

УДК 004.5, 004.6, 004.7

ИНТЕГРАЦИЯ ПРОГРАММНО-КОНФИГУРИРУЕМОЙ СЕТИ В ОБЛАЧНОЕ РЕШЕНИЕ

Тейхриб А. П.

Общество с ограниченной ответственностью «Наумен консалтинг», Москва, Россия (115230, Москва, Варшавское шоссе, 47, корп.4), e-mail: ateyhrib@naumen.ru.

Статья посвящена вопросам интеграции программно-конфигурируемых сетей в облачное решение. Обозначены основные преимущества использования программно-конфигурируемых сетей. Рассмотрены основные варианты реализации программно-конфигурируемой сети: на основе оборудования, совместимого с открытым протоколом OpenFlow, либо с использованием проприетарного оборудования с закрытым протоколом для выполнения конфигурации, а также относительно новый подход решения рассматриваемой проблемы, называемый Network Functions Virtualization. Сделан вывод о целесообразности применения протокола OpenFlow как основы для управления программно-конфигурируемой сетью. Далее рассмотрены возможности протокола OpenFlow, а также структура связей между компонентами, взаимодействующими в рамках программно-конфигурируемой сети. Рассмотрены особенности применения программно-конфигурируемых сетей в облачной инфраструктуре. Определен набор операций, которые должны быть реализованы при использовании OpenFlow в качестве протокола для реализации программно-конфигурируемой сети в рамках управления вычислительными сетями в облачной инфраструктуре.

Ключевые слова: программно-конфигурируемая сеть (ПКС), интеграция, облачные вычисления.

INTEGRATION OF THE SOFTWARE-DEFINED NETWORKING IN THE CLOUD SOLUTION

Teyhrib A. P.

«Naumen consulting» Limited Liability Company, Moscow, Russian (115230, Moscow, Warsaw highway, 47-4), e-mail: ateyhrib@naumen.ru.

This paper presents problems with the integration of software-defined networkings in the cloud solution. The main benefits of using software-defined networkings are identified. The main variants of the software-defined networking: based on equipment that is compatible with an open protocol OpenFlow, or using proprietary hardware with proprietary protocol for configuration, also a relatively new approach to solution of the problem, called the Network Functions Virtualization. The conclusion about the feasibility of the protocol OpenFlow, as a basis for managing software-defined networking, is made. Also the possibilities of protocol OpenFlow and the structure of connections between components that interact within the software-defined networking are described. The features of the application software and configurable networks in the cloud infrastructure are presented. A set of operations that must be implemented using OpenFlow as the protocol for the implementation of software-defined networking within the control area networks in the cloud infrastructure is defined.

Keywords: Software Defined Networking (SDN), integration, cloud computing.

Введение

Программно-конфигурируемые сети (ПКС или Software Defined Networkings или SDN) [2] – развивающаяся архитектура сети, где функция управления сетью разделена с функцией передачи данных и полностью программируема. Основная идея ПКС состоит в том, чтобы, не изменяя существующего сетевого оборудования, отделить (перехватить) управление этим оборудованием (маршрутизаторами и коммутаторами) за счет создания специального

программного обеспечения, которое может работать на обычном отдельном компьютере и которое находится под контролем администратора сети.

Учитывая постоянно увеличивающееся в последние годы распространение облачных технологий, одним из перспективных направлений исследований и разработки представляется интеграция программно-конфигурируемых сетей в облачное решение. С точки зрения облачных технологий, управление различными частями инфраструктуры (в нашем случае сетями) осуществляется посредством сервисов, либо удаленно. Для формирования сервисов необходимо создание программных интерфейсов, в то время как для использования в веб-среде необходима панель управления.

В рамках обозначенного направления существует ряд тем, требующих рассмотрения:

- 1) Целесообразность применения протокола OpenFlow как основы для управления ПКС.
- 2) Разработка программного интерфейса на базе протокола OpenFlow:
 - а) исключить некоторую функциональность OpenFlow;
 - б) объединить отдельные функции, получив более высокоуровневую абстракцию.

