УДК 004.7, 004.732

ПРОГРАММНО-КОНФИГУРИРУЕМЫЕ СЕТИ: OPENFLOW И ВИРТУАЛЬНЫЕ СЕТЕВЫЕ ПЕРЕКРЫТИЯ

Чугреев Д.А.[1], Шкребец А.Е.[1], Шевель А.Е.[1], Власов Д.В.[1], Грудинин В.А.[1], Каирканов А.Б.[1], Садов О.Л.[1], Титов В.Б.[1], Хоружников С.Э.[1], Сомс Л.Н.[1]

[1]Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики» (НИУ ИТМО), Санкт-Петербург, Россия (197101, г. Санкт-Петербург, Кронверкский проспект, д. 49), xse@vuztc.ru

Анализируются тенденции развития программно-конфигурируемых сетей (ПКС). Сравниваются достоинства и недостатки наиболее популярных подходов: на базе протокола OpenFlow и с использованием виртуальных сетевых перекрытий. Отмечается необходимость разделения сетевых сервисов и физической инфраструктуры. Рассматриваются существующие технические решения с применением ПКС. Среди них архитектурные и аппаратные предпочтения JuniperNetworks, платформа виртуализации VSP компании NuageNetworks, интегрированное решение VMwareNSX и WindowsNetworkVirtualization от Microsoft. Отмечается тенденция внедрения ПКС на базе существующей инфраструктуры, без замены сетевого оборудования и с сохранением существующих сервисов. При этом основные функции систем реализуются в виртуальных программных коммутаторах, непосредственно под управлением гипервизоров. Делается вывод о том, что в настоящее время протокол OpenFlow находит практическое применение лишь в качестве протокола управления виртуальными коммутаторами, а наиболее популярными являются решения на базе виртуальных сетевых перекрытий.

Ключевые слова: программно-конфигурируемые сети (ПКС), OpenFlow, сетевые перекрытия.

SOFTWARE DEFINED NETWORKS: OPENFLOW AND VIRTUAL NETWORK OVERLAYS

Chugreev D.A.[1], Shkrebets A.E.[1], Shevel A.E.[1], Vlasov D.V.[1], Grudinin V.A.[1], Kairkanov A.B.[1], Sadov O.L.[1], Titov V.B.[1], Khoruzhnikov S.E.[1], Soms L.N.[1]

[1]National Research University of Information Technologies, Mechanics and Optics, Saint-Petersburg, Russia (197101, Saint-Petersburg, Kronverkskiy pr., 49), xse@vuztc.ru

A briefdescription of the main trends in the development of Software-defined networks (SDN) is given. We compare the advantages and disadvantages of the most popular approaches: end-to-end OpenFlow and virtual network overlays. We are describing existing SDN solutions. Among them, architectural and hardware preferences of Juniper Networks, the Virtualized Services Platform (VSP) from Nuage Networks, an integrated VMware/Nicira solution NSX and Windows Network Virtualization from Microsoft. Most of them tend to implement SDN on the existing infrastructure without having to replace network equipment, maintaining existing services. Main functions of the systems are implemented in software virtual switches that are directly managed by the hypervisor. It is concluded that at the present time OpenFlow has practical application only as a control protocol for virtual switches and most solutions are based on the virtual network overlays.

Keywords: Software-defined networks (SDN), OpenFlow, network overlays.

Рассматриваются решения ведущих производителей сетевого оборудования и программного обеспечения для ЦОД в области программно-конфигурируемых сетей. Делается вывод о том, что технологии сетевых перекрытий находят наибольшее практическое применение.

Компьютерные сети являются стратегическим ресурсом, однако использованию их потенциала в полном объеме препятствуют чрезмерная сложность управления, недостаточная для адаптации к потребностям приложений гибкость, что препятствует

внедрению инноваций. Широкое распространение облачных технологий и виртуализация вычислений создает сложности для сетевой инфраструктуры, заставляет справляться с динамически распределяемой нагрузкой и поддержкой большого количества пользователей. Программно-конфигурируемые сети (ПКС) рассматриваются в качестве технологии, способной не только решить существующие проблемы, но и стать основой для дальнейшего развития.

