Информационные процессы, Том 23, № 1, 2023, стр. 113–125.

ПЕРЕДАЧА ИНФОРМАЦИИ В КОМПЬЮТЕРНЫХ СЕТЯХ

Система управления трафиком в перспективных мобильных сетях на основе технологий SDN/NFV[1]

В.М. Антонова[*,], И.Г. Бужин[*], Э.А. Гайфутдинов[*], В.С. Гнездилов[*], Н.А. Кузнецов[**], Ю.Б. Миронов[*]**

[*] *Московский технический университет связи и информатики, Москва, Россия;*
[**] *Институт радиотехники и электроники (ИРЭ) Российской академии наук, Москва, Россия;*

Поступила в редколлегию 07.02.2023

Аннотация—Задачи управления трафиком в условиях перегрузки решались многими поколениями, однако, в перспективных системах передачи данных возможны использования новых решений управления, таких как разделение сетевых элементов на сетевые слои и реализация сетевых элементов в виде виртуальных сетевых функций. В перспективных сетях региональные центры представляют собой программные контроллеры, которые управляют локальными сенсорами. Все управляющие центры связаны между собой через Транспортную подсистему и образуют сеть. Наличие сенсоров в узлах сети обеспечивает контроль пропускной способности различного вида трафика. Одной из важнейших задач управления трафиком является его фильтрация. В перспективных системах данная задача решается на сетевых сенсорах с помощью реализации алгоритмов поиска соответствий информации о трафике и правил фильтрации, имеющие большую временную сложность, которую в них можно сократить с помощью технологии фильтрации.

В статье предложен подход к использованию концепции программно-конфигурируемых сетей и виртуализации сетевых функций для реализации системы управления трафиком в перспективных мобильных сетях. В качестве сетевых сенсоров предлагается использовать коммутаторы, работающие на основе протокола OpenFlow. Также предложен способ обработки и фильтрации трафика в транспортных сетях, который реализован в виде набора библиотек и драйверов для взаимодействия сетевых элементов напрямую с приложениями, позволяющий уменьшить время поиска нужных правил на телекоммуникационном оборудовании.

КЛЮЧЕВЫЕ СЛОВА: Программно-конфигурированная сеть, Виртуализация сетевых функций, система управления трафиком и фильтрации, балансировка нагрузки, модифицированный алгоритм фильтрации.

DOI: 10.53921/18195822_2023_23_1_113

1. ВВЕДЕНИЕ

Под мониторингом сети обычно понимают постоянный сбор данных о функционировании компьютерной сети для выполнения таких функций, как: поиск медленных, неисправных или, наоборот, недогруженных систем, а также основных потребителей сетевых ресурсов, выполнение параметров SLA (соглашения об уровне сервиса) и качества предоставляемой услуги (например, доставки по требованию аудио, видео или иного содержимого, задержки при соединении распределённых и интегрированных систем). Наряду с перечисленными выше техническими аспектами под мониторингом также понимают надзор за работой сети, в значении контроля соблюдения политик доступа, информационного обмена и маршрутизации. Собранные данные могут отражать разные стороны функционирования сети в зависимости от цели

[1] Работа выполнена по государственному заданию ИРЭ им. В. А. Котельникова РАН (FFWZ-2022-0006).

и задач мониторинга. Оба предложенных выше значения термина мониторинг близки друг к другу и опираются на одни и те же технологии сбора и анализа информации.

Мониторинг в сетях переданных данных 5G и 6G (опорная сеть ядра, Midhaul, Backhaul) с помощью NetFlow/IPFIX вводит значительную задержку измерения [5, 9, 10], потому что данные по измерениям не сообщаются до тех пор, пока не завершится измеряемое соединение, что делает длину задержки пропорциональной продолжительности соединения. Метод выборки пакетов sFlow вводит значительно меньшую задержку, так как выборочные пакеты сразу же доступны для анализа трафика и, следовательно, позволяют быстро обнаруживать большие потоки. По этой причине sFlow целесообразно использовать для обнаружения аномалий в средах с использованием технологии программно-конфигурируемых сетей (SDN), хотя его механизм на основе выборки обеспечивает меньшую точность, если отбирается недостаточно пакетов. Развитие SDN и NFV подводит к использованию протокола OpenFlow (OF) для целей мониторинга [6, 7] в первую очередь сетей передачи данных, использующие SDN и NFV (опорная сеть ядра, Midhaul, Backhaul) [17]. Основной метод использования — интеграция с известными ставшими традиционными инструментами мониторинга путём перевода атрибутов, собираемых протоколом OF, в NetFlow или IPFIX. Необходимо определить подход к использованию преимуществ протокола OF и оборудования, поддерживающего его, для мониторинга традиционных сетей.

Распространенным средством для реализации правил обработки и фильтрации правил в крупных сетях является Ipfirewall [23] — open source модуль, портированный на многие ОС. ipfw — пользовательская утилита для управления ipfirewall. С помощью этой утилиты происходит взаимодействие с модулем ядра. Особенностями обработки правил в ipfw являются обработка правил сверху вниз, порядок следования правил важен, присутствует наличие возможностей оптимизации функционирования через деревья и хеш–таблицы. Другим решением для фильтрации правил является iptables/nftables — утилиты Linux, обеспечивающие фильтрацию и классификацию сетевых пакетов/датаграмм/кадров. Правила объединяются в цепочки, которые, в свою очередь, входят в состав таблиц. Таблицы содержат цепочки. Количество таблиц и их имена определяется пользователем. Тем не менее, каждая таблица имеет только одно семейство адресации и применяется к пакетам только этого семейства. Таблицы могут относиться к одному из пяти семейств. Цепочка содержит правила. При работе данные утилиты проверяют правила в цепочке последовательно, сверху вниз. Кроме того, поскольку правила имеют определённую «стоимость» выполнения, не стоит изменять их порядок исключительно на основе эмпирических наблюдений за счётчиком байтов/пакетов. Из–за отсутствия возможности указывать несколько разных действий в одном правиле, утилиты iptables/nftables имеют низкую производительность и не подходят, как и Ipfirewall, для обработки большого количества (порядка 30000) правил фильтрации. В данной статье предлагается способ фильтрации трафика для уменьшения времени нахождения соответствий правил фильтраций на высокоскоростных участках сети (100G+).