Целесообразность применения протокола OpenFlow

Использование программно-конфигурируемых сетей несет в себе ряд преимуществ [6]:

- 1) В случае использования операторами центров обработки данных применение ПКС позволяет выполнить виртуализацию сетевой инфраструктуры для поддержки универсального способа мультиарендности в вычислениях, хранении и сети.
- 2) В случае использования поставщиками сервисов ПКС позволяет получить эффективное с точки зрения цены управление сетями как внутри датацентров, так и между ними.
- 3) Замену оборудования, такого как межсетевые экраны, балансировщики нагрузки, системы обнаружения атак программными системами.
- 4) Возможность полной персонализации используемых устройств, которая позволяет на некотором оборудовании получить уникальный набор функциональности.
- 5) Работа со сложностями виртуальных сетей на основе более обобщенного подхода.

Таким образом, само по себе использование программно-конфигурируемых сетей является полезным для управления вычислительными сетями, однако возникает вопрос о применении протокола OpenFlow, а не другого варианта.

Рассмотрим возможные варианты реализации программно-конфигурируемых сетей:

- 1) На основе оборудования, совместимого с открытым протоколом OpenFlow.
- 2) Использование проприетарного оборудования с закрытым протоколом для выполнения конфигурации.

Функционально оба варианта сравнимы, однако второй вариант проигрывает первому в части открытости и стандартизованности, а также взаимозаменяемости аппаратных средств, т.к. выбор ограничивается конкретным производителем.

Альтернативой ПКС и OpenFlow, предлагающей схожие преимущества, является более молодой подход, называемый Network Functions Virtualization [4] (NFV) – виртуализация сетевой функциональности, который был предложен в 2012 году рядом крупнейших телекоммуникационных компаний (такими как AT&T, China Mobile, Deutsche Telekom, Verizon и др.) на конференции, посвященной ПКС. Работы по данному подходу проходят в рамках Европейского института по стандартизации в области телекоммуникаций (ETSI, <http://www.etsi.org/>).

Данный подход может быть использован в дополнении к подходу ПКС. Рассмотрим взаимное отношение данных подходов (рисунок 1).



Рисунок 1. Взаимное отношение NFV и ПКС

NFV пересекается с ПКС в части использования унифицированного оборудования, реализующего различную функциональность. Отличие заключается в том, что NFV подразумевает полную замену всего сетевого оборудования на унифицированные сервера, управляемые некоторым программным обеспечением, а ПКС подразумевает управление некоторым централизованным программным обеспечением, сетевым оборудованием. Другое отличие заключается в зрелости технологий – если OpenFlow и ПКС уже имеют промышленное применение, то данный подход в текущем моменте является в большей степени направлением, авторы которого предлагают подключиться сообществу для его дальнейшего развития. С учетом этого можно говорить о том, что NFV является развитием ПКС. Таким образом, с точки зрения

достижения целей исследования, OpenFlow является максимально применимым на данный момент.

Разработка программного интерфейса на базе протокола OpenFlow

Как было отмечено выше, в рамках данной темы необходимо провести ревизию протокола OpenFlow применительно к управлению сетевой инфраструктурой в облаке. Рассмотрим структуру и возможности OpenFlow.

Структура OpenFlow

В соответствии со спецификацией протокола управления и конфигурации OpenFlow версии 1.1.1 структура связей между компонентами, взаимодействующими в рамках программно-конфигурируемой сети, выглядит следующим образом (рисунок 2) [5]:

