Список используемых источников

1. Агравал Г. Нелинейная волоконная оптика: пер. с англ. / Под ред. П. В. Мамышева. М.: Мир, 1996. 324 с.
2. Листвин А.В., Листвин В.Н., Швырков Д.В. Оптические волокна для линий связи. М.: ЛЕСАРарт, 2003. 106 с. ISBN 5-902367-01-8.
3. Леонов А., Наний О., Трещиков В. Нелинейные искажения и нелинейный шум в когерентных сетях связи // Первая миля. 2014. № 4. С. 51–55.

ТЕХНОЛОГИЯ ГЛУБОКОЙ ИНСПЕКЦИИ ПАКЕТОВ В ПРОГРАММНО-КОНФИГУРИРУЕМОЙ СЕТИ

В.С. Елагин, А.В. Онуфриенко

Программно-конфигурируемые сети приобретают все большую популярность в связи с возможностью решения проблем независимости от производителя оборудования, снижения стоимости оборудования, упрощения администрирования сетью, однако при передаче данных возникает потребность в технологии глубокой инспекции пакетов.

Ключевые слова: программно-конфигурируемая сеть, глубокая инспекция пакетов, DPI, Mininet.

TECHNOLOGY OF DEEP PACKET INSPECTION IN SOFTWARE-DEFINED NETWORKING

Elagin V., Onufrienko A.

Software-defined networking are becoming increasingly popular due to the ability to solve the problems of independence from the equipment manufacturer, to reduce the cost of equipment, simplifying administration network, but the data transfer there is a need in the deep packet inspection technology.

Keywords: software-defined networking, deep packet inspection, DPI, Mininet.

Введение

Сегодня наиболее актуальным для операторов стало обеспечение видимости, того какой трафик проходит через сеть, для обнаружения изменения поведения пользователей на раннем этапе.

До сих пор в центре внимания были канальный и сетевой уровни, а не транспортный, представительный и уровень приложений. Тем не менее, нет никаких сомнений в том, что уровни 4–7 станут основным направлением для будущего изучения. Ведь, если в программно-конфигурируемых сетях физическое оборудование будет уходить на второй план, то потребности приложений и абонентов переместятся на первый план. Предполагается, что сами приложения будут запрашивать определенные возможности в сети, и в связи с этим будут формироваться определенные сетевые ресурсы. Политики для каждого приложения будут установлены в момент разработки приложения

и перенесены на платформу, как услуга, и будут охватывать, например, производительность, требования соответствия уровня надежности. В данном вопросе DPI является технологией, которая как раз справится с этими вопросами: она определяет конкретные приложения в режиме реального времени при определенных ключевых узловых точках, применяет политику. Политики могут включать в себя блокирование, оптимизации, определения приоритетов [1].

Так же система глубокого анализа может предоставить подробные данные, для информирования контроллера о состоянии сети и потоках ее трафика. Это позволяет рассматривать сеть как целостный ресурс, а не различные группы устройств. В конечном счете, внедрение DPI-систем позволит применять политики контроля и автоматизации для всей сети в целом.

Экспериментальная часть

Для проведения исследования была использована среда Mininet. Mininet близко эмулирует реальную физическую сеть, путем воссоздания операционного окружения.

Развернуть простую сеть Openflow в эмуляторе Mininet можно в консоли локальной машины с использованием удаленного контроллера на виртуальной машине (графический интерфейс показан на рис. 1):

$ sudo mn –controller=remote, ip=172.16.117.231, port=6653 –switch ovsk, protocols=OpenFlow13 –topo=tree, depth=1, fanout=2,

где *mn* – это главный исполняемый файл Mininet. Ключ *topo* создает топологию (дерево), опцией *switch* указывается тип коммутатора (*ovsk* – встроенный). Тип контроллера – удаленный (*controller remote*).