Применение SDN для фильтрации трафика и способы оптимизации нахождения правил рассматриваются в статьях [20–22, 24, 31]. В данных работах предлагается при получении коммутаторами неизвестных пакетов, передавать их по заранее запланированным маршрутам, используя для этого «правило по умолчанию». Такой подход может уменьшить время поиска нужного правила для пакета, но «правило по умолчанию» может направить потоки по неоптимизированному пути или заблокировать его. В [25] предлагается минимизировать количество управляющих сообщений за счет агрегирования правил. Данный подход также позволит сократить время поиска нужного правила, но при этом контроллер будет иметь обобщенную информацию об уровне передачи данных, что может привезти к ухудшению QoS и увеличе-

нию угроз информационной безопасности. Необходимо решение, которое оптимизирует работу правил коммутаторов OpenFlow при этом не изменит принцип их работы.

Таким образом, необходимо описать подход для реализации системы управления и фильтрации трафика в перспективных мобильных сетях в варианте распределённой иерархической схемы. Для уменьшения времени поиска правил фильтраций в данном подходе необходимо предложить способ обработки и фильтрации трафика в транспортных сетях 5G и 6G.

Статья организована следующим образом: во введении рассматриваются принципы организации мониторинга в сетях связи, исследуются недостатки способов мониторинга и предлагаются возможные методы их устранения; в разделе 2 рассматриваются принципы и особенности построения сетей связи на основании технологий SDN/NFV; в разделе 3 анализируется архитектура системы управления и фильтрации трафика в перспективных мобильных сетях на основе технологии SDN/NFV; в разделе 4 предложен способ обработки и фильтрации трафика в транспортных сетях 5G и 6G, приведено сравнение разработанного способа с общеизвестными алгоритмами; в заключении приведены основные выводы.

2. ПРИНЦИПЫ И ОСОБЕННОСТИ ПОСТРОЕНИЯ СЕТЕЙ СВЯЗИ НА ОСНОВЕ ТЕХНОЛОГИЙ SDN/NFV.

Технология SDN/NFV представляет собой набор методов, которые позволяют программным способом управлять сетевыми ресурсами и контролировать их использование (загрузку), что упрощает решение задачи обеспечения эффективного использования пропускной способности сети связи и ее масштабирования, содействует снижению эксплуатационных затрат за счет централизации и автоматизации функций управления. В архитектуре SDN выделяют 3 уровня (Рис. 1): уровень инфраструктуры сети, уровень управления и уровень приложений. ПКС предусматривает наличие в сети контролера ПКС, который предоставляет приложениям абстрактное представление сетевых ресурсов и обеспечивает оркестрацию (координацию) управления сетевыми ресурсами. При таком подходе контроллер имеет доступ к глобальному состоянию сети и принимает решение о пересылке сетевого трафика, в то время как аппаратное оборудование отвечает только за фактическую пересылку информации к местам их назначения в соответствии с указаниями контроллера (наборами правил обработки пакетов). Таким образом, функция коммутатора по управлению переносятся на отдельное центральное устройство — контроллер ПКС. Такой подход позволяет управлять и контролировать за состоянием сети на логически централизованном контроллере. Кроме того, появляется возможность уровню управления отделиться от физической составляющей, используя логическое представление сети в целом. Взаимодействие между уровнем передачи данных осуществляется посредством единого унифицированного открытого интерфейса.

С учетом перечисленных особенностей основными компонентами программно-конфигурируемых сетей на базе протокола OpenFlow являются: OpenFlow коммутатор [13], Контроллер [18] или Сетевая операционная система, Протокол управления коммутаторами, Протокол администрирования сетевого оборудования, Защищенный канал, по которому с помощью OpenFlow протокола осуществляется взаимодействие контроллера и коммутатора.

Коммутатор снабжен набором таблиц адресаций потоков (flow tables), образующих конвейер адресации (pipeline), который состоит из одной или нескольких последовательно соединенных таблиц адресаций [14,15]. Пакет, поступивший на один из входных портов коммутатора, сначала обрабатывается (считывается служебная информация из пакета), затем поступает на конвейер адресации. Начинается последовательная обработка пакета в таблицах адресаций. Под пакетом здесь понимается битовая строка, из которой можно выделить две части: заголовок и полезную нагрузку. Операции, выполняемые над потоками пакетов в таблицах адресаций, не изменяют нагрузку пакета, но способны изменять его заголовок.

