Научная статья
Original article
УДК 004.72

ПРОГРАММНО-КОНФИГУРИРУЕМЫЕ СЕТИ. ПРОБЛЕМЫ ПРИ ПЕРЕХОДЕ К СЕТЯМ ПКС

SOFTWARE-DEFINED NETWORKS. PROBLEMS IN THE TRANSITION TO SOFTWARE-DEFINED NETWORKS

Фёдоров Никита Константинович, магистрант, Национальный исследовательский университет ИТМО, г. Санкт-Петербург

Fedorov Nikita Konstantinovich, Master Student, National Research University ITMO, St. Petersburg

Аннотация

В сложных распределенных масштабируемых сетях происходят быстрые изменения. Из этого следует проблема в управлении такими сетями. Средства управления такими сетями должны отвечать следующим критериям: безопасность, масштабирование, управляемость, гибкость.

Отличным решением являются программно-конфигурируемые сети (Software Defined Networking (SDN)). Ключевые принципы программно-определяемых сетей – разделение процессов передачи и управления данными, централизация управления сетью при помощи унифицированных программных средств, виртуализация физических сетевых ресурсов.

В данной работе рассматривается понятие программно-конфигурируемых сетей, их архитектура, а также определены проблемы, с которыми специалисты могут столкнуться при переходе к программно-конфигурируемым сетям и

описаны возможные варианты их решения. Список проблем при переходе к SDN сетям включает: размещение контроллера, масштабируемость, производительность, безопасность, совместимость, надежность.

Annotation

Complex, distributed, scalable networks are undergoing rapid change. This implies a problem in the management of such networks. Management tools for such networks must meet the following criteria: security, scalability, manageability, flexibility.

An excellent solution is software-defined networks (Software Defined Networking (SDN)). The key principles of software-defined networks are the separation of data transmission and management processes, the centralization of network management using unified software tools, and the virtualization of physical network resources.

This paper discusses the concept of software-defined networks, their architecture, and identifies the problems that specialists may encounter when moving to software-defined networks and describes possible solutions. The list of challenges in moving to SDN networks includes: controller placement, scalability, performance, security, compatibility, and reliability.

Ключевые слова: программно-конфигурируемые сети, ПКС, OpenFlow, контроллер, безопасность

Keywords: software-defined networks, SDN, OpenFlow, controller, security

ВВЕДЕНИЕ

Использование пакетов данных стало одним из требований телекоммуникационной отрасли к устройству, поддерживающему распределенную сеть передачи данных. Традиционная сетевая инфраструктура использует низкоуровневую конфигурацию, ориентированную на поставщика. Аппаратная топология требует настройки на каждом устройстве. В долгосрочной перспективе сложность будет замедлять или даже сдерживать инновации. Программно-определяемые сети (SDN), использующие устройства Openflow, предлагают управление всеми сетевыми устройствами, включая управление

трафиком только через одну среду и один протокол. SDN отделяет плоскость управления от устройства и сосредоточит ее на одном устройстве, которое действует как контроллер.

Благодаря централизации контроллера сетевое программирование может быть автоматизировано и может динамически изменяться по мере необходимости. Технология SDN также обеспечивает гибкость сети, устраняет зависимость от определенных поставщиков и позволяет организации разрабатывать новые инновационные сетевые приложения. Для наиболее быстрого и эффективного развертывания SDN необходимо прежде всего определить проблемы, с которыми специалисты могут столкнуться при внедрении новых технологий.

1. ПОНЯТИЕ ПРОГРАММНО-КОНФИГУРИРУЕМЫХ СЕТЕЙ

Программно-определяемая сеть — это концепция сети, которая отделяет плоскость управления от устройства. Централизованный контроллер предназначен для повышения гибкости и способности сети расти. Контроллер SDN может выполнять функции для следующих коммутаторов, такие как конфигурация, упорядоченная доставка пакетов, получение статистических данных коммутатора и другие функции, связанные с управлением сетью [1].

