ИССЛЕДОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ВОЗМОЖНОСТЕЙ ВНЕДРЕНИЯ СОРМ В SDN

В.С.Елагин[1], В.А. Сорокин[1]

[1]Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, Санкт-Петербург, 193232, Российская Федерация
Адрес для переписки: elagin.vas@gmail.com

Информация о статье
УДК 004.942
Язык статьи – русский

Ссылка для цитирования: Елагин В.С., Сорокин В.А. Исследование технологических возможностей CORM в SDN // Труды учебных заведений связи. 2017. Т. 3. № 1. С. 28-35.

Аннотация: *Современное бурное развитие информационных технологий требует гибкости и масштабируемости от сети. Традиционные сети не всегда способны эффективно реагировать на новые потребности пользователя. Одним из перспективных направлений модернизации существующей архитектуры сети является концепция SDN. В статье рассмотрена концепция построения SDN, нормативно-правовые требования к сетям общего пользования и текущие сложности, возникающие при реализации законного перехвата трафика в сетях SDN.*

Ключевые слова: законный перехват, программно-конфигурируемые сети, SDN, СОРМ, OpenFlow, DPI.

RESEARCH OF TECHNOLOGICAL POSSIBILITIES OF THE IMPLEMENTATION OF LAWFUL INTERCEPTION IN SDN

V.S. Elagin[1], V.A. Sorokin[1]

[1]The Bonch-Bruevich Saint-Petersburg State University of Telecommunication, St. Petersburg, 193232, Russian Federation

Article info
Article in Russian

For citation: Elagin V., Sorokin V. Research of technological possibilities of the implementation of lawful interception in SDN // Proceedings of educational institutes of communication. 2017. Vol. 3. Iss. 1. PP. 28-35.

Abstract: *Traditional networks are not always able to respond effectively to the new needs of the user. The existing network architecture requires modernization. SDN is one of the promising directions of development of information networks. The article discusses the concept of SDN networks, legal requirements for public networks and the current difficulties encountered in the implementation of lawful interception of traffic on networks SDN.*

Keywords*: lawful interception, software defined network, SDN, LI, OpenFlow, DPI.*

Современное бурное развитие информационных технологий требует гибкости и масштабируемости от сети. Количество подключённых устройств каждый год растёт в геометрической прогрессии, лишь устройств Интернета вещей к 2021 году ожидается до 46 млрд. штук. Увеличивается и объём генерируемого трафика. Традиционные сети не всегда способны эффективно реагировать на новые потребности пользователя. Требуется модернизация существующей архитектуры сети. Одним из перспективных направлений развития информационных сетей является SDN (от англ. *Software Defined Network*, программно-конфигурируемая сеть) – технология построения архитектуры сетей связи, основанная на принципе разделения функций управления и функций передачи. В традиционных маршрутизаторах и коммутаторах данные функции неотделимы друг от друга, и каждый элемент принимает решения самостоятельно и относительно независимо (рис. 1).