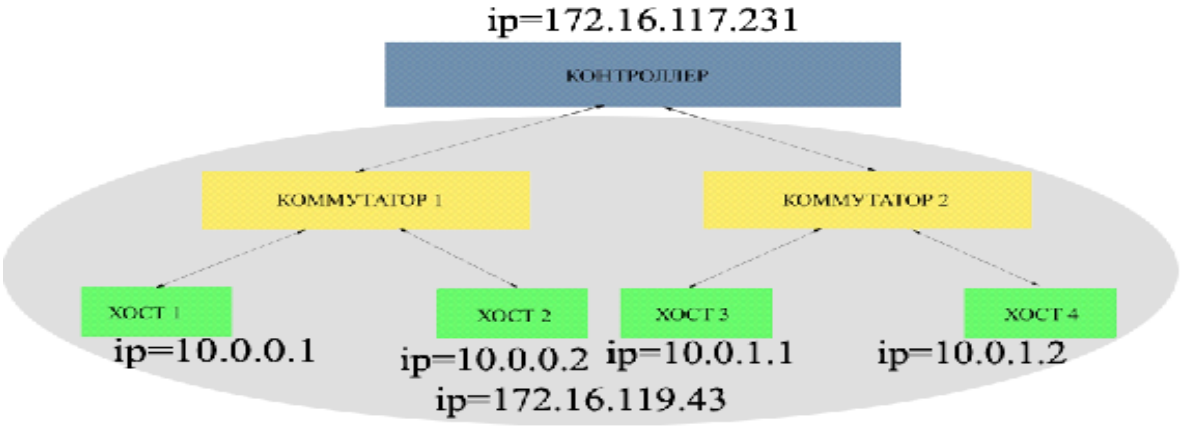

Рис. 1. Схема, развернутая в Mininet

В программно-конфигурируемых сетях для связи контроллера с коммутатором используются сообщения протокола OpenFlow, в то время как внутри самой сети используются различные протоколы [2]. Важным является детальное рассмотрение верхних уровней в модели *OSI*, а не физического, канального

и сетевого. Рассмотрим на примере, указанном на рисунке 2, реакцию сети на протоколы прикладного уровня при взаимодействии между объектами нашей сети.

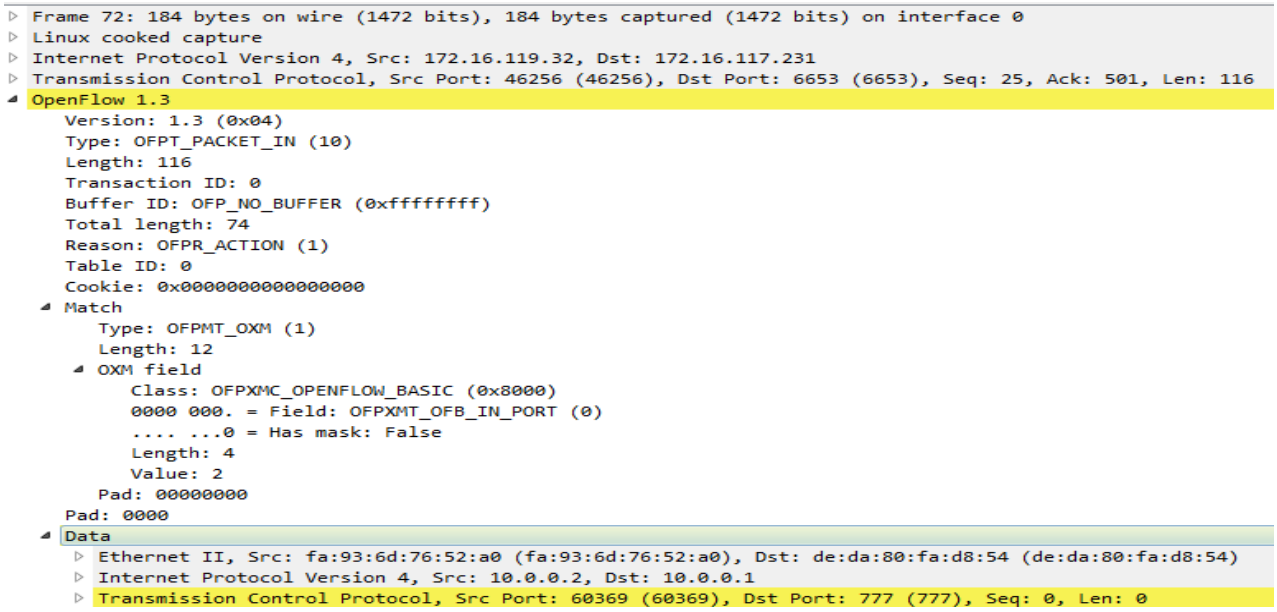

Рис. 2. Сообщение протокола OpenFlow при работе с протоколом HTTP

Можно заметить, что в сообщениях OpenFlow был виден трафик транспортного уровня, а трафик прикладного уровня при передаче HTTP пакетов в сообщениях OpenFlow не может быть распознан. Чтобы устранить этот эффект, можно использовать системы глубокого анализа, как инструмент статистического анализа.

