Оригинальная статья / Original article

https://doi.org/10.21869/2223-1560-2022-26-2-142-158

Метод ранней диагностики потерь трафика реального времени с контролем таймаутов в программно-конфигурируемых сетях

К. И. Никишин [1] ✉

[1] Пензенский государственный университет
ул. Красная, д. 40, г.Пенза 440026, Российская Федерация

✉ e-mail: nkipnz@mail.ru

Резюме

Цель исследования: *разработка метода ранней диагностики потерь трафика реального времени с контролем таймаутов в программно-конфигурируемых сетях (ПКС).*

Методы. *Приведен обзор и описание передачи разнородного трафика в коммутаторе Ethernet с поддержкой качества обслуживания и технологии Time-Triggered Ethernet. Детально описано функционирование блоков контроля таймаутов на мгновенное удаление кадра из ПКС hard timeout и в зависимости от длины кадра при достижении приемной стороны idle timeout. Данные блоки управления включены в состав аппаратного защитника контроля таймаутов для трафика реального времени. Описана работа коммутатора OpenFlow с учетом защитника контроля таймаутов. Описано и разработано формирование временных окон таймаутов при передаче кадра.*

Результаты. *С помощью предложенного метода ранней диагностики потерь трафика реального времени с контролем таймаутов в ПКС было проведено моделирование ПКС на основе сетей Петри и пакета моделирования CPN Tools. Полученная модель и результаты экспериментов согласуются с теоретическими расчетами и предложенным методом передачи трафика реального времени с учетом его потерь в ПКС.*

Заключение. *С помощью модели было исследовано функциональное и временное поведение модели, проведена верификация метода с использованием сетей Петри. Были получены результаты в виде временных диаграмм, отражающих работу по типам трафика в соответствии с предложенным методом.*

Ключевые слова*: программно-конфигурируемые сети; коммутатор; Ethernet; OpenFlow; трафик реального времени; контроль потерь; таймаут; сети Петри; CPN Tools.*

Конфликт интересов: *Автор декларирует отсутствие явных и потенциальных конфликтов интересов, связанных с публикацией настоящей статьи.*

Для цитирования*:* Никишин К. И. Метод ранней диагностики потерь трафика реального времени с контролем таймаутов в программно-конфигурируемых сетях // Известия Юго-Западного государственного университета. 2022; 26(2): 142-158. https://doi.org/10.21869/2223-1560-2022-26-2-142-158.

Поступила в редакцию 18.06.2022 *Подписана в печать 04.07.2022* *Опубликована 29.07.2022*

Method for Early Diagnosis of Real-Time Traffic Loss with Timeout Monitoring in Software-Configurable Networks

Kirill I. Nikishin [1] ✉

[1] Penza State University
40 Krasnaya str., Penza 440026, Russian Federation

✉ e-mail: nkipnz@mail.ru

Abstract

The purpose of the research *is to develop the method for early diagnosis of real-time traffic loss with time-out control in software defined networks (SDN).*

Methods. *An overview and description of the transfer of heterogeneous traffic in an Ethernet switch with support for quality of service and Time-Triggered Ethernet technology is given. The functioning of the timeout control blocks for instantaneous removal of a frame from the hard timeout control system and depending on the length of the frame when the idle timeout receiving side is reached is described in detail. These control units are included in the hardware guard of control time-outs for real-time traffic. The work of the switch OpenFlow is described using guard of control time-outs. The formation of time windows of timeouts during frame transmission is described and developed.*

Results. *With the help of the proposed method of early diagnosis of real-time traffic losses with control of timeouts in SDN, a modeling of SDN based on Petri nets and using the CPN Tools modeling package was carried out. The obtained model and experimental results are consistent with the theoretical calculations and the proposed method of transmitting real-time traffic, taking into account its losses in SDN.*

Conclusion. *Using the model, the functional and temporal behavior of the model was investigated; the method was verified using Petri nets. The results were obtained in the form of time diagrams reflecting the work on traffic types in accordance with the proposed method.*

Keywords*: software-configurable networks; switch; Ethernet; OpenFlow; Real-time traffic loss control; timeout; Petri nets; CPN Tools.*

Conflict of interest. *The author declare the absence of obvious and potential conflicts of interest related to the publication of this article.*

For citation: Nikishin K. I. Method for Early Diagnosis of Real-Time Traffic Loss with Timeout Monitoring in Software-Configurable Networks. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*. 2022; 26(2): 142-158 (In Russ.). https://doi.org/ 10.21869/2223-1560-2022-26-2-142-158.