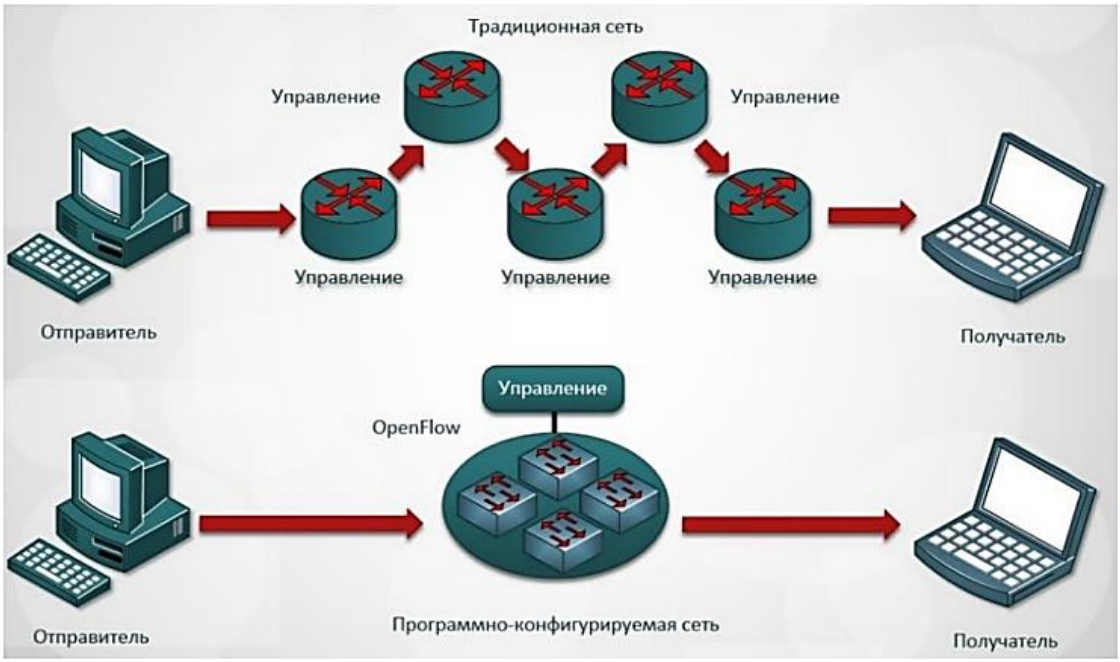

Рисунок 1. Управление сетью в традиционной сети и сети SDN

Концепция SDN предполагает вынести функции управления в отдельное устройство – контроллер, а коммутаторам оставить только функцию передачи трафика. Вся логика управления сетевыми устройствами в SDN реализуется программным способом, что позволяют разработчикам создавать собственные приложения для управления сетью через программные интерфейсы (API) контроллера.

Взаимодействие между контролером и коммутаторами осуществляется посредством протокола управления. Первым открытым стандартизированным протоколом является *OpenFlow*. Управление данными в *OpenFlow* осуществляется на уровне потоков, а не отдельных пакетов. Правила в коммутаторе уста-

навливаются с участием контроллера только для первого пакета, потом все последующие пакеты потока его используют. Стандарт *OpenFlow* принят большинством производителей сетевого оборудования, и уже доступны *OpenFlow*-решения на рынке сетевого оборудования. Существуют и проприетарные решения протоколов управления, но в связи с их закрытой документации нет возможности изучить их.

Преимуществом перехода к концепции SDN является:

1) Снижение капитальных затрат. Уменьшение выполняемых задач коммутаторами позволят существенно снизить их стоимость.

2) Централизованное управление. К контроллеру подключены все коммутаторы в сети, соответственно упрощается эксплуатация сети и он будет иметь представление о состоянии сети, что позволит производить балансировку нагрузки.

3) Гибкость. Используя программные интерфейсы контролера можно разрабатывать и внедрять новые услуги или сервисы без необходимости вносить изменений в конфигурацию сети.

4) Автоматизация. Возможность производить конфигурацию, предоставление новых услуг используя готовые алгоритмы.

Одним из нерешенных вопросов на сетях SDN является реализация функций законного перехвата, осуществляемого СОРМ (сокр. от Система технических средств для обеспечения функций Оперативно-Розыскных Мероприятий). Согласно приказу № 83 Министерства связи и массовых коммуникаций РФ от 16-го апреля 2014 года (далее – Приказ), все операторы сетей передачи данных до 31 марта 2015 года должны привести используемое, а также вводимое в эксплуатацию оборудование коммутации и маршрутизации пакетов информации в соответствие с требованиями, указанными в Приказе [1].

Как говорилось ранее, концепция SDN предполагает реализацию функции управления сетью и сложных сетевых функций на контроллере, поэтому логично было бы внедрять функции СОРМ на контроллере в виде сетевого приложения и получать требуемые приказом идентификаторы через API. Реализация системы законного перехвата трафика на контроллере позволит централизованно следить и собирать необходимую информацию в одном узле сети.

Для рассмотрения возможности данной реализации был проведён сравнительный анализ параметров, установленных Приказом, с возможными классификаторами, определенными в потоковых таблицах *OpenFlow*-коммутаторов (см. табл.). Из произведённого сопоставления можно сделать вывод о невозможности реализации СОРМ в контроллере ПКС-сети из-за нехватки полей соответствия в классификаторах протокола *OpenFlow*.