Благодаря системам глубокого анализа оператор связи может узнать, как используются ресурсы сети и при необходимости применить требуемые политики. Статистика, собираемая системой, может помочь своевременно обнаруживать проблемы, знать какой тип трафика преобладает на сети, выявить самых пользователей, наиболее посещаемые пользователями сайты [3].

Рассмотрим особенности размещения DPI систем на разных уровнях:

1) Уровень приложений.

С относительной легкостью, программное обеспечение DPI может быть встроено на уровне бизнес-приложений. Тем не менее, некоторым приложениям может потребоваться минимизированное влияние узких мест, созданных длительным путем передачи данных. Учитывая возможность задержек, это развертывание DPI лучше всего работает для не критичных ко времени использования сетевых приложений, таких как аналитика.

2) Уровень управления.

Программное обеспечение DPI может быть развернуто в контроллере. Тем не менее, часть неопознанного коммутаторами трафика должна быть отправлена на DPI-систему для опознания, что может привести к проблемам масштабируемости и производительности. После этого все последующие потоки одного

и того же типа не отправляются на коммутатор, что может привести к тому, что будут пропущены «опасные» пакеты.

3) Уровень инфраструктуры.

Сетевые устройства тоже могут запустить программное обеспечение DPI, и после идентификации приложений и метаданных, они могут либо применять предварительно определенную политику, либо отправить эту информацию в контроллер, а затем получить обратно политику или правило.

Реализации DPI в слое узла минимизирует задержку. Однако этот подход является дорогостоящим, поскольку он требует наибольшее количество экземпляров в сети.

Реализация системы DPI на коммутаторе очень выгодна в том случае, если у нас есть жесткое требование к ограничению какого-либо трафика, так как абсолютно все пакеты пройдут через систему и будут подвергнуты тщательной проверке. К сожалению, такое воздействие повлияет на производительность сети, и привести к задержкам.

Рассмотрим этот вариант подробнее (рис. 3). В качестве DPI-системы была использована утилита для создания и работы с сетевыми пакетами Scapy.

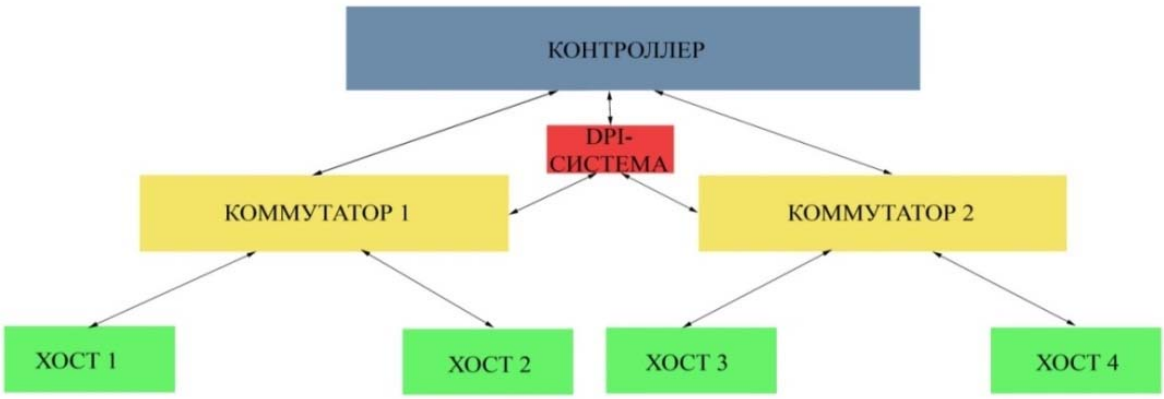

Рис. 3. Схема программно-конфигурируемой сети с внедрением DPI-системы

В ходе исследования пакетов OpenFlow была выделена постоянная часть пакета, которую и решено было взять как сигнатуру протокола OpenFlow.