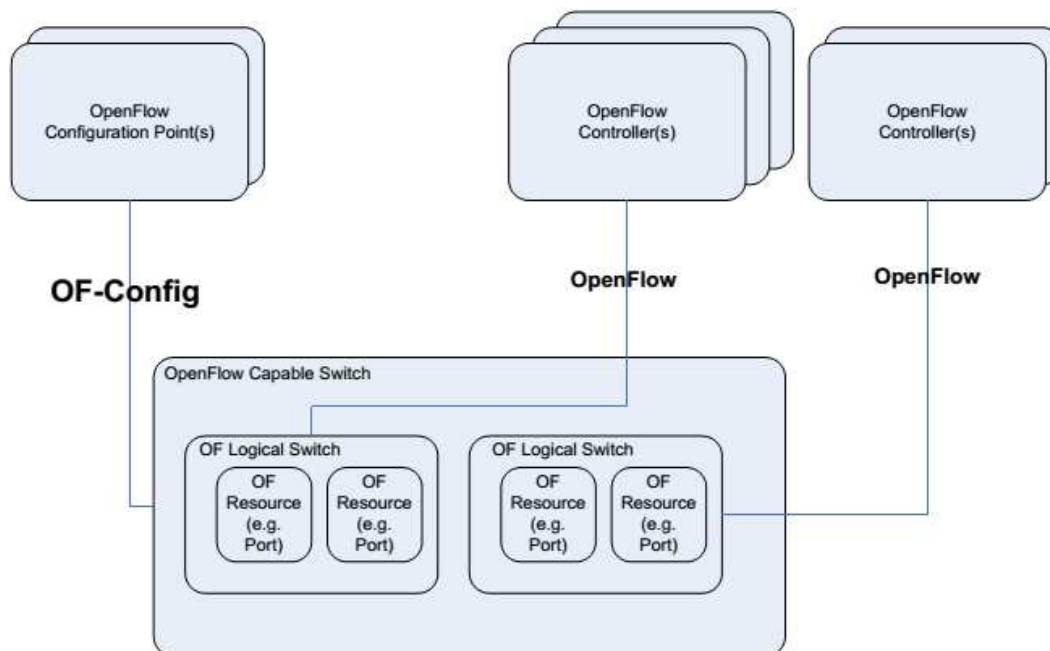


Рисунок 2. Связи между компонентами, взаимодействующими по протоколу OpenFlow

Рассмотрим компоненты, представленные на рисунке 2:

- 1) OpenFlow Capable Switch – коммутатор, который соответствует данной спецификации и может управляться по протоколу OpenFlow Config, в дальнейшем будем его называть OpenFlow коммутатор. Это физическая единица оборудования.
- 2) OpenFlow Logical Switch – логический коммутатор OpenFlow, который работает в рамках OpenFlow коммутатора, представляет собой внутреннюю программу, при этом в рамках одного OpenFlow коммутатора может одновременно выполняться несколько таких программ. Данный

компонент предназначен для обработки пакетов, поступающих на него и их дальнейшего перенаправления.

3) OpenFlow Controller – сервис для управления логическим коммутатором OpenFlow, а также отвечает на запросы логического коммутатора относительно необходимых операций, которые должны быть выполнены над пакетом, полученным логическим коммутатором OpenFlow, для которого нет правил.

4) OpenFlow Configuration Point – точка конфигурации, данный компонент предназначен для управления OpenFlow Capable Switch, и его реализация никак не фиксируется в спецификации и поэтому данный компонент может являться как самостоятельным сервисом, так и совмещенным с OpenFlow Controller.

5) OpenFlow Resource – физический ресурс коммутатора OpenFlow, который может также быть привязан к логическому коммутатору OpenFlow, ресурсы подразделяются на два типа:

а) port – порт коммутатора, на который могут поступать пакеты и через который они могут перенаправляться дальше;

б) queue – очередь поступающих пакетов, в нее помещаются поступающие пакеты, до момента обработки.

Помимо компонентов на рисунке 2 видно наличие двух протоколов:

1) OpenFlow – протокол, предназначенный для взаимодействия логического коммутатора OpenFlow и контроллера OpenFlow, направленное на решение вопросов об обработке пакетов.

2) OpenFlow Config – протокол, предназначенный для управления коммутатором OpenFlow, является дополнительным к протоколу OpenFlow и отличается тем, что в нем реализуется управление физическим устройством.

С точки зрения применения подхода ПКС в области управления вычислительными сетями в облачной инфраструктуре интерес представляет взаимодействие логического коммутатора OpenFlow и OpenFlow контроллера, а также управление этим контроллером. Такое решение связано с тем, что панель управления, а также программный интерфейс, необходимые для управления сетями в облачной инфраструктуре, реализуются посредством конфигурирования контроллера с последующей реконфигурацией.

Особенность применения для облачной инфраструктуры

Рассмотрим вычислительные сети в облачной инфраструктуре, которые управляются на основе подхода ПКС. «Облачность» в данном контексте накладывает ряд ограничений и условий на выполнимые операции:

- 1) Операции управления должны выполняться удаленно через некоторый программный интерфейс, а также панель управления.
- 2) Операции управления должны быть изолированы в рамках инфраструктуры определенных пользователей.