Для практического подтверждения полученного вывода была развёрнута виртуальная SDN-сеть. В качестве компьютерной сети использовался эмулятор *mininet*, в качестве контролера использовалась сетевая операционная система-*Floodlight*.

Таблица. Сопоставление параметров контроля и полей соответствия в классификаторах *OpenFlow*

№ п/п	Параметр контроля	Наличие поля соответствия в классификаторах *Openflow*
1	Постоянный IP-адрес (IPv4, IPv6)	Да
2	IP-адреса, определяемые по маске	Да
3	Имя учетной записи пользователя, используемое для идентификации пользователя услуг связи при доступе к сети передачи данных	Нет
4	Электронный почтовый адрес для всех почтовых сервисов с применением протоколов SMTP, POP3, IMAP4, не использующих средства защиты информации, включая криптографические	Нет
5	Электронный почтовый адрес сервисов, не использующих средства защиты информации, включая криптографические	Нет
6	Телефонный номер пользователя (вызываемого и/или вызывающего)	Нет
7	Идентификатор абонентской телефонной линии, используемый для идентификации пользователя услуг связи при доступе к сети передачи данных и телематическим услугам связи	Нет
8	Идентификатор вызываемого и вызывающего пользователя услуг связи по передаче данных для целей передачи голосовой информации	Нет
9	Международный идентификатор абонента сети подвижной связи (IMSI)	Нет
10	Международный идентификатор мобильного оборудования (IMEI)	Нет
11	Уникальный идентификатор оборудования сетей передачи данных (MAC-адрес)	Да
12	Идентификатор служб обмена сообщениями, включая ICQ	Нет
13	Мобильный идентификационный номер мобильной абонентской радиостанции (MIN)	Нет

Схематично изобразим топологию созданной сети (рис. 2), все узлы с подключениями и введем некоторые условные обозначения:

1) h1..h9 – хосты, подключенные к *OpenFlow*-коммутаторам;
2) eth0..eth4 – интерфейсы подключения (*Ethernet*);
3) s1..s4 – *OpenFlow*-коммутаторы;
4) c0 – контроллер с сетевой операционной системой *Floodlight*.

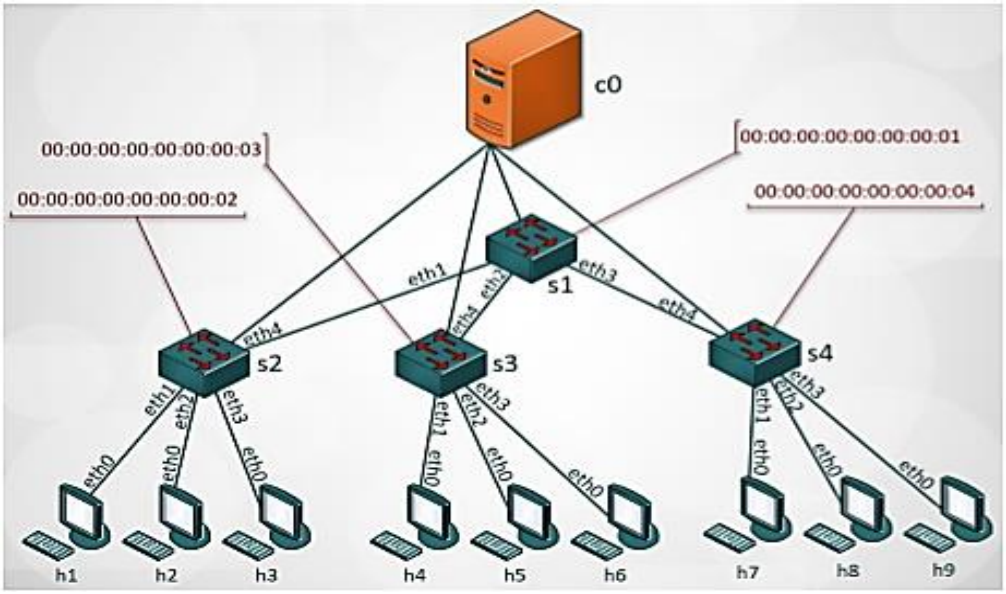

Рисунок 2. Архитектура сети эксперимента

Целью проводимого эксперимента являлось осуществление проверки на достаточность получаемой информации контроллером для реализации СОРМ. Большая часть параметров контроля установленные приказом переносятся пакетами высокоуровневых протоколов. В качестве такого протокола будут использоваться пакеты протокола прикладного уровня передачи данных – HTTP (от англ. *HyberText Transfer Protocol*, протокол передачи гипертекста).