Благодаря централизованному построению маршрутов и контролю всего сетевого трафика подозрительный поток при необходимости может быть отправлен на существующие средства DPI для анализа.

Результаты

При исследовании взаимодействия системы, были вычислены задержки при обработке пакетов OpenFlow, которые были занесены на график распределений задержек, указанный на рисунке 4.

Для анализа эксперимента было выбрано 10 416 значений.

Максимальное время обработки пакета составило 14 424 мкс.

Минимальное время обработки – 210 мкс.

Среднее время обработки – 439,173675 мкс.
Мера разброса значений времени обработки пакета – 77 186,36 мкс

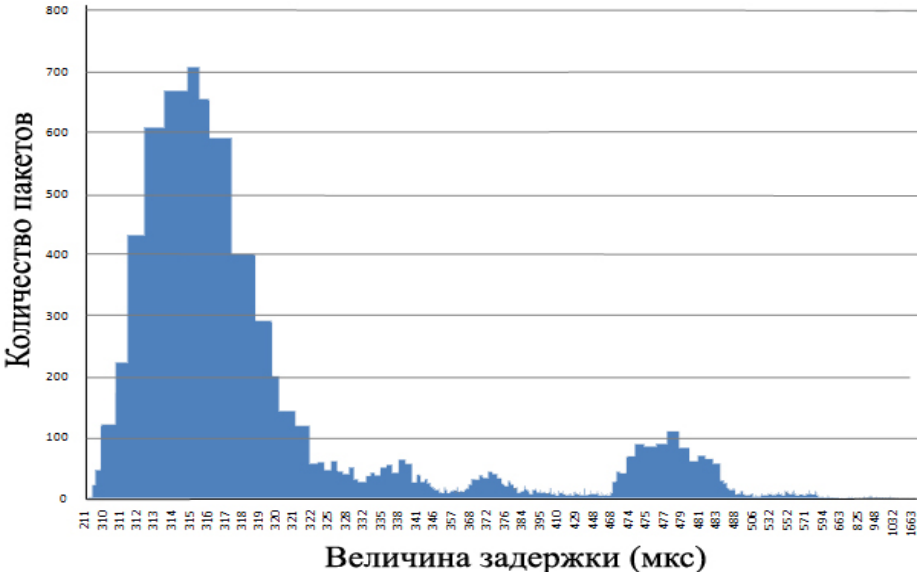

Рис. 4. График расределения задержек

На графике ярко выделяются 2 экстремума; необходимо дальнейшее исследование, чтобы оценить (не)зависимость второго экстремума от оборудования, на котором проводился эксперимент.

Заключение

В результате проведённого анализа можно сделать вывод, что внедрение DPI-системы является необходимым действием для наилучшего функционирования сети, улучшения сетевых характеристик путем контролирования трафика, а так же для внедрения новых услуг. Однако из-за появления задержек, необходимо перед установкой системы провести дополнительное исследование сети и применять DPI-систему, которая будет соответствовать требованиям данной сети. Так же применение такой системы в настоящее время может сдерживаться следующими факторами: нарушением неприкосновенности частной жизни, применением не корректных политик, а так же препятствие заключается в дороговизне приобретения и установке DPI-системы.

Список используемых источников

1. Елагин В.С. СОРМ в сетях пост-NGN. Модели и технологии [Электронный ресурс] // Вестник связи. 2015. № 06. С. 47–49. URL: http://files.iks.sut.ru/publications/2015-010-pp.pdf (дата обращения 21.05.2016).

2. Программно-конфигурируемые сети [Электронный ресурс] // Открытые системы. СУБД. 2012. № 09. URL: http://www.osp.ru/os/2012/09/13032491/ (дата обращения 20.05.2016).

3. Зачем нужен DPI [Электронный ресурс] // Telekomza [сайт]. URL: http://telekomza.ru/2012/03/26/zachem-nam-nuzhen-dpi/ (дата обращения 20.05.2016).