблонов сам по себе является проблемой, тем более что прикладные протоколы могут претерпевать изменения. Кроме того, все чаще используемое шифрование делает выявление шаблонов невозможным. Ухудшает качество распознавания и большое количество пакетов без полезной нагрузки, наличие которых обусловлено попытками обращения к уже отключившимся абонентам.

Другим направлением решения задачи выявления р2р-трафика является изучение информации сетевого и транспортного уровней, содержащейся в заголовках IP-пакетов. Работы в этом направлении можно условно разделить на две группы. В первую группу входят работы, в которых изучаются поведенческие особенности участников р2р-обмена и приводятся схемы поведения, характерные для одного или нескольких р2р-приложений. В работах второй группы применяются методы и алгоритмы технологий Data Mining (DM) и Machine Learning (ML), оперирующие характеристиками потоков данных и средствами их очистки, автоматического анализа и визуализации результатов.

Наиболее известными работами первой группы являются [9, 10]. В работе [9] приведена методика разделения трафика на две категории, “р2р” и “не р2р”, основанная на эвристических предположениях о поведении участников р2р-обмена. Подход получил развитие в работе [10], где предложен метод классификации категорий приложений, основанный на изучении особенностей поведения отдельных хостов, которые рассматриваются на трех уровнях детализации: социальном (взаимодействие с другими хостами), функциональном (клиент или сервер) и прикладном (особенности поведения на транспортном уровне).

Если целью указанных выше работ является выявление всего трафика, относящегося к категории р2р, то в работе [11] исследуется только взаимодействие по протоколу BitTorrent [12]. Предложенный в настоящей работе метод позволяет выявлять BitTorrent-пары на основании поведения, обусловленного алгоритмом блокировки (choke), влияющим на различные характеристики потоков, такие как средний размер пакетов и интервалы между ними. Вводится несколько метрик, учитывающих особенности поведения участников обмена. Для принятия решения метрики сравниваются с порогами, однако найти баланс на уровне одной метрики достаточно сложно. Поэтому изучаются различные комбинации метрик и их влияние на эффективность алгоритма. Приведенные диапазоны значений для порогов определены в результате исследования трафика специально запущенных приложений BitTorrent.

Сигнальное поведение хостов, участвующих в р2р-обмене, исследуется также в работе [13], в которой под сигнальными пакетами понимаются пакеты небольшого размера (не более 100 байт). Для рассматриваемого хоста вычисляется количество контактов и посылаемых (принимаемых) сигнальных пакетов за определенный период. При этом контакт и соответствующие ему пакеты считаются старыми, если взаимодействие с данным контактом зафиксировано в предыдущий период. На уровне хоста исследуются следующие свойства: доля новых (старых) контактов (среднее, отклонение); скорость появления новых (старых) контактов (среднее, отклонение); коэффициент корреляции между количеством новых и старых контактов. На уровне сообщения исследуются аналогичные свойства для сигнальных

пакетов и дополнительное свойство — сравнительная скорость появления старых и новых пакетов. На тренировочных данных изучаются указанные свойства каждого приложения, затем проводится классификация. В настоящей работе приведено только краткое изложение методики, однако следует отметить, что в течение 15 мин на основании сигнального поведения можно распознать 99% трафика, относящегося к BitTorrent, eMule и Skype. Однако неясно, как эта методика будет работать, если на исследуемом хосте исполняется несколько приложений.

Характеристики на уровне отдельных протоколов позволяют более точно выделить соответствующую р2р-деятельность, но неустойчивы к изменению имеющихся протоколов и появлению новых. Поэтому наибольший интерес представляет выявление особенностей поведения, характеризующих р2р-деятельность в целом.

В настоящее время рассмотрено большое количество известных алгоритмов классификации и кластеризации DM&ML для задачи определения состава сетевого трафика. Как правило, ставится обобщенная задача разделения трафика на категории, представленные различными прикладными протоколами, такие как Mail, WWW, Database, р2р и др. Объектами исследования являются поток трафика и его характеристики. При этом потоки могут быть различными: двунаправленная последовательность пакетов между двумя IP-адресами, полная TCP-сессия или однонаправленная последовательность пакетов, определяемая на основе пяти элементов $\langle src_ip, src_port, dst_ip, dst_port, protocol \rangle$. Здесь src_ip — IP-адрес источника; src_port — порт источника; dst_ip — IP-адрес назначения; dst_port — порт назначения; $protocol$ — транспортный протокол. Как правило, в качестве протоколов транспортного уровня рассматриваются TCP и UDP.