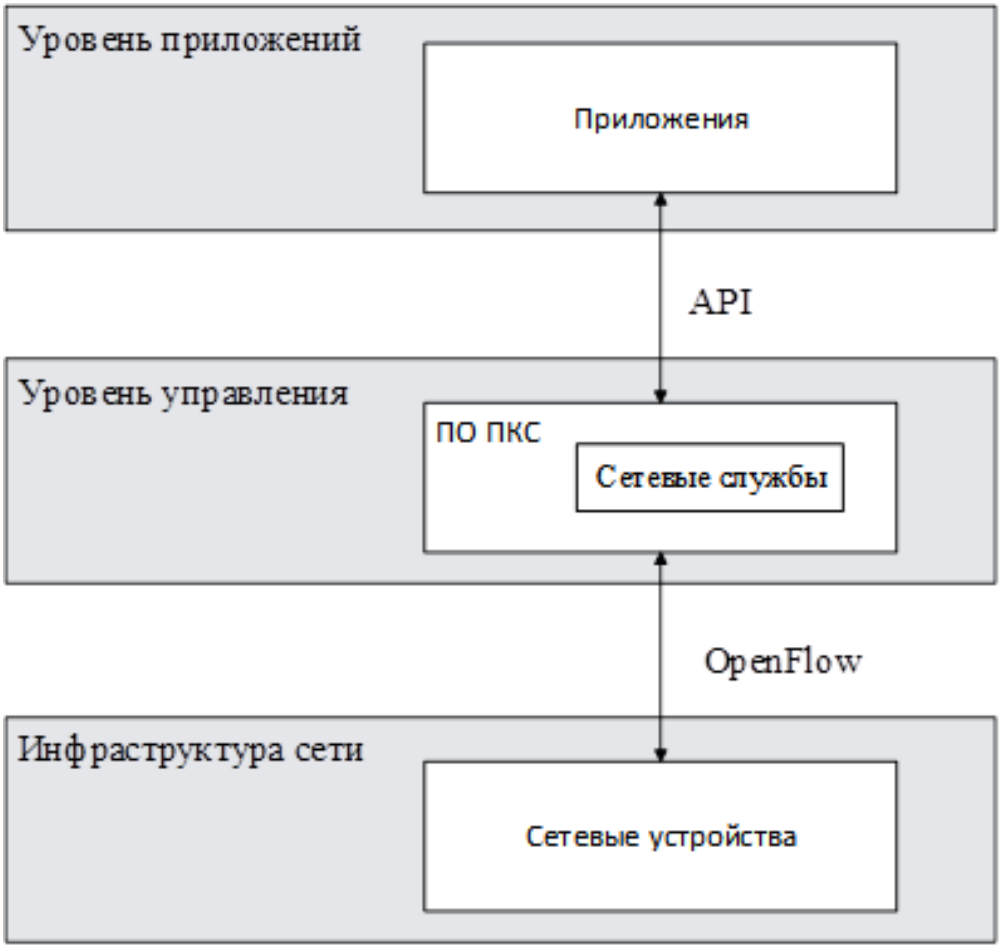

Рис. 1. Архитектура SDN

Заголовок пакета состоит из нескольких полей. Как правило, в этих полях указываются идентификаторы сетевых протоколов, которые должны обрабатывать пакет, и используемая ими служебная информация. Далее начинает пересылка информации внутри коммутатора ПКС, которая основана на потоках. Поток представляет собой последовательность пакетов между источником и пунктом назначения с одинаковыми политиками обслуживания. Абстракция потока позволяет объединить поведение различных типов сетевых устройств. В процессе прохождения потока через конвейер таблиц адресаций к заголовкам пакетов потока могут быть добавлены служебные поля (metadata), которые предназначены для передачи служебной информации внутри конвейера, и сбрасываются при поступлении пакета в один из выходных портов коммутатора. Состав и размер служебных полей определяется техническими характеристиками конкретного коммутатора ПКС. Таким образом, решения о передачи данных принимаются на основе потоков, которые представляют собой сочетание полей заголовков пакета. Коммутатор ПКС обрабатывает каждое сообщение, полученное от контроллера, с возможностью формирования ответа, если это необходимо. Если коммутатор не может полностью обработать сообщение, полученное от контроллера, он должен послать обратно контроллеру сообщение об ошибке. Это может произойти из–за перезагрузки коммутатора, QoS политики или если сообщение отправлено на заблокированный или неисправный порт.

Записи потока соответствуют пакетам в порядке приоритета, при этом первая подходящая запись используется в каждой таблице. В случае использования Потоковых Таблиц происхо-

дит и не коммутация (пересылка на уровне L2), и не маршрутизация (на уровне L3). Нам необходим новый термин, который будет описывать такой механизм — Пересылка Потока или Flow Forwarding. Суть термина заключается в том, что поиск на уровне 2 и 3 требует большой обработки. Пересылающее сетевое устройство не предполагает, что точка назначения может быть одинаковой для группы одинаковых пакетов. Каждый из пакетов должен быть оценен для правильного принятия решения о пересылки. Потоки же обрабатываются иным способом. Они содержат информацию о состоянии. У них есть информация, которая может помочь устройству для принятия решения в сложных ситуациях. Эти решения не ограничены выбором на основе MAC или IP адреса назначения. Они могут быть основаны на метках, или VLAN ID, или другой идентификационной информации. Это может быть информация, о которой приложение договорилось с другими устройствами, как например, информация DNS. Такой подход позволяет принять единое решение о передаче для всего потока и быстро применить его.

Таким образом, возможности протокола OpenFlow в сочетании с надлежащими приложениями контроллера способы собирать и накапливать все данные, для сбора которых ранее использовались протоколы IPFIX, NetFlow или sFlow. Эта гибкость ПКС сети и открытые программные интерфейсы позволяют получить явные преимущества в развитии возможностей мониторинга сети.

3. АРХИТЕКТУРА СИСТЕМЫ УПРАВЛЕНИЯ И ФИЛЬТРАЦИИ ТРАФИКА В ПЕРСПЕКТИВНЫХ МОБИЛЬНЫХ СЕТЯХ

Рассмотрим варианты реализации системы управления и фильтрации трафика (СУиФТ) в виде полностью централизованной схеме и в виде распределённой схемы. В полностью централизованной системе все функции фильтрации, классификации, передачи и хранения данных, а также управление этими функциями происходит из единого центра. В распределённой иерархической функциональность и управление СУиФТ рассредоточены по всем комплексам средств автоматизации распределённой системы СУиФТ с целью масштабируемости и резервирования последней.

Применение технологии SDN позволяет выполнить СУиФТ в варианте распределённой иерархической схемы. Технология ПКС позволяет гибко организовывать иерархию и подчиненность подсистем, серверов обработки и хранения, их уровни и степень подчинения. Также к ключевым особенностям такого подхода стоит отнести возможность динамически управлять как самой организацией иерархии и подчинения, так и степенью резервирования, отказоустойчивости её подсистем и компонентов. Для выполнения рассмотренных требований СУиФТ состоит из подсистем:

- Подсистемы фильтрации, классификации и хранения;
- Подсистемы управления СУиФТ;
- Транспортной подсистемы ? системы передачи данных между сенсорами, региональный и главный центры обработки и хранения данных (РЦОХД и ГЦОХД).