Received 18.06.2022 *Accepted 04.07.2022* *Published 29.07.2022*

***

Введение

Современные сетевые компьютерные парадигмы основаны на технологии Ethernet, которая позволяет передавать различный трафик по сети. Со временем усложнения алгоритмов и обработки трафика было введено «качество обслуживания» в Ethernet [1-3].

«Качество обслуживания» базируется на стандарте IEEE 802.1 [4], который маркирует и классифицирует трафик по приоритетам, чем выше приоритет, тем быстрее обрабатывается дан-

ный вид трафика. Максимальным приоритетом обладает трафик реального времени, для которого критична задержка кадров в сети и джиттер (отклонение от средней задержки кадра) [2]. Таким образом, в различных сетевых технологиях задержка для трафика реального времени может достигать одной микросекунды.

Современные требования, предъявляемые к обработке трафика, должны обладать свойствами мобильности, быстроты, упрощения администрирования сетевого оборудования. Все эти требования послужили к появлению ПКС [5-6].

В ПКС сетях выделяют 3 уровня: уровни приложений, управления и инфраструктурный. Основным элементом является контроллер, ведущий обработку трафика и рассчитывающий оптимальные маршруты для передачи трафика в коммутаторы. Обмен трафика производится через протокол OpenFlow [7-9].

В ПКС управление и настройка происходит над потоками данных, совокупностью набора кадров [10-12]. Входящий кадр поступает в коммутатор, работающий по протоколу OpenFlow, где ищется соответствие набора полей кадра в одной из таблиц потоков. Таблица потоков состоит из следующих полей: полей стандарта IEEE 802.1, приоритет, счетчики, инструкции, таймауты.

С помощью инструкций осуществляется дальнейшая обработка кадра в сети: передача в выходной порт, очередь коммутатора, удаление кадра, передача кадра в следующую таблицу потоков. Таймауты бывают двух видов: idle timeout – удалить запись в таблице и пакет, если он не достигнет приемной стороны и hard timeout – удалить принудительно запись в таблице и кадр, когда будет достигнуто заданное время.

Физически таблица потоков представляется сложным устройством в ПКС, поскольку содержит большое количество определенных полей и ведет статистику входящих, переданных и ошибочных кадров. К одному из недостатков ПКС можно отнести недостаточное снижение задержки трафика реального времени в сети, как, к примеру, по Time-Triggered Ethernet (TTE), позволяющей передавать трафик жесткого реального времени с точностью до 1 микросекунды.

В статье уделяется внимание передаче трафика реального времени в ПКС, поскольку важна передача с минимальной задержкой и минимальной потерей таких кадров в сети.

Материалы и методы

Согласно технологии TTE контроль трафика жесткого реального времени осуществляет аппаратный защитник. Физически защитник кадров находится на входе коммутатора Ethernet. Защитников может быть несколько в сети на случай возникновения ошибок в сети или сбоев в работе одного из защитников. На рис. 1 изображена сеть по TTE.

Защитник формирует на определенное время временное окно (тайм слот) для передачи трафика реального

времени. Учитывается непосредственно время передачи такого кадра до момента прихода к защитнику и ограничение максимальной задержки, которая допустима для такого кадра. В случае если кадр реального времени не проходит в отведенное временное окно, такой кадр удаляется из сети и заново отправляется в коммутатор Ethernet.

В статьях [13-15] было проведено моделирование коммутатора по TTE с качеством обслуживания и предложен новый метод передачи на основе сетей Петри. Для исследования компьютерных сетей подходит пакет CPN Tools, который позволяет исследовать сети с функциональной, временной точек зрения, а также верифицировать разработанные модели.

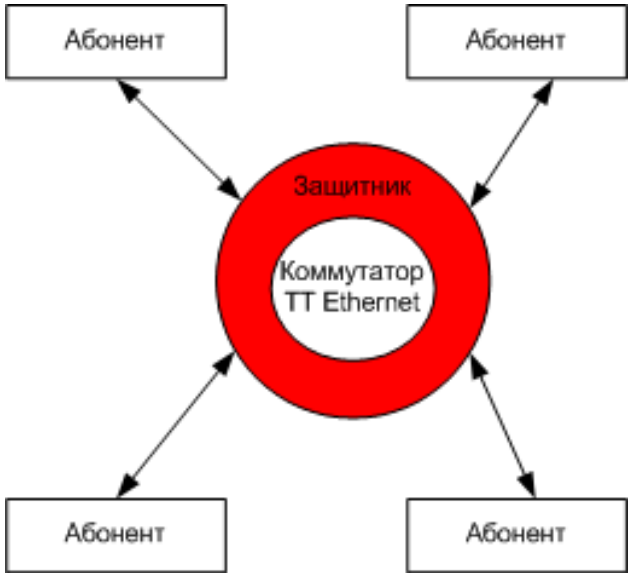

Рис. 1. Структура компьютерной сети по TTE

Fig. 1. Computer network structure by TTE

Возвращаясь к ПКС, контроллер и вся сеть реагирует на возможный случай потери трафика реального времени слишком поздно. Для передачи кадра от контроллера к коммутатору OpenFlow затрачивается некоторое время обработки, поскольку кадр передается не мгновенно. Кроме этого, кадр сравнивается в таблицах потоков на некоторое соответствие перечню полей, если кадр не находится в одной таблице потоков происходит переход к следующей и так далее. Если кадр не находится и в последней таблице потоков, только в этом случае происходит удаление кадра из ПКС и обновление таблиц потоков.