Большое внимание уделяется выбору необходимых и непротиворечивых характеристик потоков. В работе [14] указаны преимущества и недостатки методов проверки адекватности набора атрибутов, позволяющих получить эффективные классификаторы. В работе [15] приведено 249 характеристик потоков, которые можно использовать при анализе. Трудность состоит в том, что необходимо выделить свойства, адекватно характеризующие приложения, но не зависящие от реализации, являющиеся уникальными и (желательно) позволяющие идентифицировать приложения в реальном времени.

В ряде работ приведено сразу несколько алгоритмов, применяемых к одному и тому же набору данных, с целью выявления наиболее эффективного алгоритма. В работе [16] сравниваются результаты кластеризации сетевого трафика с помощью алгоритмов K-Means, DBSCAN и Autoclass. В [17] исследуются пять алгоритмов, три алгоритма классификации C4.5, RandomForest, Naive Bayes и два алгоритма кластеризации K-Means и EM для решения задачи классификации в реальном времени. В работе [18] сравниваются алгоритмы классификации Naive Bayes и C4.5 с точки зрения их стабильности относительно состава исследуемых данных и длительности периода между настройками. В качестве критериев сравнения рассматриваются эффективность, вычислительная производительность, производительность процесса обучения, для деревьев решений — размеры деревьев. Из анализа результатов сравнения алгоритмов следует, что по эффективности они ближе друг к другу, чем по другим параметрам.

Следует отметить, что несмотря на высокую эффективность рассмотренных алгоритмов, при построении моделей практически всегда вводятся предположения относительно исследуемых данных. Например, рассматриваются определенные категории трафика, исследуются только полные TCP-сессии, предполагаются независимость характеристик потоков, минимальное влияние сетевого окружения на задержки между пакетами и т. д. В случае кон-

тролируемой классификации важной задачей является выбор адекватных тренировочных данных, а в случае кластеризации — интерпретация результатов.

2. Метод исследования трафика и рассматриваемые данные. Для исследования выбран основанный на эвристических предположениях о поведении участников р2р-обмена метод, предложенный в [9]. Конечной целью является разработка собственного алгоритма классификации, применимого к “готовым” потокам формата NetFlow, поставляемым сетевыми маршрутизаторами Cisco. Отсюда следуют ограничения: недоступна информация прикладного уровня; отсутствуют данные для вычисления “тонких” характеристик потоков (например, интервала между пакетами потока, медианы, отклонения и других характеристик, используемых в алгоритмах DM&ML).

Исходными данными служат пакеты трафика, захваченные утилитой Unix tcpdump на интерфейсе соединения локальной и опорной сетей. Рассмотрение необработанных данных обусловлено необходимостью иметь независимое суждение о распределении трафика. Пакеты размещаются в файлы, каждый из которых содержит трафик, собранный в течение 30 мин для одного дня сбора (аналогия с системой сбора потоков Cisco NetFlow). Файл обрабатывается утилитой преобразования пакетов в потоки согласно следующим правилам: рассматриваются только TCP/UDP-пакеты; однонаправленный поток собирается с учетом значений пяти элементов $\langle src_ip, src_port, dst_ip, dst_port, protocol \rangle$; поток считается законченным, если тайм-аут между пакетами составляет более 64 с. Результирующий файл содержит записи вида $\langle t_beg, t_end, src_ip, src_port, dst_ip, dst_port, protocol, num_packets, num_bytes, flags \rangle$, где t_beg, t_end — время начала и окончания сбора пакетов в поток; src_ip, src_port — адрес и порт источника; dst_ip, dst_port — IP-адрес и порт получателя; $num_packets$ и num_bytes — количество пакетов и байтов; $flags$ — побитный OR флагов в пакетах. В табл. 1 приведены результаты классификации трафика по категориям согласно номерам портов по умолчанию для двух наборов потоков D1 и D2.