Для генерации *HTTP*-трафика между хостами достаточно запустить на одном из хостов простейший веб-сервер, а с другого хоста выполнить запрос к данному серверу. В рамках эксперимента был запущен на «h2» веб-сервер, написанный на языке программирования *Python*, а осуществление HTTP-запроса к серверу будем делать средствами консольной команды *wget* с хоста «h8». Для мониторинга *OpenFlow* пакетов на интерфейсах коммутаторов (рис. 3) и контроллера (рис. 4) использовался сетевой анализатор трафика *wireshark*.

No.	Time	Source	Destination	Protocol	Length	Info
22	1.012738	10.0.0.8	10.0.0.2	HTTP	176	GET / HTTP/1.1
31	1.313622	10.0.0.2	10.0.0.8	HTTP	1251	HTTP/1.0 200 OK (text/html)

Рисунок 3. Мониторинг *OpenFlow*-пакетов на интерфейсах коммутаторов

Рассмотрим подробнее *OpenFlow*-обмен для изучения получаемой информацией контроллером. В связи с отсутствием записей в таблице потоков коммутатора «s2» для поступившего на него HTTP-пакета (№ 22 на рис. 3), коммутатор информирует контроллер о новом поступившем пакете с помощью сообщения *packet_in* (№ 506 на рис. 4), который содержит копию HTTP-запроса пользовательского трафика до транспортного уровня модели OSI (рис. 5).