Как можно увидеть из описанного выше алгоритма протокол OpenFlow затрачивает значительное время для поиска кадра в таблицах потоков и сообщает контроллеру об удалении кадра слишком поздно, пока не пройдет поиск по всем таблицам потоков, в случае его несоответствия по полям. Кроме этого

таблица потоков слишком громоздкая, можно было бы достичь упрощения таблицы за счет вынесения контроля таймаутов на входе коммутатора OpenFlow.

Предлагается метод ранней диагностики потерь трафика реального времени с контролем таймаутов в ПКС. На входящий порт вводится специальный защитник по контролю таймаутов, не много схожий по технологии TTE. Специальный защитник контролируется только трафик реального времени, обычный трафик осуществляет передачи и управление также стандартным образом по протоколу OpenFlow.

Защитник состоит из нескольких блоков контроля таймаутов: контроль на мгновенное удаление кадра из ПКС hard timeout и контроль в зависимости от длины кадра достижения приемной стороны idle timeout. Аппаратный защитник контроля таймаутов представлен на рис. 2.

Таким образом, на вход коммутатора по протоколу OpenFlow добавляется аппаратный защитник таймаутов (рис. 3), защитников может быть несколько для избегания ошибок в ПКС.

Вначале производится анализ типа трафика – реального времени или обычный, если трафик реального времени поступает на вход защитника контроля таймаутов и проверку успешного прохождения кадра к коммутатору, в случае неуспешной передачи сообщение контроллеру ПКС на ранней стадии передачи кадра, чем стандартным методом по таблице потоков.

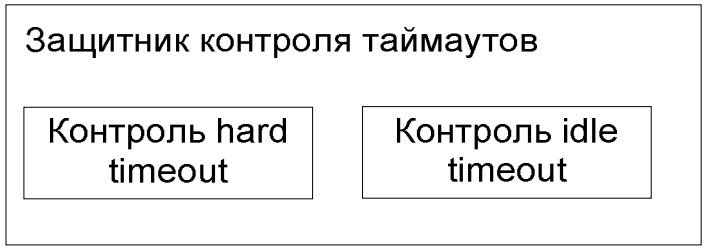

Рис. 2. Аппаратный защитник контроля передачи трафика реального времени с учетом таймаутов

Fig. 2. Hardware guard for controlling the transmission of real-time traffic and using timeouts

Обычный трафик поступает на нулевую таблицу потоков и далее по алгоритму протокола OpenFlow [16-18]. После проверки на успешную передачу кадр реального времени отправляется также в таблицы потоков для определения маршрута в ПКС.

Блок исполнить набор команд выполняет обработку кадра для передачи в выходной порт, направить в очередь кадр, удалить кадр и сообщить контроллеру. Если кадр реального времени не успевает передаться за время, заложенное в проверках таймаутов, то вырабатывается команда для информирования контроллера на повторную передачу кадра на ранней стадии. В остальных случаях в случае несоответствия кадра любого типа трафика во всех таблицах потоков, также такой кадр удаля-

ется из ПКС с информированием команды контроллера.

Моменты прибытия кадров реального времени контролируются в соответствии с временным окном hard timeout (рис.4).

Ttr + *Tlatency* < *Thtime,* где *Ttr* – время передачи от контроллера, *Tlatency* – задержка самой передачи на линиях, *Thtime* – время hard timeout.

Время начала Thtime_beg = *Tsys* + *Tlatency,* где *Tsys* – время в системе коммутатора.

Время окончания Thtime_end = *Thtime_beg* + *Tdelta,* где *Tdelta* – время настройки таймаута.

Формирование времени начала и окончания окна контроля таймаута представлено на рис. 5.