В архитектуре SDN уровень данных и уровень управления разделены, а логика управления (мозг) реализована в логически централизованном контроллере, поскольку трафик фактически контролируется интеллектуальным контроллером. На рисунке 1 показана архитектура SDN.

Уровни программно-конфигурируемых сетей (SDN) [2,3]**:**

1. Уровень управления в SDN – это контроллер, который отвечает за определение оптимального пути для потоков данных. Контроллер – это специализированное программное обеспечение, установленное на персональный компьютер с предустановленной сетевой операционной системой или сервер. Контроллер

предоставляет приложениям сетевые сервисы и программный интерфейс для управления сетевыми элементами и сетью.

2. Уровень данных в SDN представляет из себя набор сетевых элементов (например, коммутаторы и маршрутизаторы) и отвечает за перенос пользовательских данных по сети в соответствии с путем, определенным уровнем управления.

3. Уровень сетевых приложений – набор SDN-приложений, взаимодействующих с SDN-контроллером через программный протокол (API) для сбора, анализа, развёртывания и управления сетевой инфраструктурой на уровне приложений.

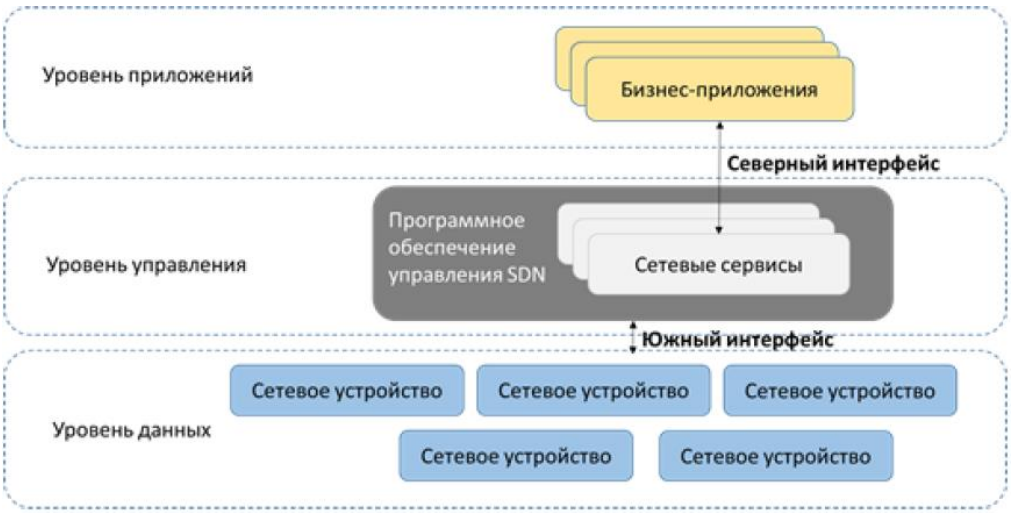


Рисунок 1 – Архитектура SDN

Взаимодействие между уровнями обеспечивается северным и южным интерфейсами [4]:

- Северный интерфейс (Northbound interface).

Северный интерфейс – это API, который используется для обмена информацией между контроллером SDN и приложениями, совместимыми с SDN, работающими в сети. API-интерфейсы Northbound, по существу, берут сетевые требования из приложений SDN и согласовывают потребности с сетевым контроллером, который

отвечает за предоставление приложениям оптимальных сетевых ресурсов и путей.

- Южный интерфейс (Southbound interface).

Южный интерфейс – это связывающее соединение между средствами управления и сетевыми устройствами. Как только северный интерфейс сообщает контроллеру, какие возможности необходимы приложению SDN, контроллер должен сообщить аппаратному обеспечению коммутатора SDN, как обрабатывать поток данных. OpenFlow используется в качестве южного интерфейса для передачи этой информации между контроллером и аппаратным обеспечением коммутатора, совместимого с SDN.