Из табл. 1 следует, что классификация трафика по портам дает “грубое” распределение, поскольку большая часть трафика относится либо к категории Web, либо к категории Others.

3. “Абсолютная истина”. Для того чтобы выявить и проверить эвристики в условиях реального трафика, а также оценить работу предлагаемого алгоритма, необходим “туру”, позволяющий правильно классифицировать исследуемые данные. В рассматриваемом случае таким “туру” является система предотвращения вторжений Snort (<http://www.snort.org>), включающая средства идентификации р2р-трафика методом сигнатур. Система Snort предоставляет возможность создавать набор правил, включающих информацию о характеристиках транспортного уровня и содержимом полезной нагрузки пакетов для различных приложений. Поиск набора контекстов может производиться на всю глубину пакета. Следует отметить, что набор сертифицированных правил Snort для выявления р2р-трафика потребовалось дополнить, так как, например, наиболее распространенный в настоящее время р2р-протокол BitTorrent представлен только правилами относительно обмена данными, и не содержит правил относительно управления. В табл. 2 представлены шаблоны исследованных р2р-протоколов.

Результатом работы системы Snort является лог-файл, в котором содержится информация о выявленных пакетах, относящихся к категории р2р. Для того чтобы пометить соответствующей категорией потоки, применялась следующая процедура. Если время получения пакета, содержащего пять элементов $\langle src_ip, src_port, dst_ip, dst_port, protocol \rangle$ и идентифицированного как р2р, входит в интервал $[t_beg, t_end]$, соответствующий началу и окончанию сбора потока, определяемому теми же пятью элементами, то весь поток

Таблица 1

Классификация трафика по категориям

Категория трафика	Порты	Протокол прикладного уровня	D1, %	D2, %
Mail	25, 109, 110, 113, 143	smtp, pop2, pop3, ident, imap	0,66	0,13
Web	80, 8080, 443	http, https	41,97	39,05
data	20, 21, 3306, 66, 1521, 1526, 1524	ftp, MySQL, sqlnet, Oracle, Ingres	<0,01	0,01
Network manage-t	53, 137, 138, 139, 445, 161, 123, 783, 8200	domain, netbios, snmp, ntp, spamassassin, GoToMyPC	0,19	0,18
Interactive	22, 23, 513, 543	ssh, telnet, rlogin, klogin	3,32	24,01
nntp	119	nntp	0	0
Chat	194, 6891–6901, 1863, 5050, 5190	irc, msn messenger, yahoo messenger, ICQ	0,02	0,03
streaming	554, 1755, 1220, 8000–8005, 7070, 7071, 6970	rtsp, ms-streaming, Apple quicktime, internet radio (shoutcast), Real Audio & Video	<0,01	<0,01
Malware & games	1433, 1434, 666, 1999, 31337, 12345, 12346, 20034, 1024, 1025, 31338, 31339, 3127, 27015, 27016, 26000, 27001, 27960, 3724	Ms-sql-s, ms-sql-m, backdoor, Back Orifice, NetBus, netspy, myDoom, HalfLife, Quake, QuakeWorld, QuakeIII, WarCraft	<0,01	<0,01
p2p	411, 412, 1214, 3531, 4111, 4661–4665, 4672, 6346, 6347, 6669, 6881–6889, 23302, 32285, 59049, 41170, 57990	Direct Connect, Fasttrack, Kazaa, eDonkey, Gnutella, Napster, BitTorrent, Ares, Mp2p, Azureus	1,35	1,51
Others (остальные)	—	—	52,48	35,08

считается р2р. Кроме того, все потоки в обратном направлении, соответствующие данным пяти элементам, классифицируются как р2р. Если пакет с пятью элементами $\langle src_ip, src_port, dst_ip, dst_port, protocol \rangle$ классифицирован как р2р, то все пакеты, содержащие $\langle src_ip, src_port \rangle$ или $\langle dst_ip, dst_port \rangle$ (и соответственно потоки), в рассматриваемом интервале считаются “возможными р2р”.

4. Эвристические предположения. Рассматривается два эвристических предположения о поведении участников р2р-обмена [9].