При построении ПКС на основе протокола OpenFlow предполагается, что первый пакет каждого потока направляется на контроллер, который прокладывает маршрут по всей сети и прописывает его в таблицу продвижения каждого коммутатора, не только на периметре, но и в ядре. Такой подход не является достаточно масштабируемым. Главной проблемой является резкое увеличение количества информации о состоянии продвижения пакетов (forwardingstateexplosion). Контроллер ПКС должен управлять каждым коммутатором при появлении нового потока, что вносит дополнительные задержки и снижает надежность. Для связи контроллера с коммутаторами необходима выделенная (out-of-band) сеть, которая должна быть построена по традиционной схеме без применения OpenFlow. Использование предварительного занесения некоторых маршрутов в таблицы управления потоками не решает проблему полностью. При попытке применить такой подход на практике возникают и дополнительные сложности: ограниченное количество моделей коммутаторов с поддержкой OpenFlow, неполная реализация стандарта, невозможность пользователей обновлять парк оборудования. Все это привело к переносу внимания на программные коммутаторы, работающие под управлением гипервизоров и взаимодействующие при помощи туннелей сквозь ядро сети.

Для эффективного использования сетевого оборудования в ЦОД с высокой степенью виртуализации необходимо разделение сетевых сервисов и физической инфраструктуры, создание виртуальных сетей, функции которых охватывали бы весь спектр операций, от базового установления соединений до механизмов безопасности, оптимизации протоколов передачи данных, обеспечения качества обслуживания. Построение полностью виртуальных сетей возможно только на основе сетевых перекрытий (networkoverlays) – логических каналов, созданных поверх существующей сети и связывающих узлы, входящие в сервисный домен. Сетевые перекрытия должны обеспечивать масштабируемость, изоляцию, упрощение и автоматизацию всех операций в динамических многопользовательских окружениях. Сетевые перекрытия включают в себя три основных компонента: логическое представление сети, плоскость управления и распределения адресной информации, протоколы туннелирования и форматы инкапсуляции. Модульный дизайн предусматривает независимость этих уровней друг от друга. В настоящее время применяются три основных

варианта создания сетевых перекрытий: VXLAN (разработан VMware и Cisco), NVGRE (Microsoft, Intel и Dell) и STT (VMware/Nicira). Все они предусматривают многопользовательскую поддержку и эффективную доставку инкапсулированных L2 пакетов. Компания IBM активно развивает свое решение DOVE (DistributedOverlayVirtualEthernet) [5].

Преимуществами сетевых перекрытий являются сокращение размера таблиц управления потоками в физических коммутаторах, отсутствие необходимости хранить адреса виртуальных машин. Добавление нового пользователя, новой виртуальной машины, применение новой политики требует взаимодействия только с коммутаторами на периметре сети, что повышает устойчивость физической сети. Перекрытия могут применяться в существующей сети с сохранением действующих сервисов без необходимости замены оборудования. К недостаткам перекрытий можно отнести усложнение поиска неисправностей и трудоемкость обеспечения качества обслуживания для индивидуальных пользователей. При этом сетевые перекрытия могут использовать OpenFlow в качестве протокола управления, а OF-Config для организации туннелей.

Практическую ценность развития сетевых перекрытий подтверждают ведущие производители сетевого оборудования и ПО для ЦОД.

Для внедрения технологий ПКС JuniperNetworks приобрела контроллер у компании Contrail. Несмотря на реализацию протокола OpenFlow в коммутаторах и маршрутизаторах компании, эта технология не рассматривается в качестве основной [1]. Новые специализированные микросхемы (ASIC) от Juniper, применяемые в серии EX9200, предусматривают возможность обработки инкапсулированной информации: извлечение идентификаторов виртуальной сети и использование их при управлении очередями коммутатора с ведением независимой статистики по каждому из них. Тем самым решается проблема обеспечения качества обслуживания для каждого пользователя с сохранением изоляции трафика. Для создания туннелей Juniper поддерживает протоколы MPLS-over-GRE и VXLAN. Выбор VXLAN обусловлен тем, что он считается де-факто стандартом для сетевых перекрытий в ЦОД и поддерживает множественные пути (multipathing). Применение MPLS упрощает взаимодействие с сетями провайдеров и позволяет использовать не только L2, но и L3 перекрытия. В качестве управляющего протокола для обмена информацией о маршрутах и принадлежности к VPN выбран XMPP, который, по мнению Juniper, имеет преимущество над OpenFlow за счет использования более высокого уровня абстракции (маршруты вместо потоков). Для взаимодействия распределенных контроллеров в кластере и маршрутизирующих шлюзов применяется BGP.