Одним из основных элементов СУиФТ является Подсистема фильтрации, классификации и хранения данных. Источником информации для решения задач мониторинга можно считать сенсор, в котором происходит первичный анализ — фильтрация данных сети. Сенсор — это элемент СУиФТ, расположенный непосредственно на прослушивание транспортной сети, мониторинг которой осуществляет СУиФТ, далее контролируемая транспортная сеть (КТС). С целью сокращения нагрузки на транспортную сеть СУиФТ целесообразно возложить функции фильтрации на сенсор. Это позволит отсечь ненужный для дальнейшего анализа трафик

КТС от транспортной сети СУиФТ, снизить нагрузку как на саму транспортную сеть, так и на средства хранения и анализа СУиФТ.

Применение концепции SDN для организации подсистемы фильтрации представляется самым целесообразным. Функция фильтрации предполагает отбор пакетов, циркулирующих в КТС, на основе значения полей заголовков пакетов на уровнях L2–L4. Организация сенсора на принципах SDN, а именно, в виде SDN коммутатора, работающего на основе протокола OpenFlow, позволит избавиться от таких проблем как большой объем буферного накопителя, большое количество сетевых интерфейсов. Такой сенсор позволяет динамически фильтровать трафик и направлять его в соответствии с правилами фильтрации на указанные порты коммутатора. Все вышесказанное является обоснованием того, почему целесообразно применять SDN подход в организации сенсора. Таким образом, базовым элементом рассматриваемой сети мониторинга предлагается использовать сенсор, реализованный на основе SDN коммутатора с поддержкой протокола OpenFlow.

Система хранения в СУиФТ должна иметь иерархическую организацию и строится по региональному принципу, т. е. есть РЦОХД и есть ГЦОХД, связанные через транспортную сеть СУиФТ (Транспортную подсистему). РЦОХД выполняют функцию промежуточной буферизации и сглаживания временной нехватки пропускной способности от сенсоров до РЦОХД, а также контур управления РЦОХД может принимать участие в управлении подсистемы фильтрации. Информация в каждом РЦОХД и ГЦОХД должна быть согласованной и консистентной. Это говорит о том, что для построения ЦОХД необходимо применять технологии распределенных сетевых хранилищ данных (РХД).

Логически централизованный контур управления СУиФТ образуют контур управления ГЦОХДБ, контуры управления РЦОХД и контур управления транспортной сети, обеспечивающей доставку данных от сенсоров в ГЦОХД через РЦОХД, команд управления сетевыми устройствами контура данных СМУИБ, доставку служебной информации в СМУИБ. Контур управления СУиФТ должен быть отказоустойчив, обеспечивать непрерывность доступа сенсоров к сервисам РЦОХД и ГЦОХД. Важным аспектом распределенности контура управления СУиФТ является эффективность масштабирования системы мониторинга при изменении масштабов сети передачи данных, для мониторинга которой используется СУиФТ. Другим важным аспектом распределенности контура управления СУиФТ является повышение уровня его безопасности, а именно то, что сеть передачи данных между компонентами контура управления СУиФТ изолирована от внешнего мира и может быть атакована только изнутри. Подсистема управления СУиФТ представляет собой многоуровневую сетевую модель (Рис. 2), состоящую из:

– Главного Центра мониторинга СУиФТ, расположенного в ГЦОХД (главный кластер ПКС контроллеров);
– Региональных центров мониторинга СУиФТ, расположенных в РЦОХД.

Главный Центр мониторинга СУиФТ и Региональные центры мониторинга связаны между собой через Транспортную подсистему и образуют сеть. С целью отказоустойчивости, балансировки нагрузки, обеспечения сетевой связности необходимо предусмотреть наличие как вертикальных, так и горизонтальных сетевых связей РЦОХД между собой, РЦОХД и ГЦОХД, как показано на рисунке. Подсистема управления логически организует подчиненность среди ГЦОХД и РЦОХД в соответствии с принятой политикой управления СУиФТ, гибко выстраивая логически подчиненность путем программной конфигурации каналов для связи Г(Р)ЦОХД между собой. При возникновении нештатных ситуаций: выхода из строй отельных РЦОХД, обрыва физических соединений в сети СУиФТ, используемых в сконфигурированной модели потоков данных, внезапной нехватки ресурсов отдельного РЦОХД или линий связи, — Подсистема управления в состоянии динамически перестроить маршруты и

