

*Алексеева К. Ю.  
Магистрант 2 курс  
Поволжский университет телекоммуникации и информатики  
Россия, г. Самара*

## **АНАЛИЗ РАБОТЫ ИНТЕРФЕЙСОВ В ПРОГРАММНО- КОНФИГУРИРУЕМЫХ СЕТЯХ**

*Аннотация: в статье рассмотрены интерфейсы программно-конфигурируемых сетей, каждый из которых отвечает за свой поток данных от логического центра сети – контроллера к другим плоскостям. Выделены основные стандартные интерфейсы, используемые на данный момент времени, протоколы, используемые ими и стандарты. Предоставлена краткая информация об их работе.*

*Ключевые слова: программно-конфигурируемая сеть, южный интерфейс, северный интерфейс, OpenFlow, OpenDaylight*

*Alekseeva K. Y.  
2d year master's student  
Povolzhsky University of Telecommunications and Informatics  
Russia, Samara city*

## **ANALYSIS OF INTERFACES WORK IN SOFTWARE-DEFINED NETWORKS**

*Abstract: the article considers the interfaces of software-configurable networks, each of which is responsible for its own data flow from the logical center of the network – controller to other planes.*

*The main standard interfaces that are currently used are highlighted, their protocols and standards. Brief information about their work is provided.*

*Keywords: software-configurable network, south interface, north interface, OpenFlow, OpenDaylight*

Отличие программно-конфигурируемых сетей в том, что их интерфейс открыт, контроллер является логическим центром, юг отвечает за связь с уровнем данных, а север - за связь с прикладным уровнем, приложениями.

В исследованиях открытого интерфейса южный интерфейс контроллера широко изучался как ключевой элемент разделения передачи данных и управления данными, в итоге стал объектом пристального внимания можно сказать всей отрасли. Из-за нынешнего разрыва уровня управления и уровня данных улучшения для этих двух плоскостей стали независимы, относительно независимы. Между уровнями должны быть предусмотрены только стандартные южные интерфейсы. Южный интерфейс является ключевым элементом многоуровневой архитектуры ПКС, однако логически он должен не только обеспечивать нормальную связь между уровнем данных и уровнем управления, но также и поддерживать каждый из них. Уровень обновляется независимо, вследствие у производителей техники возникает необходимость в разработке оборудования, что будет обеспечивать поддержку данного интерфейса. Так как наша нынешнее традиционное сетевое оборудование не может работать в сети SDN, поэтому разработка стандартных южных интерфейсов стала важной частью фундаментальных исследований SDN.

Многие организации начали разрабатывать стандартные «южные» интерфейсы. CDPI, предложенный ONF, стал основным интерфейсом, использующим протокол OpenFlow. OpenFlow является первым широко используемым протоколом интерфейса уровня управления данными в SDN и получил широкое внимание научного сообщества, протокол превращает единое интегрированное и закрытое сетевое устройство в гибкое и управляемое коммуникационное устройство. Протокол OpenFlow основан на концепции управления потоком в соответствии с правилами, поэтому

коммутатору необходимо поддерживать таблицу потоков для поддержки OpenFlow и пересылать данные в соответствии с составляемой таблицей.

Каждая запись в такой таблице содержит проверяемые поля пакета, а также какое-либо действие. Стандартно используемые для таких случаев отправить, отбросить или изменить. В формате запросов воспроизводятся как `send-out-port`, `drop` и `modify-field`. В случае если мы получим пакет, не попадавший ранее и для которого не известна информация, то происходит переадресация данного пакета ядру сети, то бишь контроллеру, который в свою очередь определяет как должны быть обработаны поступившие данные. Создать запись в таблице для отправки пакета в случае возникновения схожих ситуаций или отбросить его.

Для того, чтобы способствовать развитию производителей поддержки устройств оборудования, использующих протокол OpenFlow был представлен стандарт OpenFlow.

Казалось бы, раз уж речь идет об открытом стандарте, то достаточно себя огородить от неминуемо появляющихся проприетарных решений, как в рамках одной версии стандарта все станет прекрасно. Но не тут-то было. Из-за несовершенства железа в разных режимах функционирования изрядное количество проверок полей и даже действий может просто не поддерживаться на уровне коммутационной матрицы. А самое печальное, что узнать информацию такого уровня крайне сложно, поскольку производители крайне неохотно выкладывают ее в открытый доступ. И тут мы плавно переходим к следующему пункту.