1. TCP/UDP-эвристика. Если в течение рассматриваемого интервала времени для пары адресов зафиксированы как TCP-, так и UDP-потоки, то пары предположительно участвуют в р2р-обмене. В рассматриваемом случае такое поведение характерно для выявленных

Таблица 2

Протоколы и их шаблоны

р2р-протокол	Шаблон	Транспортный протокол	Порты по умолчанию	Тип пакета
eDonkey	"0xe311"	udp	4660-4799	Server status request
eDonkey	"0xe314"	tcp	4660-4799	File status request
Direct Connect	"\$MyINFO"	tcp (connection established)	411-412	Traffic
BitTorrent	"0x13BitTorrent protocol"	tcp (connection established)	6881-6889	Data transfer
BitTorrent	"CET" "/announce" "info_hash=" "event=started"	tcp (connection established)	—	Announce request
BitTorrent	"d1:ad2:id20:", "e1:q9:get_peers1:"	udp	—	DHT get_peers request
BitTorrent	"d1:rd2:id20:"	udp	—	DHT nodes reply
BitTorrent	"d1:ad2:id20:"	udp	—	DHT ping
Skype	"17030100"	tcp (established, to client)	—	Login
Skype	"16030100"	tcp (established, to server)	—	Login startup

системой Snort протоколов BitTorrent, Skype и eDonkey. Взаимодействие с адресами абонентов, выявленных с помощью TCP/UDP-эвристики, также зачастую указывает на р2р-деятельность.

2. IP/Port-эвристика. Факт, что при обращении к паре $\langle dst_ip, dst_port \rangle$ количество адресов источников практически совпадает с количеством портов источников, характеризует р2р-деятельность. Такое поведение характерно, например, для сигнального взаимодействия с раздающим данные BitTorrent-клиентом. При обращении к web-серверу клиенты обычно открывают сразу несколько соединений (web-эвристика) с разными номерами портов, поэтому количество адресов клиентов обычно не равно количеству портов.

Следует отметить, что для некоторых протоколов, отличных от р2р-протоколов, также характерны указанные признаки. Например, для ряда протоколов, таких как domain, netbios, irc, игры и потоковые приложения, характерно использование парой участников обмена одновременно протоколов TCP и UDP (табл. 3).

5. Алгоритм. Ниже предлагается алгоритм find_p2p, выполняющий классификацию адресов и потоков за текущий интервал времени. В рассматриваемом случае интервал равен 5 мин. Главной задачей алгоритма является поиск взаимодействий, соответствующих выбранным р2р-эвристикам.

Шаг 1. Рассматриваются пары адресов, взаимодействующие одновременно по протоколам TCP и UDP. Если при этом TCP/UDP-порты не входят в табл. 3, то оба адреса заносятся

Таблица 3

Исключения	
Сервис	Порт
NETBIOS	137, 138, 139
Microsoft-DS	445
DNS	53
NTP	123
Isakmp	500
Streaming: RTSP, Real Audio & Video (неофициально), ms-streaming, Yahoo messenger voice chat (неофициально)	554, 7070, 6970, 1755, 5000, 5001
Gaming & Malware: StarCraft, Backdoor (неофициально)	6112, 6868, 6899
IRC (неофициально)	6667, 7000, 7514

в массив $p2p_addrs$. Все потоки, содержащие хотя бы один адрес из $p2p_addrs$, маркируются как р2р. Адреса, встречающиеся в этих потоках и отличающиеся от входящих в $p2p_addrs$, заносятся в массив $p2p_addrs1$ вместе с номерами портов. Это также р2р-адреса, но их взаимодействие рассматривается только с учетом номеров портов, в отличие от [9]. Экспериментально установлено, что количество адресов, ложно идентифицированных как р2р, уменьшается, так как часто с одного адреса одновременно устанавливается несколько сессий по разным прикладным протоколам.

Шаг 2. Для каждой неклассифицированной пары $\langle IP\text{-address}, port \rangle$, встречающейся в потоках в качестве $\langle dst_ip, dst_port \rangle$, заводятся два массива — массив различных адресов источников $IPSet$ и массив различных портов источников $PortSet$. Если в конце интервала оказывается, что $IPSet$ содержит более двух адресов, а разница между размерами массивов меньше двух, то считается, что пара $\langle IP\text{-address}, port \rangle$ принимает участие в р2р-деятельности. Все потоки, в которых встречается эта пара, маркируются как р2р, а адреса и порты заносятся в массив $p2p_pairs$. В этом заключается отличие данного алгоритма от алгоритма, предложенного в [9], где рассматриваются все пары $\langle IP\text{-address}, port \rangle$.