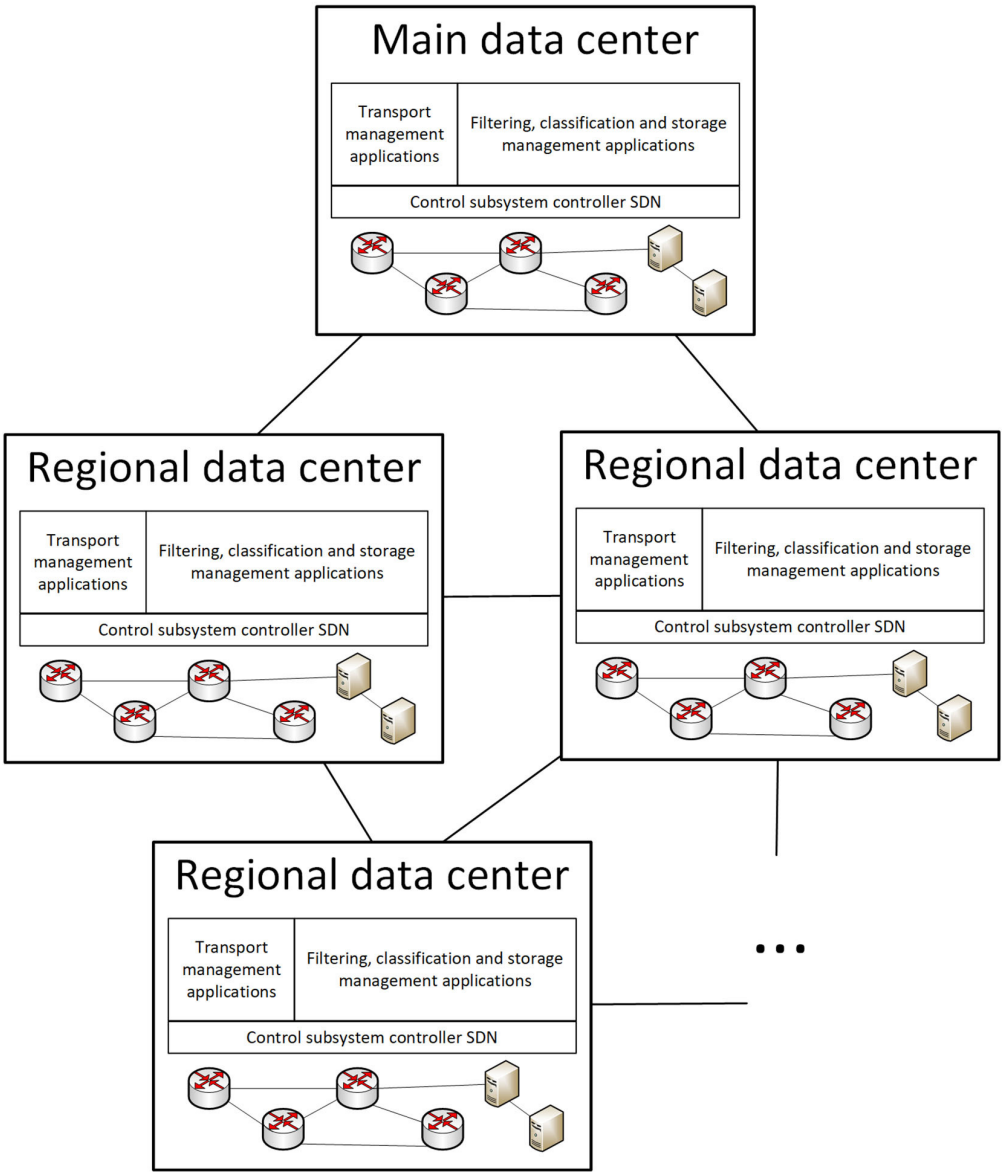

Рис. 2. Система управления трафиком в перспективных мобильных сетях на основе технологий SDN/NFV

распределить нагрузку по сети, обеспечив выполнение заданного уровня обслуживания всей СУиФТ.

Таким образом, Главный Центр мониторинга СУиФТ включает в себя главный кластер SDN контроллеров — логически-централизованный, но физически распределенный контроллер. Этот кластер работает по схеме резервирования Active/Active, то есть для каждого регионального центра (одного или нескольких) выделен отдельный курирующий SDN контроллер, в случае его отказа, управление регионом передается другому из оставшихся функционирующих контроллеров главного кластера SDN контроллеров. Каждый контроллер главного кластера, ответственный за какой–то регион, представляет собой приложение для кластера контроллеров регионального центра.

4. ОБРАБОТКА И ФИЛЬТРАЦИЯ ТРАФИКА В ТРАНСПОРТНЫХ СЕТЯХ 5G И 6G

Обработка и фильтрация внешнего трафика на границах сети является важной задачей системы фильтрации трафика. Данное решение должно быть программным, которое устанавливается и работает на стандартных аппаратных сетевых решениях. Но количество аппаратных устройств ограничено, и в случае масштабирования — рост количества устройств будет нелинейным. При этом осуществление обработки и фильтрации трафика на границах сети является ответственным процессом: при возникновении ошибки в правилах фильтрации возможен частичных или полный выход из строя внутренней сети. При этом суммарно скорость обработки и фильтрации трафика должна быть от 300 Gbit/s и выше.

При возрастании сложных правил до порядка 30000, а также при появлении в правилах так называемых сетей с Wildcard mask — это маска, которая показывает какая часть (сколько бит) IP адреса могут меняться. Она может применяться при объявлении сетей в протоколах маршрутизации таких как IGRP, EIGRP, OSPF, в списках доступа. Принцип работы маски тоже такой же, как и у сетевой маски алгоритма CIDR, за исключением того, что вместо единиц ставятся нули, а вместо нулей единицы. Написание алгоритмов для обработки такого количество правил и с такими особенностями является сложным и трудным процессом.

Для удовлетворения скорости по обработки трафика (100G+), универсальности написания алгоритмов фильтраций, независимости от производителя аппаратной платформы для реализации системы фильтрации транспортной сети 5G и и разрабатываемых сетей следующих поколений рекомендуется использовать технологию DPDK [26], при работе которой взаимодействие с сетевой картой осуществляется через специализированные драйверы и библиотеки. Поступившие пакеты попадают в кольцевой буфер. Приложение периодически проверяет этот буфер на наличие новых пакетов. Если в буфере имеются новые дескрипторы пакетов, приложение обращается к буферам пакетов DPDK, находящимся в специально выделенном пуле памяти, через указатели в дескрипторах пакетов. Если в кольцевом буфере нет никаких пакетов, то приложение опрашивает находящиеся под управлением DPDK сетевые устройства, а затем снова обращается к кольцу.

При применении DPDK для обработки и фильтрации трафика предполагается отделение Data plane от Control plane (Рис. 3). Задачей Data plane является перекладывание пакетов из очереди в очередь, их модификация. Control plane отвечает за динамическую настройку и контроль (настройка логических портов, определение MAC адресов, определение IP адресов, настройка VLAN, настройка маршрутов BGP, компиляция ACL правил оптимальным образом, Сборка и агрегация метрик и таблиц внутреннего состояния Data Plane) Data plane. Задача