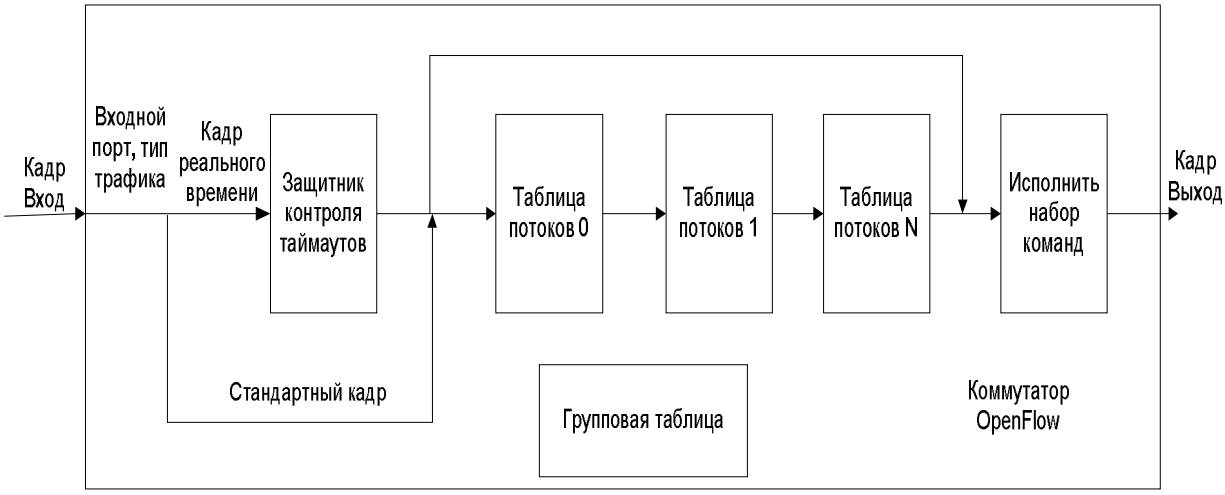

Рис. 3. Коммутатор OpenFlow с учетом защитника контроля таймаутов

Fig. 3. Switch OpenFlow switch using guard of control the timeout

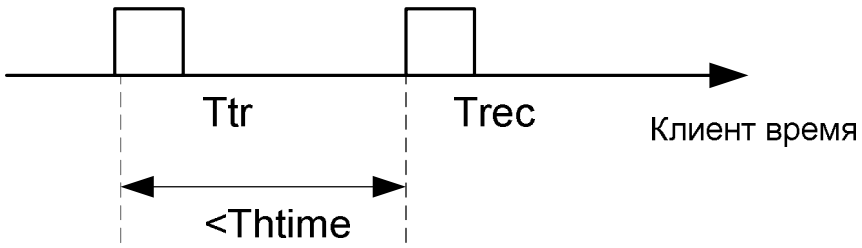

Рис. 4. Временное окно контроля таймаута hard timeout

Fig. 4. Temporal window of control timeout hard timeout

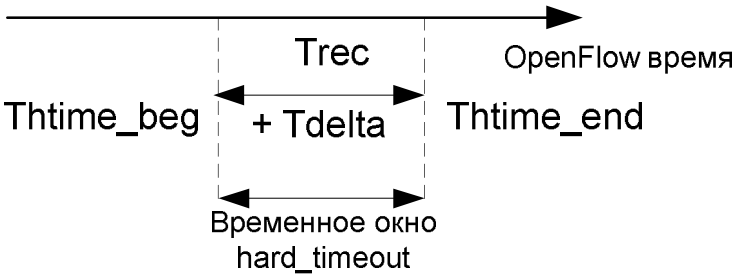

Рис. 5. Время начала и окончания окна контроля таймаута hard timeout

Fig. 5. The begin and end time of window of control the timeout hard timeout

После прохождения контроля таймаута hard timeout кадр реального времени отправляется на проверку контроля таймаута idle timeout. Моменты прибытия кадров реального времени контролируются в соответствии с новым временным окном idle timeout (рис.6).

Ttr + Tlatency < Titime, где *Titime* – время idle timeout.

Время начала Titime_beg = Tsys + Tlatency.

Время окончания Titime_end = Titime_beg + Tdelta + Tsize, где *Tsize* – время возможности передачи кадра реального времени в зависимости от его длины.

Формирование времени начала и окончания окна контроля таймаута представлено на рис. 7.