Для того чтобы исключить из разряда р2р деятельность, для которой характерна одна из эвристик, определяются эвристические предположения, выявляющие взаимодействие “не р2р”.

Шаг 3. Обозначим через $mail_port$ порт, принадлежащий множеству портов, относящихся к почтовой службе: $\{25,110,113\}$. IP-адрес будем считать адресом почтового сервера, если при исследовании потоков выяснится, что существуют потоки, в которых пара $\langle IP\text{-address}, mail_port \rangle$ является источником, и потоки, в которых пара $\langle IP\text{-address}, mail_port \rangle$ является назначением. Будем говорить, что для IP-адреса верна mail-эвристика. Все потоки, содержащие адрес почтового сервера, считаются “не р2р”.

IP-адрес будем считать адресом сервера доменных имен, если существуют потоки, в которых пара $\langle IP\text{-address}, 53 \rangle$ является источником, и потоки, в которых пара $\langle IP\text{-address}, 53 \rangle$ является назначением. Все потоки, содержащие адрес сервера доменных имен, считаются “не

p2p”. Будем говорить, что для IP-адреса верна dns-эвристика. Заметим, что при этом потоки, содержащие обращения к dns-службе со стороны участников p2p-обмена, также считаются “не p2p”. Однако p2p-клиенты имеют небольшое количество обращений к dns-службе, так как получают нужную информацию друг от друга.

Игры и вредоносные программы (malware) характеризуются однотипными потоками, имеющими одну и ту же длину или небольшой разброс средних размеров пакетов в потоке (например, множество длин не превышает трех). Для исключения такого взаимодействия сохраняется соответствующая информация и проводится проверка. Если диапазоны характерны для игр или вредоносных программ, адреса и потоки считаются “не p2p”. Будем говорить, что верна malware-эвристика.

Если пара $\langle dst_ip, dst_port \rangle$ подвергается распределенному сканированию или атаке со стороны множества пар $\langle src_ip, src_port \rangle$, то обычно ответы от пары $\langle IP_address, port \rangle$ отсутствуют или их крайне мало. В этом случае, если данная пара не зафиксирована ранее как p2p, она считается “не p2p”, несмотря на верность IP/Port-эвристики. Будем говорить, что верна scan-эвристика.

Если пара $\langle ip, port \rangle$ встречается только в потоках, в которых порт источника относится к числу хорошо известных портов, то пара считается “не p2p”. Будем говорить, что верна history-характеристика.

Наконец, потоки, в которых порт источника равен порту назначения и оба меньше или равны 500, считаются “не p2p”, при этом соответствующие пары $\langle src_ip, src_port \rangle$ и $\langle dst_ip, dst_port \rangle$ также считаются “не p2p”. Такое поведение нехарактерно для p2p, но характерно для ряда легальных взаимодействий, например для сервисов ntp (порт 123) или domain (53) при коммуникации между dns-серверами. Будем говорить, что верна knownports-эвристика.

Информация обо всех парах, для которых верна одна из эвристик, представленных на шаге 3, сохраняется в массиве *Rejected* и учитывается при принятии решения.