За исключением логической архитектуры, которая показывает, что все коммутаторы, маршрутизатор и другие устройства управляются контроллером, требуется стандартный и безопасный канал между контроллером SDN и сетевыми устройствами. OpenFlow – это наиболее широко распространенный открытый стандарт южного API для SDN.

2. ПРОБЛЕМЫ ПРИ ПЕРЕХОДЕ К SDN СЕТЯМ

Хотя SDN является подходящим решением для ИТ и облачных провайдеров и предприятий, SDN сталкивается с некоторыми проблемами, которые препятствуют его производительности и внедрению. Список проблем SDN включает [5,6]:

- Размещение контроллера.
- Масштабируемость.
- Производительность.
- Безопасность.
- Совместимость.
- Надежность.

Контроллеры SDN должны быть правильно настроены, а топология сети SDN должна быть аутентифицирована, чтобы предотвратить ручные ошибки и повысить доступность сети. В традиционной сети, когда одна сеть или несколько сетевых устройств выходят из строя, сетевой поток данных направляется через другой или близлежащие узлы или устройства для обеспечения непрерывности потока данных.

Однако в архитектуре централизованного контроллера SDN за все сети отвечает один контроллер, и в случае отказа центрального контроллера вся сеть разрушается, поскольку нет альтернативного контроллера. Чтобы решить эту проблему, облачной организации необходимо сосредоточиться на том, как эффективно использовать основные функции контроллера, которые могут повысить надежность сети. Контроллер SDN должен иметь возможность поддерживать решения с несколькими путями или быструю перенаправление трафика на активные пути в случае сбоя пути/канала. Если основной контроллер выходит из строя, новые архитектуры поддерживают альтернативный контроллер, который может обрабатывать поток трафика. Контроллеры также поддерживают такие технологии, как протокол резервирования виртуального маршрутизатора (VRRP) и группы агрегации каналов с несколькими шасси, чтобы повысить доступность сети.

Вторая проблема SDN — это **масштабируемость**, потому что в этом подходе плоскости данных и управления разделены, но они могут развиваться независимо, пока API соединяет их. Поскольку архитектура SDN включает централизованные или частично распределенные контроллеры, взаимодействующие с плоскостями данных на нескольких устройствах, существует вероятность того, что контроллеры станут узким местом в сети. В частности, большие сети с объемами сетевых запросов могут перегружать контроллеры. По мере роста сетей узкие места сужаются, и производительность сети снижается.

Масштабируемость можно улучшить с помощью децентрализованной архитектуры управления или аналогичного решения, такого как разделенные или

полностью распределенные плоскости управления. Но такие решения могут создавать новые препятствия, такие как конвергенция и бесчисленные экземпляры управления, которые нужно настраивать и управлять.

Производительность сети – самая большая проблема для всех сетей. Независимо от того, насколько надежна, безопасна, масштабируема или совместима сеть, ее нельзя использовать, если ей не хватает производительности. SDN — это метод, основанный на потоках, поэтому производительность измеряется по двум показателям: время настройки потока и количество потоков в секунду, которые может переключить контроллер. Flow-setup работает в двух режимах: проактивном и реактивном. Эти два режима имеют свои соответствующие накладные расходы на инициирование и ограничение потока. Чтобы преодолеть ограничение производительности, необходимо сосредоточить внимание на факторах, влияющих на время настройки потока и производительность ввода-вывода контроллера.

Существуют средства и способы повышения производительности за счет рассмотрения хорошо известных методов оптимизации, таких как пакетирование ввода / вывода и использование подхода Maestro, который использует такие методы, как порог пакетной обработки ввода (IBT) и порог ожидающих необработанных пакетов (PRT).

Безопасность. Требуется больше внимания уделять безопасности, если SDN будет приемлемой в более широком развертывании.

Поскольку плоскость управления играет такую центральную функцию в архитектуре SDN, стратегии безопасности должны быть сосредоточены на защите контроллера и аутентификации доступа приложения к плоскости управления. Одно из возможных решений - авторизация на основе ролей. Однако авторизация на основе ролей сама по себе не является решением сложной проблемы SDN, требующей изоляции приложений или ресурсов.