В апреле 2013 года NuageNetworks (дочерняя компания Alcatel-Lucent) представила свою платформу виртуализации сетей VirtualizedServicesPlatform (VSP) [4]. Основными преимуществами платформы является использование любой сетевой инфраструктуры (уровни виртуализации L2-L4), широкие возможности масштабирования и безболезненная интеграция с MPLS для взаимодействия с частными облачными сервисами и удаленными площадками. Платформа предназначена для автоматизации создания многопользовательских сетей в ЦОД. Она состоит из трех основных компонентов. Все управление и поддержание виртуальных представлений сети для каждого пользователя выполняется контроллером VSC (VirtualizedServicesController). Для задания политик управления, описания сервисов и аналитики служит каталог VSD (VirtualizedServicesDirectory). Взаимодействие с сетевыми сервисами осуществляется при помощи агента VRS (VirtualRouting&Switching). Это расширенный вариант программного коммутатора OpenvSwitch. Для создания туннелей между гипервизорами применяется VXLAN. Проблема широковещательного трафика решается за счет применения MP-BGP для распределения информации о доступности MAC- и IP-адресов. Каждый VRS заранее получает всю информацию, необходимую для продвижения пакетов. Для связи с маршрутизаторами на периметре сети использован MPLS-over-GRE. Агент VRS следит за изменениями в окружении, за появлением, удалением или перемещением виртуальных машин и реагирует на эти события в соответствии с заданной политикой, внося соответствующие изменения в сетевую конфигурацию для удовлетворения потребностей приложений. Детектирование изменений не зависит от платформы управления, в равной степени поддерживаются CloudStack, OpenStack, vCloudDirector, SCVMM. Расширения VRS работают со всеми основными гипервизорами: KVM, XEN, Hyper-V и ESXi. Для интеграции физических серверов и промежуточных устройств обработки используется шлюз VRS-G (VRSGateway). При этом VSD управляется при помощи RESTAPI, взаимодействие между VSD и VSC выполняется по протоколу XMPP, распределение информации о топологии между контроллерами в кластере по BGP, а для переноса информации о продвижении в коммутаторы применяется OpenFlow. Для доступа к виртуализованному ЦОД извне используются традиционные L2/L3 VPN.

Выступая в мае 2013 года в Стэнфордском университете (StanfordUniversity) один из авторов подхода ПКС Скот Шенкер (ScottShenker), подводя итоги пяти лет развития технологии, выделил четыре положения, которые оказались ошибочными [7]. Изначально предполагалось, что управляющая программа должна самостоятельно выполнять конфигурирование всех коммутаторов, коммутаторы функционально однородны и играют одну и ту же роль, сеть состоит из аппаратных коммутаторов, а плоскость данных проста и не включает в себя никаких промежуточных устройств обработки (middlebox). Как

оказалось, физические топологии слишком сложны для непосредственного управления, и приложения должны взаимодействовать только с виртуальными представлениями, коммутаторы в ядре сети и на ее периметре принципиально отличаются друг от друга, число виртуальных программных коммутаторов превышает число физических, а промежуточные устройства обработки применяются массово. С этими выводами трудно не согласиться, однако нельзя не отметить, что компания Nicira, основателем которой являлся Шенкер, начала свою деятельность несколько лет назад с создания виртуального коммутатора OpenvSwitch, а ее основным продуктом является платформа виртуализации сетей NVP (NetworkVirtualizationPlatform).