Выполнение алгоритма find_p2p осуществляется в пассивном режиме за три прохода для каждого временного интервала. При первом проходе выявляются IP-адреса, для которых характерны mail-, dns- и TCP/UDP-эвристики и которые не классифицированы на предыдущих интервалах. В результате формируются массивы *Mail_servers*, *Dns_servers* и *p2p_addrs*. Последний массив состоит из адресов, использующих одновременно TCP и UDP, не являющихся mail- или dns-серверами, а также не входящих в разряд исключений. Потоки помещаются в массив *FT* для дальнейшего рассмотрения. С целью экономии оперативной памяти проход выделен в отдельный этап. При втором проходе просматриваются потоки из *FT*. Во-первых, массив *p2p_addrs1* пополняется адресами, взаимодействующими с адресами из *p2p_addrs*, а соответствующие потоки помечаются как p2p. Во-вторых, массив *p2p_pairs* пополняется адресами, взаимодействующими с адресами, выявленными ранее по IP/Port-эвристике. В третьих, пары $\langle dst_ip, dst_port \rangle$, не выявленные ранее как p2p (входящие в *p2p_addrs*, *p2p_addrs1* или *p2p_pairs*), не являющиеся mail- и dns-серверами и не классифицированные как “не p2p” (массив *Rejected*), помещаются в массив *IPPort*. Для этих пар сохраняется информация об адресах источников в массиве *IPSet*, портах источников в массиве *PortSet*, средних длинах пакетов (до четырех различных) и потоков (до двух различных), а также информация о том, были ли ответы (пара действует как $\langle src_port, dst_port \rangle$).

По окончании просмотра потоков рассматриваются пары из *IPPort*. Если оказалось, что с парой взаимодействует более двух адресов ($IPSet.length > 2$) и разность размеров масси-

вов *IPSet* и *PortSet* меньше 2 или 10 (в случае, если порт является известным р2р-портом), то для этой пары проводятся дополнительные проверки на все эвристики шага 3, которые позволяют классифицировать пару как “не р2р”. В зависимости от результатов пара заносится либо в массив *p2p_pairs*, либо в массив *Rejected*. Остальные пары заносятся в массив *Rejected*.

При третьем проходе массив *p2p_pairs* пополняется адресами, взаимодействующими с адресами, выявленными при втором проходе, а также выдается информация о р2р-адресах и р2р-потоках, идентифицированных в текущем интервале.

6. Оценка эффективности. Эффективность алгоритма определяется на основе двух основных параметров — *FP* и *FN*. Параметр *FP* (false positive) — доля трафика, приписанного к классу *X*, но не принадлежащего к *X* по отношению к мощности *X*. Параметр *FN* (false negative) — доля трафика, принадлежащего классу *X*, но не приписанного к классу *X*. Используются также понятия “правильность” (accuracy) — доля правильно классифицированных единиц по отношению ко всем классифицированным единицам ($(all-FP-FN)/all$); “точность” (precision) — доля правильно классифицированных единиц (*TP*) относительно полученного класса $TP/(TP+FP)$; “полнота” (recall) — доля правильно классифицированных единиц относительно реального класса $TP/(TP+FN)$.

В качестве примера приводятся результаты классификации 30-минутного файла D1, для которого сигнатурным методом зафиксирована большая доля р2р-трафика. Всего в трафике содержится 16 334 IP-адреса, 1 569 783 пакета, 979 268 468 байт, 67 269 потоков. Сигнатурным методом выявлено 11 326 р2р-адресов, что составляет приблизительно 69 %, вместо 2 % адресов, выявленных на основе номеров портов. Количество пакетов и байтов, выявленных сигнатурным методом, составляет приблизительно 10 и 3,2 % общего количества соответственно. В данном случае основная доля р2р-трафика соответствует протоколу BitTorrent (приблизительно 98 %). Несмотря на то что значительное количество адресов участвует в р2р-обмене, количество байтов трафика относительно небольшое, так как во многих случаях передача данных не осуществлялась.

Обозначим через *S* множество р2р-адресов, идентифицированных сигнатурным методом. В результате применения алгоритма *find_p2p* к файлу D1 12 735 адресов классифицированы как р2р. Обозначим это множество через *A*. 1815 адресов из *A* не входит в *S*, а 408 адресов из *S* не входит в *A*. Если считать, что множество *S* совпадает с классом адресов р2р (True), то $FP = 16\%$ ($1815 \cdot 100/11\,326$), $FN = 3,6\%$ ($408 \cdot 100/11\,326$).

В результате дополнительной проверки установлено, что 1135 адресов из 1815 относятся к классу р2р и только 90 адресов — к классу “не р2р”. При этом 590 адресов остались невыясненными (будем считать их “не р2р”). Таким образом, класс р2р (True) составляет $|S| + 1135 = 12\,461$; $FP = ((90 + 590) \cdot 100)/12\,461 = 5,5\%$; $FN = (408 \cdot 100)/12\,461 = 3,3\%$. “Полнота” = $((12\,735 - 680) \cdot 100)/12\,461 = 96,7\%$. “Точность” = $((12\,735 - 680) \cdot 100)/12\,735 = 94,6\%$. “Правильность” = $(16\,334 - 680 - 408)/16\,334 = 93,3\%$.