Контроллеры являются особенно привлекательной целью для атак в архитектуре SDN, открытой для несанкционированного доступа и использования. Более того, при отсутствии надежной и безопасной платформы контроллера злоумышленник может маскироваться под контроллером и выполнять злонамеренные действия.

Технология безопасности, такая как Transport Layer Security (TLS) с взаимной аутентификацией между контроллерами и их коммутаторами, может уменьшить эти угрозы. Текущие спецификации OpenFlow описывают использование TLS. Однако функция безопасности не является обязательной, и стандарт TLS не указан.

С одним контроллером, управляющим набором сетевых узлов, реализация аутентификации с помощью TLS может обеспечить необходимую безопасность. Однако, когда несколько контроллеров взаимодействуют с одним узлом, или несколько процессов управления взаимодействуют с одним централизованным контроллером, авторизация и управление доступом становятся более сложными. Вероятность несанкционированного доступа возрастает и может привести к манипулированию конфигурацией узла и / или трафиком через узел со злым умыслом.

Одна из возможных вредоносных атак — это атака отказа в обслуживании (DoS). Есть два варианта обработки нового потока, когда в таблице потоков нет совпадения. Либо полный пакет, либо часть заголовка пакета передается контроллеру для разрешения запроса. При большом объеме сетевого трафика отправка полного пакета контроллеру потребует высокой пропускной способности.

Однако, если контроллеру передается только информация заголовка, сам пакет должен храниться в памяти узла до тех пор, пока не будет возвращена запись таблицы потоков. В этом случае злоумышленнику будет легко выполнить DoS-атаку на узел, настроив ряд новых и неизвестных потоков. Поскольку элемент

памяти узла может быть узким местом из-за высокой стоимости, злоумышленник потенциально может перегрузить память коммутатора.

С другой стороны, архитектура SDN поддерживает высокоактивную систему мониторинга, анализа и реагирования на безопасность. С точки зрения безопасности SDN может:

- Быстро идентифицировать угрозы и управлять ими посредством цикла сбора информации из сети, ее анализа, обновления политики и последующего перепрограммирования для оптимизации работы в сети.
- Изменять политики безопасности: позволяет определить политику безопасности и применить ее ко всем элементам инфраструктуры, уменьшая частоту неправильной конфигурации и конфликтующих политик в инфраструктуре.
- Внедрение служб безопасности: упрощение внедрения служб безопасности, когда приложения, такие как межсетевые экраны и системы обнаружения вторжений (IDS), могут применяться к указанному трафику в соответствии с политиками организации.

Совместимость – было бы просто развернуть совершенно новую инфраструктуру на основе технологии SDN. Для этого все элементы и устройства в сети будут поддерживать SDN. Однако сегодня существует обширная установленная база сетей, поддерживающих жизненно важные системы и предприятия. Простая «замена» этих сетей на новую инфраструктуру невозможна и хорошо подходит только для закрытых сред, таких как центры обработки данных и сети университетского городка.

Поэтому переход к SDN требует одновременной поддержки SDN и устаревшего оборудования. Элемент вычисления пути IETF (PCE) может помочь в постепенном или частичном переходе на SDN. С помощью PCE компонент вычисления пути в сети перемещается из сетевого узла в централизованную роль,

в то время как традиционные сетевые узлы, не использующие PCE, продолжают использовать свою существующую функцию вычисления пути. Специальный протокол (PCEP) обеспечивает связь между элементами сети. Однако PCE не предоставляет полную SDN. Централизованный контроллер SDN поддерживает полное вычисление пути для потока через несколько сетевых узлов.

Дальнейшее развитие требуется для создания гибридной инфраструктуры SDN, в которой традиционные узлы с поддержкой SDN и гибридные сетевые узлы могут работать в гармонии. Такая совместимость требует поддержки соответствующего протокола, который вводит требования к интерфейсам связи SDN и обеспечивает обратную совместимость с существующими технологиями IP-маршрутизации и уровня управления MPLS. Такое решение снизило бы стоимость, риски и время простоя для корпоративных сетей и сетей операторов, переходящих на SDN.