В 2012 году Nicira была приобретена VMware. Однако интеграция технических решений потребовала значительного времени. Только в марте 2013 года была анонсирована система VMwareNSX, которая должна объединить vCloudNetworkandSecurity (vCNS) и NVP [2]. Платформа NSX предназначена для создания уровня виртуализации над существующим оборудованием и предоставляет интерфейс к упрощенным логическим устройствам: портам, коммутаторам, маршрутизаторам, распределенным межсетевым экранам. В ее функции входят также мониторинг, обеспечение безопасности и качества обслуживания. Логические сетевые устройства при помощи программного интерфейса (API) собираются в топологию, создавая виртуальную сеть с изоляцией трафика и полной поддержкой большого количества пользователей (multi-tenant). Основными компонентами NSX являются: кластер контроллеров, виртуальные коммутаторы под управлением гипервизора, шлюзы, компоненты сторонних производителей и управляющая программа NSXManager. Кластер контроллеров отвечает за развертывание всей виртуальной архитектуры, принимая запросы от платформ управления (vCloud, OpenStack), рассчитывая виртуальную сетевую топологию и программируя виртуальные коммутаторы и шлюзы в упреждающем режиме (proactively). Для решения этих задач ему доступна информация обо всех виртуальных машинах и сервисах NSX. Каждый элемент кластера является равноправным и взаимозаменяемым. Для масштабирования системы достаточно просто добавить дополнительные узлы. Виртуальный коммутатор в гипервизоре состоит из высокопроизводительного модуля ядра, базы данных конфигурации и программируемой плоскости данных с уровнями L2-L4. Кроме виртуализации топологии, такая архитектура создает новые возможности и для виртуализации сетевой безопасности, позволяя отвязать политики безопасности от IP-адресов и реализовать межсетевые экраны в модулях гипервизора, непосредственно взаимодействующих с виртуальными машинами. При этом политики безопасности смогут оперировать объектами более высокого уровня, нежели заголовок TCP, и без отрыва от контекста. Для взаимодействия нескольких гипервизоров в рамках одной виртуальной сети

контроллер динамически организует между ними туннели с применением инкапсулирующих протоколов STT и VXLAN, осуществляя развязку адресных пространств виртуальной и физической сети. Для соединения NSX с физическими серверами, удаленными офисами и внешними сетями предназначены сервисы шлюзов. Они реализованы на базе таких же виртуальных коммутаторов и также находятся под управлением кластера контроллеров. Платформа является расширяемой и позволяет регистрировать сервисы сторонних производителей в контроллере NSX, в том числе для интеграции в виртуальные сети устройств обработки трафика (L4-L7 serviceappliance). Подсистема NSXManager предоставляет графическую панель управления с веб-интерфейсом для взаимодействия администратора с кластером контроллеров, настройки системы и решения проблемных ситуаций. По каждому элементу виртуальной сети доступен журнал регистрации событий и текущее состояние. Предусмотрены механизмы для облегчения установления соответствия между виртуальными топологиями и соответствующими им физическими элементами сети. NSXManager способен создавать мгновенные снимки состояния всей сети, архивировать и анализировать их, возвращаться к предыдущим состояниям. Запуск всей платформы планируется во второй половине 2013 года.

Для решения аналогичных задач компания Microsoft использует Hyper-VNetworkVirtualization на базе NVGRE [6]. Функциональная схема этого решения во многом аналогична. Виртуальный коммутатор (vmSwitch) выполняет коммутацию на уровне L2 под управлением гипервизора Hyper-V. Ядром системы служит модуль виртуализации WindowsNetworkVirtualization (WNV), который размещен между физическим сетевым адаптером (NIC) и внутренней виртуальной подсетью (virtualsubnet, VSID). Разница состоит в логике работы: WNV является полнофункциональным коммутатором уровня L3. WNV не использует мосты, все пакеты продвигаются на основании IP-адресов, даже внутри единой виртуальной подсети (VSID). Кроме того, не используется широковещательная передача для динамического установления соответствия MAC-адресов виртуальных машин и точек входа в туннели (VTEP). Вся информации загружается предварительно. Для установления соответствия между MAC-адресом виртуальной машины и IP-адресом гипервизора, а также между MAC- и IP-адресами виртуальной машины используется центральный контроллер. Модули WNV самостоятельно обслуживают ARP-запросы и детерминируют обработку каждого пакета. Таким образом, не требуется поддержка множественной адресации (multicast) в транспортной сети и улучшаются возможности масштабирования. К недостаткам решения можно отнести то, что, поскольку модуль WNV использует продвижение на уровне L3, виртуальная сеть не может использовать протоколы, отличные от IP. Впрочем, благодаря полной поддержке WNV не только IPv4, но и IPv6 как в