Из результатов проверки следует, что невыявленных р2р-адресов достаточно мало. Следует отметить, что большая их часть относится к протоколу DirectConnect, для которого основная часть трафика передается по ТСР. В данном случае использовался только протокол ТСР, клиент обращался к разным парам $\langle IP\text{-address}, 411 \rangle$ (порт по умолчанию для серверов, служащих для поиска файлов и источников скачивания), при этом использовались разные порты, поэтому не подошла ни одна из р2р-эвристик. В этом случае на основе номера порта можно классифицировать все адреса. Имеет смысл ввести дополнительное эвристическое предположение, предусматривающее данную ситуацию: если некоторый

IP-адрес взаимодействует только с парами $\langle IP\text{-address}, port \rangle$, где порт является портом по умолчанию для р2р-протокола, то это взаимодействие можно считать р2р.

Если сравнить результаты классификации потоков, то оценки меняются: $FP = 11\%$, $FN = 4,5\%$. Результаты классификации байтов трафика становятся еще менее достоверными. Это объясняется тем, что выявленный трафик BitTorrent представлен в основном “сигнальной” информацией, а трафик невыявленного Direct Connect и потоков, ложно классифицированных как р2р, представлен передачей данных. Тем не менее полнота составляет 90%, а правильность — 86%.

При увеличении интервала до 30 мин эксперимент показал, что значение FN увеличилось, например, за счет того, что при обращении к паре $\langle IP\text{-address}, port \rangle$ за этот период могут совпасть случайные номера портов, используемые различными участниками обмена. Величина FP уменьшилась за счет исключения из разряда р2р web- и dns-серверов, которые хорошо выявляются на большом интервале.

Заключение. В данной работе осуществлена идентификация р2р-трафика в случае, когда исходными данными являются готовые характеристики потоков формата Cisco NetFlow, экспортируемые сетевыми устройствами. К таким потокам сложно применить методы DM&ML, поскольку недоступны “тонкие” характеристики потоков, при этом невозможно использовать эвристические предположения, основанные на наблюдении за длиной пакетов.

В основу рассматриваемого принципа классификации положены эвристические предположения о поведении участников р2р-обмена, предложенные в [9]. Представлен модифицированный алгоритм пассивного анализа потоков IP-трафика, используемый при идентификации р2р-трафика. Алгоритм реализован на языке Perl и выполняется в среде FreeBSD.

Для определения эффективности алгоритма результаты сравнивались с результатами классификации, выполненной на основе поиска шаблонов в полном содержимом пакетов. Установлено, что приведенную методику целесообразно использовать при выявлении трафика часто используемого протокола BitTorrent. Что касается невыявленного трафика Direct Connect, то он качественно состоял из однотипного взаимодействия, что не позволило применить эвристики. Вопрос о том, какое минимальное количество и качество данных необходимо для идентификации того или иного протокола, остается открытым.

Для трафика Skype прежде всего необходимо адекватно определить шаблоны прикладного уровня. Эта задача требует дополнительного исследования с использованием испытательного стенда. Тем не менее все адреса, выявленные сигнатурным методом, идентифицированы рассматриваемым алгоритмом.

Список литературы

1. KARAGIANNIS T., BROIDO A., BROWNEE N., ET AL. File-sharing in the Internet: A characterization of p2p traffic in the backbone: Tech. report. Riverside, 2004. [Electron. resource]. <http://www.cs.ucr.edu/~tkarag/papers/tech.pdf>.
2. KIM M., KANG H., HONG J. Towards peer-to-peer traffic analysis using flows // Self-managing distributed systems: 14th IFIP/IEEE Intern. workshop on distributed systems: operations and management, Heidelberg (Germany), Oct. 20, 2003. Berlin: Springer LNCS, 2003. V. 2867. P. 55–67.
3. PORT NUMBERS. [Electron. resource]. <http://www.iana.org/assignments/port-numbes>.
4. ALOK MADHUKAR, CAREY WILLIAMSON. A longitudinal study of p2p traffic classification // Proc. of the 14th IEEE Intern. symp. on modeling, analysis, and simulation MASCOTS. Washington: S. n., 2006. P. 179–188.