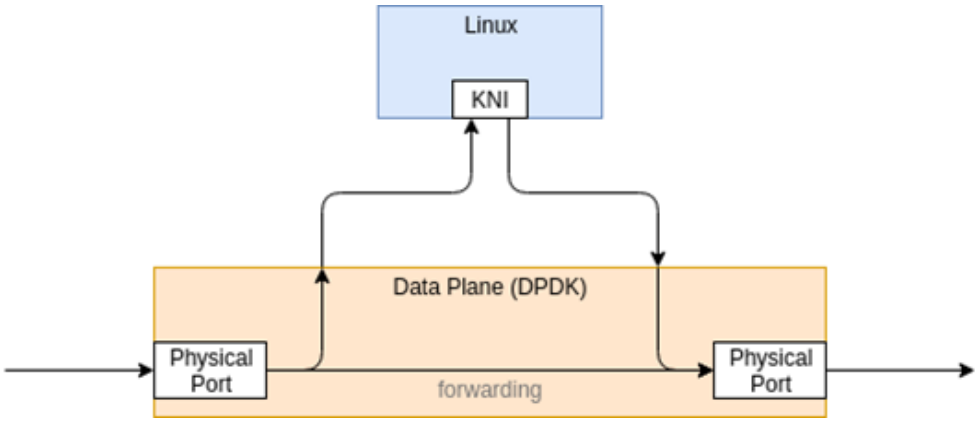

Рис. 3. Пример использования DPDK

состоит в том, что на транспортных сетях 5G и 6G в ACL может быть большое число правил, при поступлении пакета на фильтр в независимости от размера набора правил и сохранении логики порядка правил необходимо наиболее быстрым способом найти нужное соответствие (Задача поиска подмножества нужных правил в огромном множестве всех правил).

Содержание правил ACL можно классифицировать по следующим полям: Протокол, Source Port, Destination Port, Source Net, Destination Net. Таким образом, изначально выделяются несколько исходных подмножеств. Далее, если осуществить перенумерацию битовых масок в каждом классификаторе по порядку и переумножить все пути достижения финальных правил, получится декартово произведение всех множеств классификаторов, мощность (и мощность вычисления) которого будет слишком большой. Выходом из этой ситуации является разбиение исходных множеств на пары/тройки для снижения мощности промежуточных множеств и уменьшения времени на поиск соответствий. Данный подход соответствует алгоритму recursive flow classification (RFC) [19] и схематично изображен на Рис. 4. В RFC [30] большинство слож-

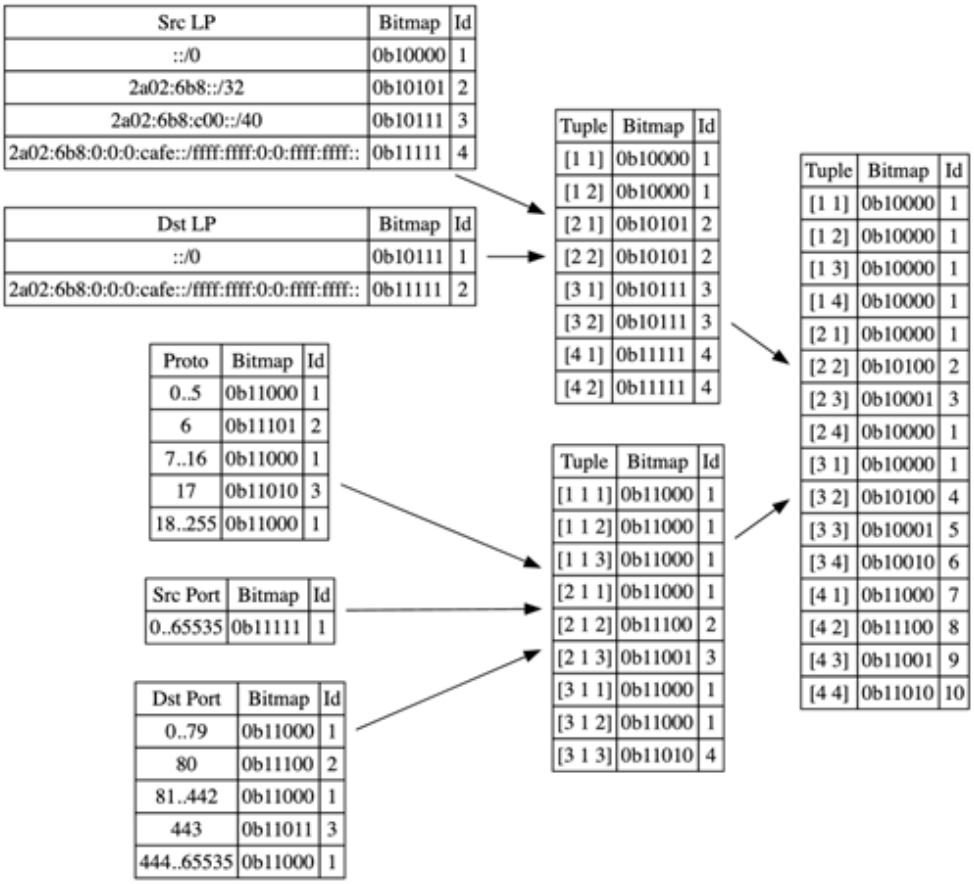

Рис. 4. Поиск соответствий в подмножествах с помощью RFC

ных операций перенесено на стадию предварительной обработки для создания эффективных структур данных для высокоскоростной классификации. Для классификатора размерности k (k — константа) RFC требует постоянного числа обращений к памяти для классификации пакета, т. е. его временная сложность классификации равна $O(k)$. Однако сложность его хранения и предварительной обработки довольно высока, что делает использование RFC при большом количестве памяти.

Для экономии памяти с помощью библиотеки LPM в DPDK реализуется алгоритм Longest Prefix Match (LPM), используемый для пересылки пакетов в зависимости от их IPv4–адреса. Основные функции этой библиотеки заключаются в добавлении и удалении IP–адресов, а также в поиске нового адреса с использованием LPM–алгоритма. Для IPv6–адресов аналогичная функциональность реализована на базе библиотеки LPM6. LPM реализует метод поиска в таблице длиннейших совпадений префиксов (LPM) для 32–битных ключей, который обычно используется для поиска наилучшего совпадения маршрутов (в данном случае — правил ACL) в приложениях IP–переадресации. Основным параметром конфигурации для экземпляров компонентов LPM является максимальное количество поддерживаемых правил. Префикс LPM представлен парой параметров (32–битный ключ, глубина) с глубиной в диапазоне от 1 до 32. Правило LPM представлено префиксом LPM и некоторыми пользовательскими данными, связанными с префиксом. Префикс служит уникальным идентификатором правила LPM. В этой реализации пользовательские данные имеют длину 1 байт и называются следующим переходом в связи с их основным использованием для хранения идентификатора следующего перехода в записи таблицы (множества) правил.