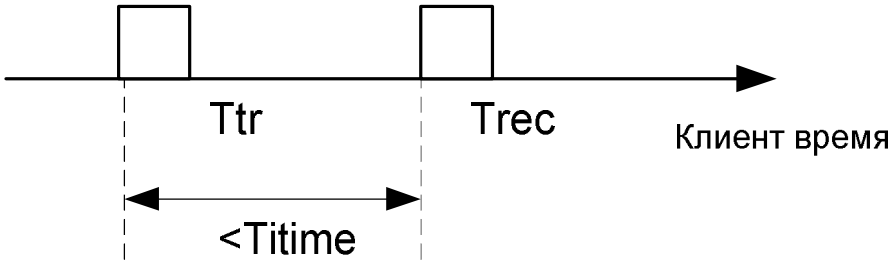

Рис. 6. Временное окно контроля таймаута idle timeout

Fig. 6. Temporal window of control timeout idle timeout

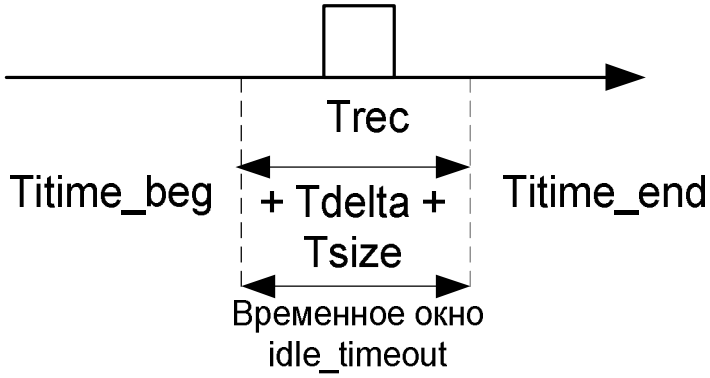

Рис. 7. Время начала и окончания окна контроля таймаута idle timeout

Fig. 7. The begin and end time of window of control the timeout idle timeout

Методы передачи трафика реального времени

В компьютерных сетях на основе Ethernet основным методом передачи разнородного трафика, в том числе трафика реального времени, является передача трафика на основе классификации приоритетов в буфере коммутатора. Приоритет трафика формируется в формате кадра согласно QoS и стандарту IEEE 802.1. Недостаток метода – невозможно определить на раннем этапе потерю трафика реального времени, кроме как на этапе передачи в выходной порт, может достигаться значительная задержка передачи и джиттер, зависящие от алгоритмов диспетчеризации в коммутаторе.

Компьютерные сети, использующие технологию TTE и данный метод передачи, обладают достаточно большим быстродействием передачи разнородного трафика, сеть содержит отказоустойчивые узлы, защиту от ошибок, контроль учета временных параметров трафика. Недостаток метода – использование только для таких сетей, где время принятия решения по передаче трафика реального времени и его маршрута достигается задержкой до 1 микросекунды, что не подходит для использования его в ПКС, поскольку транспортные затраты на принятие решения контроллером и администратором сети будут больше отведенного времени.

Метод передачи разнородного трафика в ПКС основывается на протоколе OpenFlow. К недостаткам передачи трафика в ПКС можно отнести следующее: таблицы потоков являются сложными устройствами, что требует в свою очередь больших аппаратных затрат на хранение и учет данных, таймаутов и статистики потоков; не достаточное снижение задержки для трафика реального времени, как в методе TTE; отсутствие контроля потери кадров (потоков) трафика реального времени на раннем этапе, а только при прохождении n количества по таблицам потоков; отсутствие аппаратного устройства на входе коммутатора, работающего по протоколу OpenFlow, который может сообщать контроллеру ПКС о необходимости повторной передачи трафика реального времени на раннем этапе.

Отличительные особенности предложенного метода ранней диагностики потерь трафика реального времени с контролем таймаутов в ПКС от описанных выше методов в компьютерных сетях:

– упрощаются таблицы потоков в OpenFlow для трафика реального времени за счет контроля и хранения таймаутов в своей структуре и повышение быстродействия анализа передачи кадра за счет снятия контроля таймаутов;

– осуществляется определение типа трафика на входе ПКС для дальнейшей передачи;

– для трафика реального времени осуществляется контроль на мгновенное удаление кадра из ПКС hard timeout за счет прибытия кадра в строго отведенное временное окно таймаута. В случае если момент прихода кадр не укладывается в отведенное окно hard timeout, кадр удаляется из ПКС и через

набор инструкций протокола OpenFlow сообщается контроллеру ПКС;

– после успешного контроля на hard timeout осуществляется дальнейший контроль в зависимости от длины кадра достижения приемной стороны idle timeout в строго отведенное временное окно таймаута. В случае если момент прихода кадр не укладывается в отведенное окно idle timeout, кадр удаляется из ПКС и через набор инструкций протокола OpenFlow сообщается контроллеру ПКС;

– предложен аппаратный защитник по совмещению функций контроля данных таймаутов и состоящий из блоков управления hard timeout и idle timeout, работающий по описанному выше алгоритму.

На рис. 8 представлен алгоритм метода ранней диагностики потерь трафика реального времени с контролем таймаутов в ПКС.