Если мы выносим план управления из коммутаторов — они должны стать проще. Так, да не совсем: новый протокол, а точнее огромное количество учитываемых по умолчанию полей в таблицах маршрутизации выдвигает повышенные требования к объемам памяти на коммутаторах. В современных высокоскоростных коммутаторах для хранения таблиц используется TCAM (Ternary Content Addressable Memory) — это единственный адекватный по скорости поиска информации способ

хранения данных, от которого невозможно никуда деться, если мы хотим заниматься коммутацией и маршрутизацией на скоростях интерфейса. И стоит TCAM достаточно дорого. Поэтому даже коммутаторы на мощных относительно современных ASIC'ах порой могут поддерживать OpenFlow исключительно формально, обеспечивая менее тысячи OF-записей в TCAM. В принципе, можно было бы организовать многоуровневое хранение записей, сделав в TCAM буфер для самых актуальных записей и сложив в оперативную память все остальное, но для этого необходимо весьма ощутимо модифицировать платформу коммутаторов, большинство существующих моделей на такое просто не способно.

И это, не говоря о том, что многие ASIC в принципе не способны поддерживать часть заявленных в OpenFlow функций, поскольку попросту не имеют их поддержки. Конечно, этот недостаток легко купируется добавлением к ASIC сетевых процессоров (NPU) с высокой степенью программируемости, но это сразу влечет ощутимый рост как энергопотребления, так и стоимости. А это уже, в свою очередь, вызывает вопрос о целесообразности всей затеи, поскольку вместо упрощения и удешевления конечных устройств мы получим строго обратное.

Большую часть вопросов, озвученных выше, можно успешно решить на стадии проектирования сети, тщательно подойдя к выбору оборудования. Хорошо выбранный контроллер, продуманная архитектура, подобранные коммутаторы и работоспособная сеть на OpenFlow становится реальностью, хотя, конечно, до повсеместного внедрения ему еще далеко. Но строить небольшие высокопроизводительные узкоспециализированные сетевые структуры на OpenFlow уже можно без вовлечения существенных усилий разработчиков.

В дополнение к исследованию южного интерфейса, северный интерфейс контроллера и восточно-западный интерфейс между контроллерами также являются предметом исследования. Северный интерфейс отвечает за связь между уровнем, на котором расположен

контроллер и различными приложениями. Сетевые абстрактные ресурсы требуются для управления информацией всей сети, что удобно для пользователей, чтобы быстро продвигать услуги по настройке сети и развертыванию приложений, однако из-за разнообразия сервисов приложений северный интерфейс также демонстрирует разнообразие требований, которым сложно удовлетворить. Обычно каждый контроллер имеет свой собственный северный интерфейс, так как стандарт до сих пор не был разработан.

Изначальная конфигурация SDN подразумевало вмешательство пользователей в работу, для персональной настройки под личностные потребности, для различных сценариев применения. На основе этого следует, что унифицированный стандарт северного интерфейса будет напрямую влиять на более строгую и организованную разработку различных прикладных служб. Чтобы унифицировать северный интерфейс, организации также, как и для южного стали разрабатывать стандарты северного интерфейса. OpenDaylight считается контроллером имеющим наиболее стандартизированный вариант, также известен NBI от ONF. Большая часть контроллеров, к примеру, Floodlight, Trema, NOX используют свои собственные северные интерфейсы.

Тем не менее, такие стандарты только описывают функции и в их задачу не входит определение и подробный разбор метода реализации. Создание и внедрение единого стандарта северного интерфейса является еще одной крупным отраслевым направлением.

В отличие от вышеперечисленных интерфейс восток-запад отвечает за связь между контроллерами. Стоит помнить о том, что наши мощности не бесконечны, так и у контроллера ограничена его производительность. Разработка стандартов интерфейса восток-запад позволяет контроллеру масштабироваться, а также обеспечивает технические гарантии для балансировки нагрузки и повышения производительности.

### *Список литературы/ References*

1. Электронный ресурс: book.itер–  
<http://book.itер.ru/4/41/openflow.htm>
2. Электронный ресурс: SIGCOMM –  
<http://ccr.sigcomm.org/online/files/p69-v38n2n-mckeown.pdf>
3. Электронный ресурс: CYBERLENINKA –  
<https://cyberleninka.ru/article/n/sravnitelnyy-analiz-sdn-kontrollerov/viewer>
4. Электронный ресурс: THENEWSTACK –  
<https://thenewstack.io/sdn-series-part-iii-nox-the-original-openflow-controller/>