5. NETFLOW services solutions guide. [Electron. resource]. <http://www.cisco.com/univercd/cc/td/doc/cisintwk/intsolns/netfslol/nfwhite.pdf>.
6. CLAFFY K. C., BRAUN H. W., POLYZOS G. C. A parameterizable methodology for Internet traffic flow profiling // IEEE J. Select. Areas Commun. 1995. V. 13, iss. 8. P. 1481–1494.
7. ЩЕРБАКОВА Н. Г. Вопросы идентификации p2p-трафика. Метод сигнатур // Проблемы оптимизации сложных систем: Тр. 5-й Междунар. азиат. шк.-семинара, Бишкек, 12–22 авг. 2009 г. Новосибирск: ИВМиМГ СО РАН, 2009. С. 158–169.
8. SEN S., SPATSCHECK O., WANG D. Accurate, scalable in-network identification of p2p traffic using application signatures // Proc. of the 13th Intern. world wide web conf., New York, May 2004. N. Y.: ACM, 2004. P. 512–521.
9. KARAGIANNIS T., BROIDO A., FALOUTSOS M., KC. CLAFFY. Transport layer identification of p2p traffic // Proc. of the 4th Workshop on Internet measurement (IMC). ACM SIGCOMM, Taormina (Italy), Oct. 25–27, 2004. N. Y.: ACM, 2004. P. 121–134.
10. KARAGIANNIS T., PAPAGIANNAKI K., FALOUTSOS M. BLINK: multilevel traffic classification in the dark // Proc. of the conf. on application technologies, architectures and protocols for computer communications, Philadelphia (USA), Aug. 21–26, 2005. N. Y.: ACM, 2005. P. 229–240.
11. WANCHAI NGIWLAY, CHALERMEK INTANAGONWIWAT, YUYONG TENG-AMNUAY. BitTorrent peer identification based on behaviors of a choke algorithm // Proc. of the 4th Asian conf. on Internet engineering, 2008. N. Y.: ACM, 2008. P. 65–74.
12. BITTORRENT Protocol Specification v1.0. [Electron. resource]. <http://wiki.theory.org/BitTorrentSpecification>.
13. WU C., CHEN K., CHANG Y., LEI C. Detecting peer-to-peer activity by signaling packet counting // Proc. of the conf. on application technologies, architectures and protocols for computer communications, Seattle (USA), Aug. 17–22, 2008. [Electron. resource]. http://mmnd.iis.sinica.edu.tw/wu08_signaling.pdf.
14. GUYON I., ELISSEEF A. An introduction to variable and feature selection // J. Machine Learning Res. 2003. V. 3. P. 1157–1182.
15. MOORE A. W., ZUEV D. Discriminators for use in flow-based classification: Tech. rep. RR-05-13. L.: Queen Mary Univ., 2005.
16. ERMAN J., ARLITT M., MAHANTI A. Traffic classification using clustering algorithms // SIGCOMM'06 Workshop, Pisa (Italy), Sept. 11–15, 2006. N. Y.: ACM, 2006. P. 281–286.
17. ZHAO J., HUANG, SUN Q., MA Y. Real-time feature selection in traffic classification // J. China Univ. Post Telecommun. 2008. Suppl. 15. P. 68–72. [Electron. resource]. <http://www.sciencedirect.com>.
18. LI W., CANINI M., MOORE A. W., BOLLA R. Efficient application identification and the temporal and spatial stability of classification schema // Computer Networks. 2009. V. 53, N 6. P. 790–809.

Бредихин Сергей Всеволодович — канд. техн. наук, зав. лабораторией Института вычислительной математики и математической геофизики СО РАН; e-mail: bred@nsc.ru;
Щербакова Наталья Григорьевна — ст. науч. сотр. Института вычислительной математики и математической геофизики СО РАН; e-mail: nata@nsc.ru

Дата поступления — 21.12.11 г.