Результаты и их обсуждение

С помощью предложенного метода ранней диагностики потерь трафика реального времени с контролем таймаутов в ПКС было проведено моделирование компьютерной сети. Для этих целей был выбран математический аппарат сетей Петри, были выбраны цветные временные иерархические сети Петри, а в качестве инструмента построение таких сетей Петри пакет CPN Tools. Данный пакет хорошо зарекомендовал себя в части исследования телекоммуникационных и компьютерных сетей, их алгоритмов, протоколов, различного оборудования [1-3, 19-20].

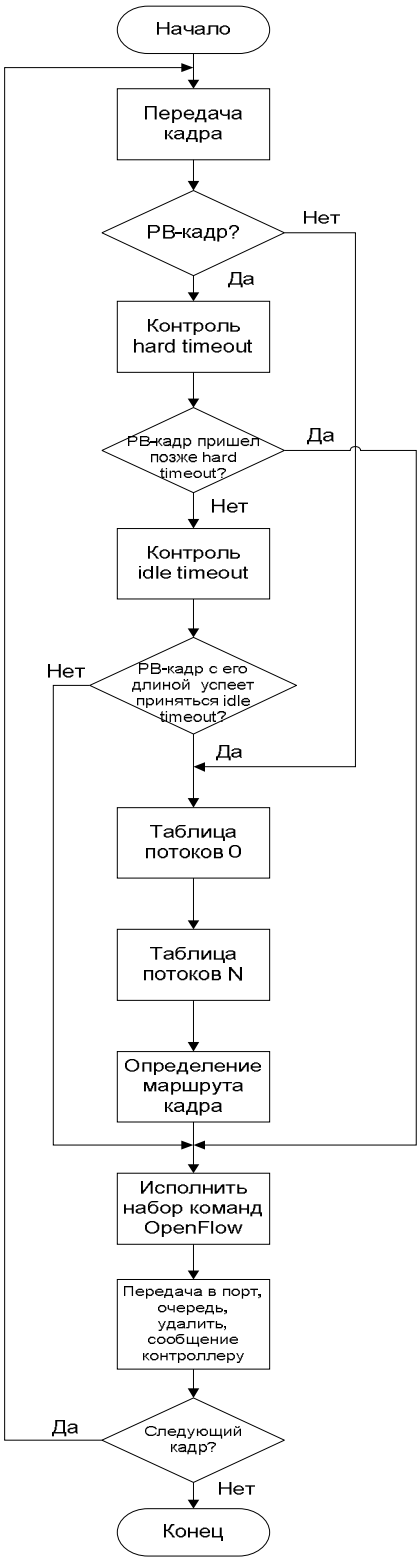

Рис. 8. Алгоритм предложенного метода с контролем таймаутов в ПКС

Fig. 8. The algorithm of the proposed method with the control of timeouts in SDN

На рис. 9 показана сеть Петри одного из главного компонента предложенного метода контроля hard timeout. Выполняется контроль времени прибытия трафика реального времени с временным окном контроля hard timeout. Подробное описание модели предложенного метода, алгоритмы функционирования на сетях будут рассмотрены в дальнейших статьях и исследованиях.