ЗАКЛЮЧЕНИЕ

В сложных распределенных масштабируемых сетях происходят быстрые изменения. Из этого следует проблема в управлении такими сетями. Средства управления такими сетями должны отвечать следующим критериям: безопасность, масштабирование, управляемость, гибкость.

Отличным решением являются программно-конфигурируемые сети (Software Defined Networking (SDN)). Ключевые принципы программно-определяемых сетей – разделение процессов передачи и управления данными, централизация управления сетью при помощи унифицированных программных средств, виртуализация физических сетевых ресурсов.

Были выявлены проблемы при переходе к SDN сетям и описаны возможные варианты их решения. Список проблем при переходе к SDN сетям включает: размещение контроллера, масштабируемость, производительность, безопасность, совместимость, надежность.

Литература

1. Hend Abdelagder Eissa, Kenz A. Bozed, Hadil younix. Software Defined Networking // International conference on Sciences and Techniques of Automatic control & computer engineering (STA). – October 2019.
2. Saleh Asadollahi, Dr. Bhargavi Goswami, Dr. Atul M Gonsai. Software Defined Network, Controller Comparison // International Journal of Innovative Research in Computer and Communication Engineering. – Vol.5. – April 2017.
3. Клепче В. Программно-определяемые сети сегодня // Connect WIT. – №1-2. – 2019.
4. Панеш А. Х. Содержание и перспективы технологий программно-конфигурируемых сетей и виртуализации сетевых функций // Вестник Адыгейского Государственного университета. – 2014. – 2. – 120-127.
5. Veeramani Shamugam, I Murray, J A Leong, Amandeep S Sidhu. Software Defined Networking challenges and future direction: A case study of implementing SDN features on OpenStack private cloud // IOP Conf. Series: Materials Science and Engineering. – 2016.
6. Sezer, S., Scott-Hayward, S., Chouhan, P. K., Fraser, B. Are We Ready for SDN? Implementation Challenges for Software-Defined Networks // IEEE Communications Magazine, 51(7), 36-43. – 2013.

Literature

1. Hend Abdelagder Eissa, Kenz A. Bozed, Hadil younix. Software Defined Networking // International conference on Sciences and Techniques of Automatic control & computer engineering (STA). – October 2019.
2. Saleh Asadollahi, Dr. Bhargavi Goswami, Dr. Atul M Gonsai. Software Defined Network, Controller Comparison // International Journal of Innovative Research in Computer and Communication Engineering. – Vol.5. – April 2017.
3. Klepche V. Software-defined networks today // Connect WIT. – №1-2. – 2019.
4. Panesh A.Kh. Content and prospects of technologies of software-defined networks and virtualization of network functions // Bulletin of the Adyghe State University. – 2014. – 2. – 120-127.

5. Veeramani Shamugam, I Murray, J A Leong, Amandeep S Sidhu. Software Defined Networking challenges and future direction: A case study of implementing SDN features on OpenStack private cloud // IOP Conf. Series: Materials Science and Engineering. – 2016.
6. Sezer, S., Scott-Hayward, S., Chouhan, P. K., Fraser, B. Are We Ready for SDN? Implementation Challenges for Software-Defined Networks // IEEE Communications Magazine, 51(7), 36-43. – 2013.

© Фёдоров Н. К., 2022. Научно-образовательный журнал для студентов и преподавателей «StudNet» №4/2022

Для цитирования: Фёдоров Н. К. ПРОГРАММНО-КОНФИГУРИРУЕМЫЕ СЕТИ. ПРОБЛЕМЫ ПРИ ПЕРЕХОДЕ К СЕТЯМ ПКС// Научно-образовательный журнал для студентов и преподавателей «StudNet» №4/2022