На Рис. 5 представлен пример нахождения соответствия правил в ACL с использованием LPM6. Сравнительный анализ сложности алгоритмов представлен в таблице 1. В итоге, реа-

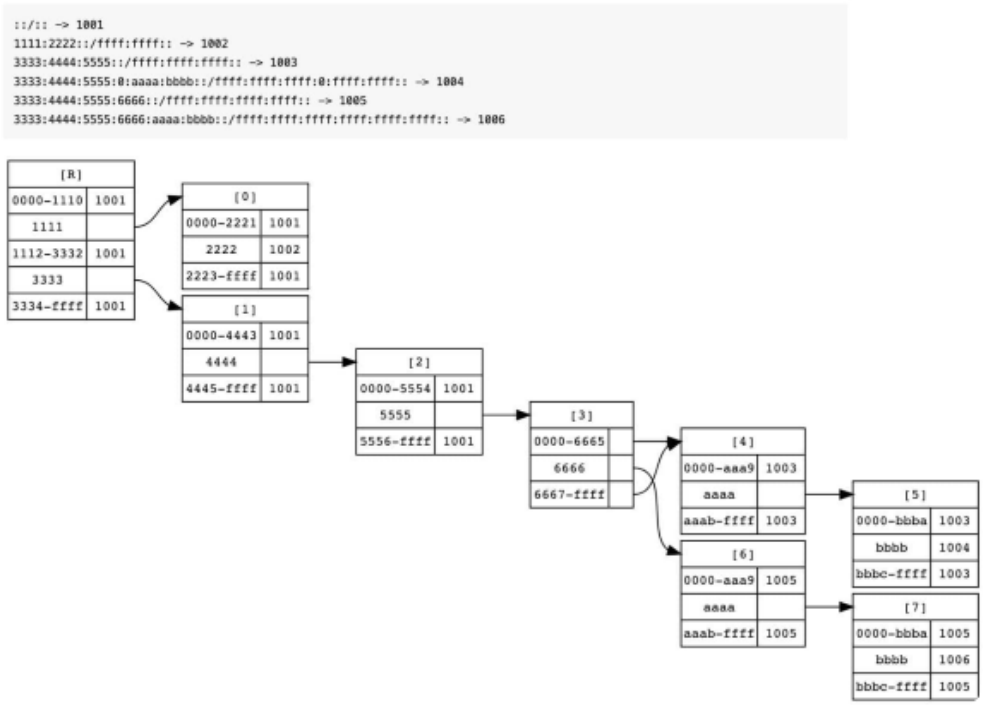

Рис. 5. Пример нахождения правил в ACL с использованием LPM6

Таблица 1. Сравнение сложности алгоритмов фильтрации

Алгоритмы	**Временная сложность**
Common ACL	$O(k \cdot l)$
RFC	$O(k)$
LPM	$O(1)$

лизация ACL на DPDK с использованием алгоритма LPM обрабатывает до 8 мегапакетов в

секунду на 1 CPU ядре, обработка пакета занимает O(1), что значительно ниже алгоритма RFC.

5. ВЫВОДЫ

В данной статье рассмотрены принципы и архитектура системы управления и фильтрации трафика в перспективных мобильных сетях. Применение технологий программно–конфигурируемых сетей и виртуализации сетевых функций позволяют выполнить данную систему в варианте распределённой иерархической схемы. Основными ее элементами являются подсистема фильтрации, классификации и хранения, подсистема управления, транспортная подсистема. Базовым элементом рассматриваемой системы управления предлагается использовать сенсор, реализованный на основе программно–конфигурируемого коммутатора с поддержкой протокола OpenFlow. Для повышения эффективности обработки и фильтрации трафика на транспортных сетях в перспективных мобильных сетях предложен способ реализации алгоритма фильтрации трафика с использованием набора библиотек DPDK и алгоритма LPM, что позволяет обрабатывать до 8 мегапакетов в секунду на 1 ядре процессора. Временная сложность предложенного способа составляет O(1), что значительно ниже аналогичных алгоритмов.

СПИСОК ЛИТЕРАТУРЫ

1. Case J. et al. Simple Network Management Protocol. Request for Comments 1157. Network Information Centre, SRI International. 1990.
2. Claise B. Cisco systems netflow services export version 9. 2004. rfc3954.
3. Phaal P., Panchen S., McKee N. InMon corporation's sFlow: A method for monitoring traffic in switched and routed net–works. 2001.
4. RFC 7426 — Software–Defined Networking (SDN): Layers and Architecture Terminology.
5. Samouylov K. E., Shalimov I. A., Buzhin I. G., Mironov Y. B. Model of functioning of telecommunication equipment for software–concurated networks. Modern Information Technologies and IT-Education. 2018. vol. 14, no. 1. doi:10.25559/SITITO.14.201801.013-026.
6. Ashraf U. Rule minimization for traffic evolution in software–defined networks. IEEE Communications Letters. 2016, vol. 21, no. 4, pp. 793–796 DOI 10.1109/LCOMM.2016.2636212.
7. Assefa B. G., Özkasap Ö. A survey of energy efficiency in SDN: software–based methods and optimization models. Journal of Network and Computer Applications. 2019, vol. 137, no. 4, pp. 127–143. DOI 10.1016/j.jnca.2019.04.001.
8. ETSI GS NFV 002 V1.2.1 «Network Functions Virtualisation (NFV); Architectural Framework», 2014.
9. V. K. Tsvetkov, V. I. Oreshkin, I. G. Buzhin, Y. B. Mironov. Model of Restoration of the Communication Network Using the Technology of Software Defined Networks. ELCONRUS 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1559–1563. doi: 10.1109/EIConRus.2019.8656723.
10. I. G. Buzhin, Y. B. Mironov. Evaluation of delayed telecommunication equipment of Software Defined Networks. SOSG 2019, Institute of Electrical and Electronics Engineers Inc., 2019, p. 8706825. doi: 10.1109/SOSG.2019.8706825.
11. ETSI GS NFV–SEC 003 V1.1.1 (2014–12) Network Functions Virtualisation (NFV); NFV Security; Security and Trust Guidance.
12. OpenFlow Switch Specification version 1.3.0 [Electronic resource] Open Network Foundation. URL: https://www.opennetworking.org/.
13. ONF TR–539 OpenFlow Controller Benchmarking Methodologies [Electronic resource] Open Network Foundation.