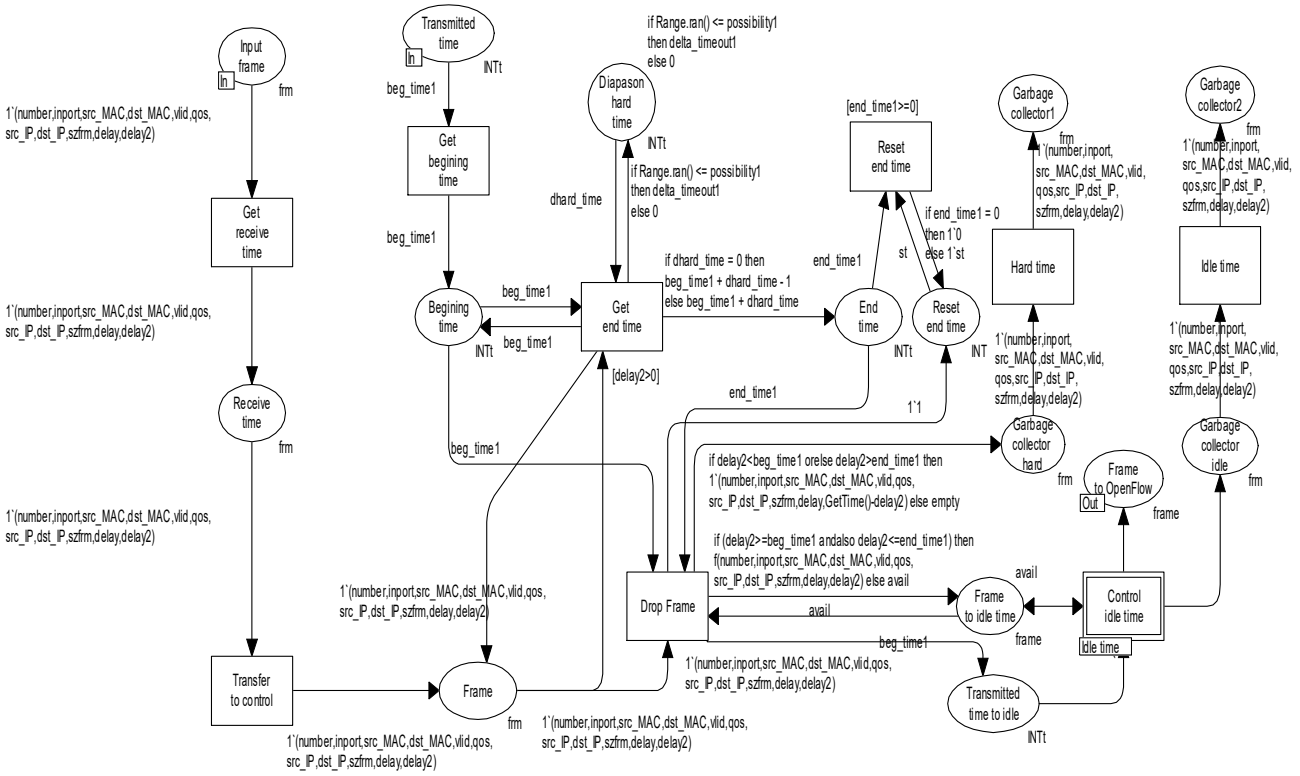

Рис. 9. Сеть Петри контроля hard timeout

Fig. 9. Petri nets of control hard timeout

Модель позволила исследовать предложенный метод, как на функциональное, так и на временное поведение модели, провести верификацию метода с использованием сетей Петри.

Профиль исследуемого трафика представлен на рис. 10, где показано распределение трафика реального времени (регулярный трафик) и стохастический трафик, который распределяется в диапазоне от 512 бит до 12 000 бит. Более подробное описание работы и исследование трафика для других методов и алгоритмов коммутаторов Ethernet описано в статье 19.

На рис. 11 представлено сравнение предложенного метода с классическим методом передачи потоков (кадров) в ПКС через коммутатор OpenFlow. Как видно из рис. 11 было проведено сравнение моделей на сетях Петри и показано, что задержка уменьшается.

Контроллер ПКС за счет ранней диагностики потери трафика реального

времени позволяет быстрее передать данные по сети, чем пока пройдется полное сравнение по всем таблицам потоков коммутатора OpenFlow, тем самым снижается задержка для трафика реального времени.

На рис. 12 представлены временные диаграммы, показывающие работу по типам трафика предложенного метода.

На рис. 12 передается разнородный трафик: РВ – трафик реального времени и ЭТ – обычный (эластичный) трафик. В случае неуспешного прохождения кадра через этап 1 на проверку hard timeout уже через время t1 сообщается контроллеру ПКС о повторной передаче, на проверке hard timeout сразу же через время t2. Стандартным методом передачи трафика в ПКС и контроль потерь РВ-трафика может происходить минимум только через время t3 при сравнении с правилами в таблице потоков 0, а максимальном варианте только через время N*t4, пока не произойдет сравнение со всеми таблицами потоков N и только после этого сообщение об удалении РВ-кадра контроллеру ПКС.

Полученная модель и результаты экспериментов согласуются с теоретическими расчетами и предложенным методом передачи трафика реального времени с учетом его потерь в ПКС.