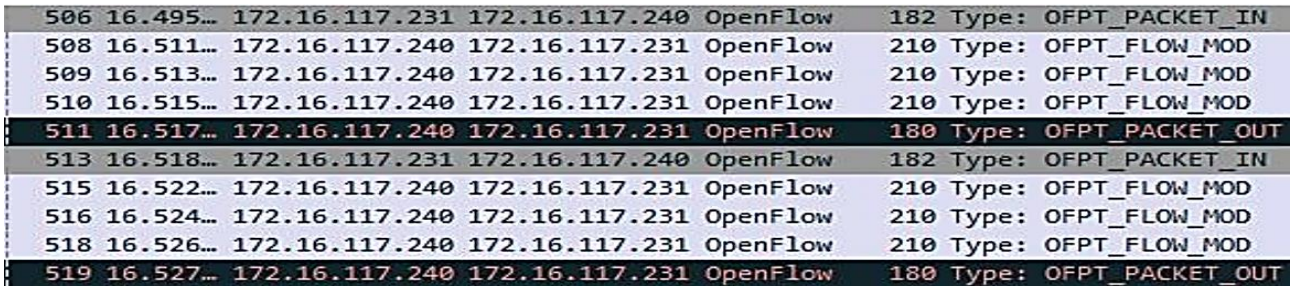

Рисунок 4. Мониторинг *OpenFlow*-пакетов на интерфейсах контроллера

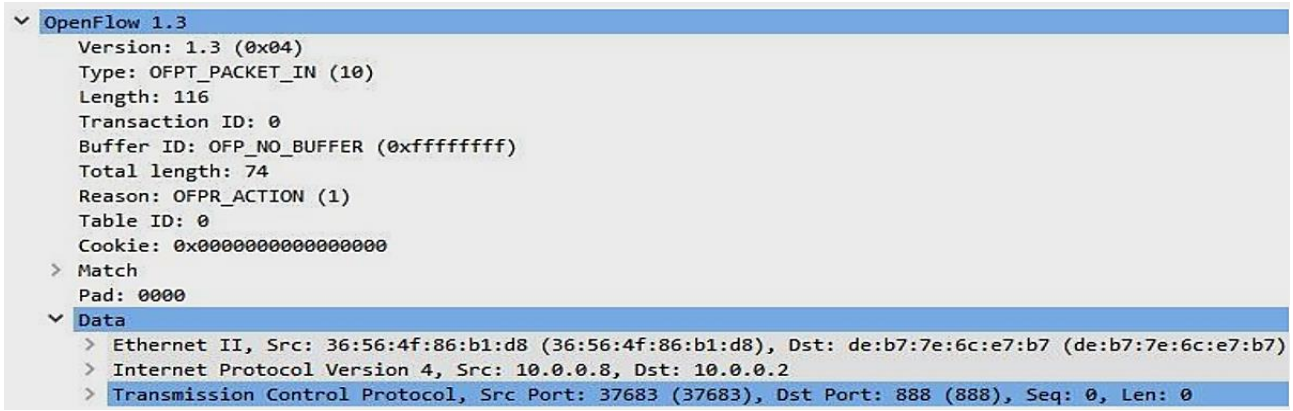

Рисунок 5. Содержимое *packet_in*-пакета, полученного контроллером

Контроллер, на основе полученных данных в пакете, добавляет данные о потоке в таблицы потоков коммутаторов «s2», «s1», «s4» (пакеты №№ 3094-3096 на рис. 4). После назначения маршрута контроллер указывает коммутатору «s2», с помощью пакета *packet_out* (№ 511 на рис. 4) отправить полученный пакет на «s1» по интерфейсу «eth4». Получив инструкции, коммутатор «s2» отправляет пакет коммутатору «s1», последний коммутатору «s4». Хост «h2», получив HTTP-запрос, формирует ответ, который будет отправлен по аналогично полученной инструкции.

Результаты сопоставления параметров и проведенного эксперимента говорят о том, что контроллер, благодаря полям соответствия классификаторов протокола *OpenFlow*, имеет возможность получать информацию о потоках (пакетах) включительно до транспортного уровня, чего недостаточно для интеграции СОРМ в контроллер SDN сети, так как некоторые параметры контроля передаются с помощью высокоуровневых протоколов. Таким образом, если реализовать систему законного перехвата трафика в контроллере без внесения изменений в имеющиеся классификаторы *OpenFlow* протокола, то СОРМ будет реализован неполноценно и трафик с параметрами передаваемые в высокоуровневых протоколах, не будут обнаруживаться [2].

В качестве альтернативного варианта, рассмотрим модель реализации СОРМ на уровне коммутаторов SDN (рис. 6).