Рассмотрим возможные способы реализации интерфейса для удаленного управления. Существует несколько вариантов, которые могут использоваться:

- 1) SOAP – это облегченный протокол, предназначенный для обмена структурированной информацией в децентрализованной, распределенной среде. Для определения наращиваемой оболочки обмена сообщениями, обеспечивающей структуру сообщения, которая может быть использована при обмене различными базовыми протоколами, SOAP использует XML технологии. Эта оболочка разработана независимой от любой конкретной модели программирования и других особых семантик реализации [3].
- 2) REST – представляет собой архитектурный стиль, который можно использовать для создания программных средств, в которых клиенты (агенты пользователей) могут отправлять запросы службам (конечным точкам). REST является одним из способов реализации архитектурного стиля «клиент-сервер» – по сути, REST явно опирается на архитектурный стиль «клиент-сервер» [1].

По функциональным возможностям оба подхода предоставляют схожие возможности, поэтому основным критерием становится простота использования достижения целей исследования. С этой точки зрения, а также на основе имеющегося опыта предпочтение отдается подходу, основанному на REST. При этом вопрос изоляции операций решается аутентификацией и авторизацией пользователя, выполняющего запросы в рамках REST.

Необходимый интерфейс

Задача управления сетями в облачной инфраструктуре подразумевает управление взаимодействием следующих сущностей:

- а) вычислительными узлами в рамках сети;
- б) вычислительными сетями;
- в) данными (пакетами данных), передаваемых по сети.

Управление этими сущностями в рамках управления сетями в облачной инфраструктуре требует следующих типов операций, которые должны быть реализованы в системе:

- а) добавление/удаление/перенос вычислительной машины в сети;
- б) создание/удаление/изменение сетей;
- в) создание/удаление/изменение правил маршрутизации пакетов в сети;

- г) получение информации о состоянии сети;
- д) получение информации о правилах маршрутизации;
- е) получение информации о состоянии выполнения операций.

Эти операции являются базовыми, которые необходимы пользователю при работе с системой, поэтому программный интерфейс управления сетями в облачной инфраструктуре должен позволять их реализовать. При этом сам интерфейс может иметь дополнительные методы, которые необходимы для выполнения данных операций.

Выводы

Таким образом, можно сделать вывод о целесообразности применения протокола OpenFlow в качестве протокола для реализации ПКС в рамках управления вычислительными сетями в облачной инфраструктуре. Также определен набор операций, которые должны быть реализованы при использовании OpenFlow в качестве протокола для реализации программно-конфигурируемой сети в рамках управления вычислительными сетями в облачной инфраструктуре.

Работа выполнена при финансовой поддержке Минобрнауки в рамках государственного контракта №14.514.11.4085.

Список литературы

1. Введение в службы RESTful с использованием WCF [Электронный ресурс] //Microsoft: [web-сайт] . – USA, 2009. – Режим доступа: <http://msdn.microsoft.com/ru-ru/magazine/dd315413.aspx> (дата обращения: 11.07.2013).
2. Ермолаева А. Б., Ромашова Н. В. Программно-конфигурируемые сети и их развитие [Электронный ресурс] //ИТМиВТ: [web-сайт]. – М., 2012. – Режим доступа: <http://www.ipmce.ru/about/press/popular/pks/> (дата обращения: 08.07.2013).
3. Понимание SOAP [Электронный ресурс] // Thalion Group: [web-сайт]. – Киев, 2013. – Режим доступа: <http://www.thalion.kiev.ua/idx.php/108/260/article/> (дата обращения: 08.07.2013).
4. Network Functions Virtualisation [Электронный ресурс] //ETSI: [web-сайт]. – FRANCE, 2012. – Mode Access: http://portal.etsi.org/NFV/NFV_White_Paper.pdf (дата обращения: 11.07.2013).
5. OpenFlow Management and Configuration Protocol [Электронный ресурс]: //ONF [web-сайт] . – USA, 2013. – Mode Access: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow-config/of-config-1-1-1.pdf> (дата обращения: 11.07.2013).

6. SDN: Transforming networking to accelerate business agility [Электронный ресурс] //Open Networking Summit 2013: [web-сайт]. – USA, 2013. – Mode Access: <http://www.opennetsummit.org/why-sdn.html> (дата обращения: 10.07.2013).

Рецензенты:

Коновалов А.В., д.т.н., профессор, заведующий лабораторией механики деформаций института Машиноведения Уральского отделения Российской академии наук, г.Екатеринбург.

Суханов В.И., д.т.н., доцент, заведующий кафедрой программных средств и систем Уральского федерального университета им. Б. Н. Ельцина, г.Екатеринбург.