14. Bera S., Misra S., Jamalipour A. Flowstat: adaptive flow–rule placement for per–flow statistics in SDN. IEEE Journal on Selected Areas in Communications. 2019, vol. 37, no. 3, pp. 530–539 DOI 10.1109/JSAC.2019.2894239.

15. Zhao G., Xu H., Fan J., Huang L., Qiao C. Achieving fine–grained flow management through hybrid rule placement in sdns. IEEE Transactions on Parallel and Distributed Systems. 2020, vol. 32, no. 3, pp. 728–742 DOI 10.1109/TPDS.2020.3030630.

16. Tsai, C.–C., Lin, F. J. and Tanaka, H. Evaluation of 5G Core Slicing on User Plane Function. Communications and Network. 2021, vol. 13, pp. 79–92. https://doi.org/10.4236/cn.2021.133007.

17. Lin, Y.–B.; Tseng, C.–C., Wang, M.–H. Effects of Transport Network Slicing on 5G Applications. Future Internet. 2021, vol. 13, p. 69. https://doi.org/10.3390/fi13030069.

18. Kotani D., Suzuki K., Shimonishi H. A Design and Implementation of OpenFlow Controller handling IP Multicast with Fast Tree Switching. In Proceedings of the IEEE/IPSJ International Symposium on Applications and the Internet (SAINT), Izmir, Turkey, 16–20 July 2012. p. 60–67.

19. Gupta P., McKeown N. Packet classification on multiple fields. ACM SIGCOMM Computer Communication Review. 1999, vol. 29, no. 4, pp. 147–160.

20. Rawat D. B., Reddy S. R. Software defined networking architecture, security and energy efficiency: a survey. IEEE Communications Surveys & Tutorials. 2016, vol. 19, no. 1, pp. 325–346. DOI 10.1109/COMST.2016.2618874.

21. Nguyen X.–N., Saucez D., Barakat C., Turletti T. Rules placement problem in openflow networks: a survey. IEEE Communications Surveys & Tutorials. 2015, vol. 18, no. 2, pp.1273–1286. DOI 10.1109/COMST.2015.2506984.

22. Yu M., Rexford J., Freedman M. J., Wang J. Scalable flow–based networking with difane. ACM SIGCOMM Computer Communication Review. 2011, vol. 41, no. 4, pp. 351–362. DOI 10.1145/1851275.1851224.

23. Chen Y., Zhu A. The design and implementation of firewall based on FreeBSD. International Conference on Information Science, Electronics and Electrical Engineering. 2014, vol. 1, pp. 657–659.

24. Sheu J.–P., Lin W.–T., Chang G.–Y. Efficient tcam rules distribution algorithms in software defined networking. IEEE Transactions on Network and Service Management. 2018, vol. 15, no. 2, pp. 854–865. DOI 10.1109/TNSM.2018.2825026.

25. Galan–Jimenez J., Polverini M., Cianfrani A. Reducing the reconPиcuration cost of flow tables in energy–efficient software–defined networks. Computer Communications. 2018, vol. 128, no. 2, pp. 95–105. DOI 10.1016/j.comcom.2018.07.02.

26. Cerrato I., Annarumma M., Risso F. Supporting fine–grained network functions through Intel DPDK. Third European Workshop on Software Defined Networks. IEEE. 2014, pp. 1–6.

27. The Data Plane Development Kit (DPDK) software [Electronic resource] URL: https://www.dpdk.org/.

28. Leis V., Kemper A., Neumann T. The adaptive radix tree: ARTful indexing for main–memory databases //2013 IEEE 29–th International Conference on Data Engineering (ICDE). IEEE, 2013, pp. 38–49.

29. LPM Library [Electronic resource] URL:https://doc.dpdk.org/guides-16.04/prog_guide/lpm_lib.html.

30. Gong X. Y., Wang W. D., Cheng S. D. ERFC: an enhanced recursive flow classification algorithm. Journal of Computer Science and Technology. 2010, vol. 25, no. 5, pp. 958–969.

31. Смелянский Р. Л., Антоненко В. А., Концепции программного управления и виртуализации сетевых сервисов в современных сетях передачи данных. 2020. 152 с.

Traffic management system in promising mobile networks based on SDN/NFV technologies

V.M. Antonova, I.G. Buzhin, E.A. Gayfutdinov, V.S. Gnezdilov, N.A. Kuznetsov, Y.B Mironov

In promising data transmission systems, new management solutions can be used, for example, we can divide network elements into network layers or implement network elements in the form of virtual network functions. In promising networks, regional centers are software controllers that control local sensors. All control centers are interconnected through a Transport subsystem and are included in one network. Sensors provide monitoring of the ability of various types of traffic. One of the most important tasks of traffic management is its filtering. In promising systems, this problem is solved on network sensors by implementing algorithms for matching traffic information and filtering rules.

The article proposes an approach to using the concept of software–configurable networks and virtualization of network functions to implement a traffic management system in promising mobile networks. It is proposed to use switches based on the OpenFlow protocol as network sensors. The article also proposes a method for processing and filtering traffic in transport networks, which is implemented in the form of a set of libraries and drivers for interacting network elements directly with applications, which reduces the search time for the necessary rules on telecommunications equipment.

KEYWORDS: software–defined networking, Virtualization of network functions, traffic management and filtering system, balancing, modified filtering algorithm.