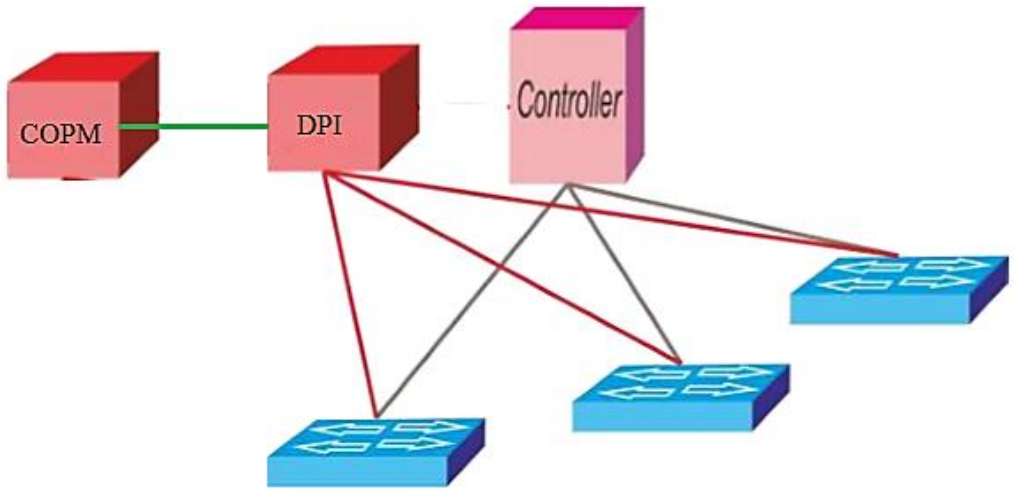

Рисунок 6. Схема подключения при реализации модели СОРМ на уровне коммутаторов

Идея данной модели будет заключаться в «пассивном» режиме съёма трафика путём настройки «зеркалирования» трафика коммутаторов на специальный узел связи, где будет происходить дальнейший его анализ. Для получения идентификаторов высокоуровневых протоколов потребуется реализовывать технологию DPI [3]. Суть технологии DPI (от англ. *Deep Packet Inspection*, технология накопления статистических данных, проверки и фильтрации сетевых пакетов по их содержимому) заключается в глубоком анализе пакета на верхних уровнях модели OSI. Анализ и идентификация протокола/приложения может осуществляться не только по формату заголовков, номерам портов и т.п., но и на базе поведенческого анализа трафика.

Технология DPI позволяет анализировать пакеты не только на нижних трех уровнях модели OSI – а это MAC-, IP- и TCP-заголовки, но и до седьмого уровня включительно. Как раз таки самой главной особенностью технологии DPI и является ее возможность анализировать прикладной уровень, что, как правило, является очень ресурсоемким процессом.

Система DPI, как правило, устанавливается на границе сети оператора в разрыв существующих каналов на пограничных маршрутизаторах. Тем самым, весь трафик, который покидает или входит в сеть оператора, проходит через DPI, что даёт возможность его мониторинга и контроля.

Реализация DPI в SDN позволит на основе анализа пакетов получать и своевременно обнаруживать параметры контроля, передаваемые с помощью высокоуровневых протоколов (не использующие алгоритмы шифрования). Внедрение технологии DPI непосредственно на коммутаторы будет требовать больших мощностей и производительности от коммутаторов, приводя тем самым к увеличению стоимости сетевого оборудования, лишая концепцию SDN одного из свойств – преимуществ. Поэтому было решено вынести функционал DPI в отдельный узел. Для осуществления запросов на основе параметров контроля СОРМ будет подключён к узлу с DPI напрямую.

У каждой из рассмотренных моделей реализаций имеются свои преимущества и недостатки. Внедрение функции СОРМ на контроллере в виде сетевого приложения требует добавление и отслеживание классификаторов высокоуровневых протоколов на коммутаторах, что приведёт к их усложнению, либо производить на коммутаторах полную инкапсуляцию приходящего на пакета и отправлять его на контроллер, этот вариант приведёт к увеличению нагрузки на служебный канал. Модель реализации СОРМ на уровне коммутаторов имеет шанс на существование, но требует большого капиталовложения в оборудование, усложнения схемы и дальнейшего исследования данной реализации. Об окончательной модели реализации СОРМ говорить пока рано ввиду продолжающегося становления и развития самой концепции SDN. Однако уже существуют готовые решения для организации сети на базе данной концепции, и можно предположить их практическое внедрение в ближайшее время; поэтому вопрос организации СОРМ на программно-конфигурируемых сетях является своевременным и важным.

Список используемых источников

1. Приказ Минкомсвязи РФ от 16 апреля 2014 года № 83 «Об утверждении Правил применения оборудования систем коммутации, включая программное обеспечение, обеспечивающего выполнение установленных действий при проведении оперативно-розыскных мероприятий. Часть III. Правила применения оборудования коммутации и маршрутизации пакетов информации сетей передачи данных, включая программное обеспечение, обеспечивающего выполнение установленных действий при проведении оперативно-розыскных мероприятий» // Российская газета. 18 июля 2014 г.

2. Елагин В.С. Подходы к моделированию систем законного перехвата трафика в SDN // Актуальные проблемы инфотелекоммуникаций в науке и образовании: материалы V международной научно-технической и научно-методической конференции. 2016. С. 353- 358.

3. Высоцкий С.А., Пряжников В.С., Сорокин В.А. Исследование возможности реализации и дальнейшего применения DPI на SDN сетях // 70 региональная научно-техническая конференция студентов, аспирантов и молодых ученых. Студенческая весна 2016. С. 100-104. URL: http://www.sut.ru/doci/nauka/70rntk.pdf.