виртуальной, так и в транспортной сети в большинстве случаев это не принципиально. Также не будут работать нестандартные решения, например манипуляции с ARP для организации кластера с общим IP-адресом.

Интересно, что в родственных областях коммуникационных приложений также приходят к инициативам вида NFV [3], т.е. виртуализации.

Анализ текущего состояния технологий ПКС приводит к следующему выводу: несмотря на то, что OpenFlow является де-факто единственным протоколом низкого уровня для управления продвижением пакетов в коммутаторах и его в той или иной степени поддерживают все ведущие производители сетевого оборудования, роль OpenFlow в существующих массовых решениях весьма незначительна. Часто практическое применение находят системы виртуализации сетей, не зависящие от используемого сетевого оборудования, совместимые с различными гипервизорами и платформами управления облачными сервисами. Помимо автоматизации операций, они предоставляют пользователям возможность управлять упрощенными виртуальными топологиями, могут быть внедрены в существующие ЦОД без перерыва в обслуживании и способны взаимодействовать с внешними сетями по стандартным протоколам, таким как MPLS и BGP.

Исследования проводились при финансовой поддержке Министерства образования и науки Российской Федерации в рамках государственного контракта № 14.514.11.4045 от 01 марта 2013 г.

Список литературы

1. Juniper Networks Proactive Overlay Versus Reactive End-To-End [Электронный ресурс]. – Режим доступа: http://www.juniper.net/us/en/forms/overlay-sdn-wp/ (дата обращения: 20.07.2013).

2. Naguib H. VMware NSX Network Virtualization [Электронный ресурс]. – Режим доступа: http://blogs.vmware.com/vmware/2013/03/vmware-nsx-network-virtualization.html (дата обращения: 20.07.2013).

3. Network Functions Virtualisation – Introductory White Paper [Электронный ресурс]. – Режим доступа: http://portal.etsi.org/NFV/NFV_White_Paper.pdf (дата обращения: 20.07.2013).

4. Nuage Networks Unconstrained Datacenter Networks for the Cloud Era [Электронный ресурс]. – Режим доступа: http://www.nuagenetworks.net/products/ (дата обращения: 20.07.2013).

5. Lewin-Eytan L. Designing Modular Overlay Solutions for Network Virtualization IBM Research Report / L. Lewin-Eytan, K. Barabash, R. Cohen, V. Jain, A. Levin // Computer Science, August 28, 2011.

6. Pepelnjak I. Hyper-V Network Virtualization (WNM/NVGRE): Simply Amazing [Электронный ресурс]. – Режим доступа: http://blog.ioshints.info/2012/12/hyper-v-network-virtualization-wnvnvgre.html (дата обращения: 20.07.2013).
7. Shenker S. Software-Defined Networking at the Crossroads [Электронный ресурс]. – Режим доступа: http://ee380.stanford.edu/cgi-bin/videologger.php?target=130515-ee380-300.asx (дата обращения: 10.06.2013).

Рецензенты:

Парфенов В.Г., д.т.н., профессор, декан факультета информационных технологий и программирования Санкт-Петербургского национального исследовательского университета информационных технологий, механики и оптики (НИУ ИТМО), г. Санкт-Петербург.

Горелик С.Л., д.т.н., профессор Санкт-Петербургского национального исследовательского университета информационных технологий, механики и оптики (НИУ ИТМО), г. Санкт-Петербург.