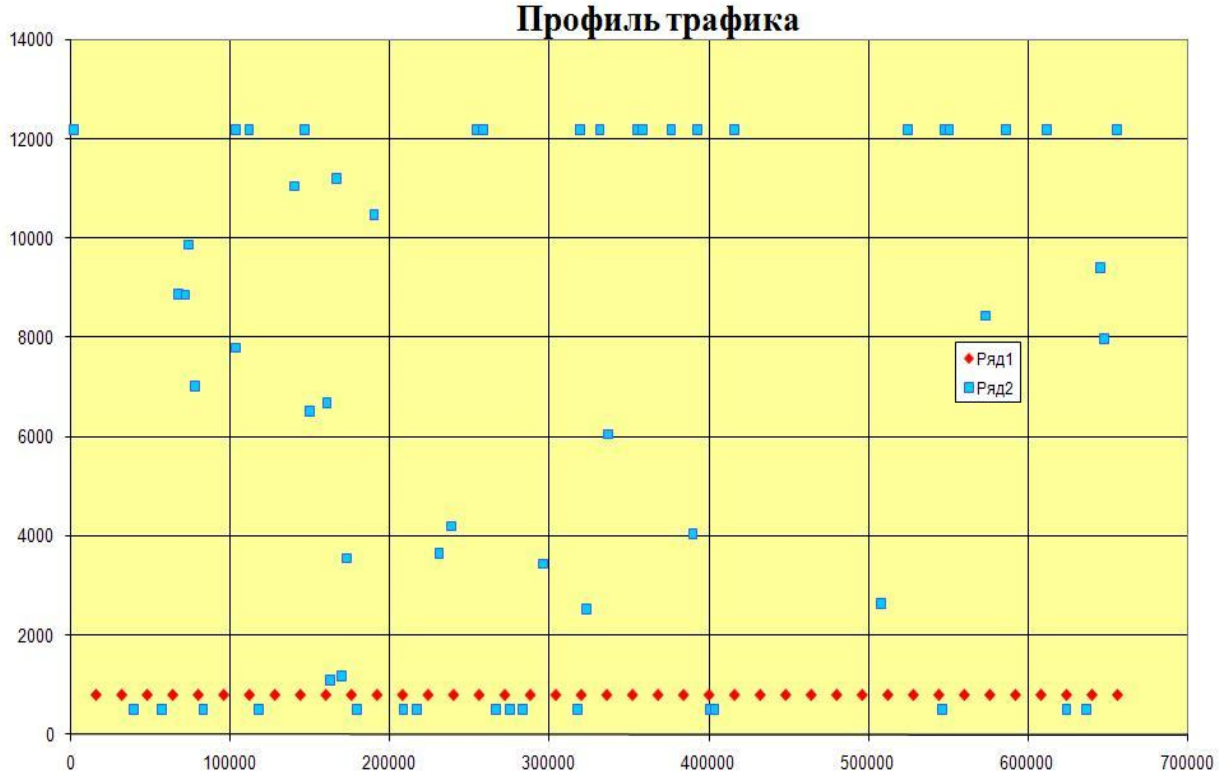

Рис. 10. Профиль трафика

Fig. 10. Profile of traffic

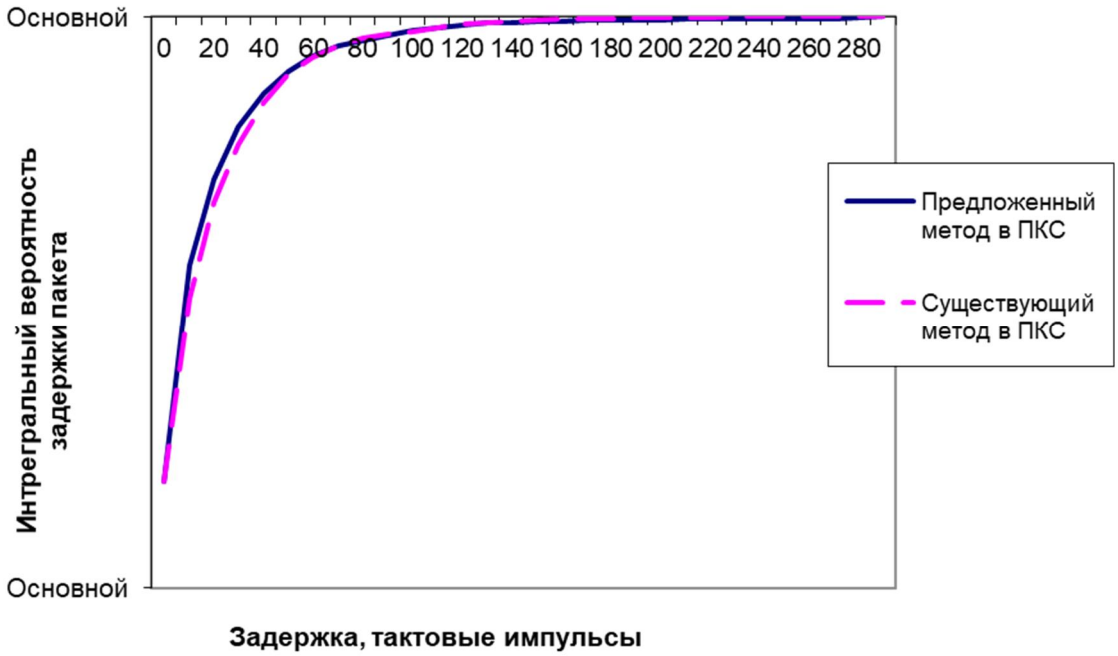

Рис. 11. Задержка передачи трафика двумя методами ПКС

Fig. 11. Delay of traffic transmission with two methods SDN

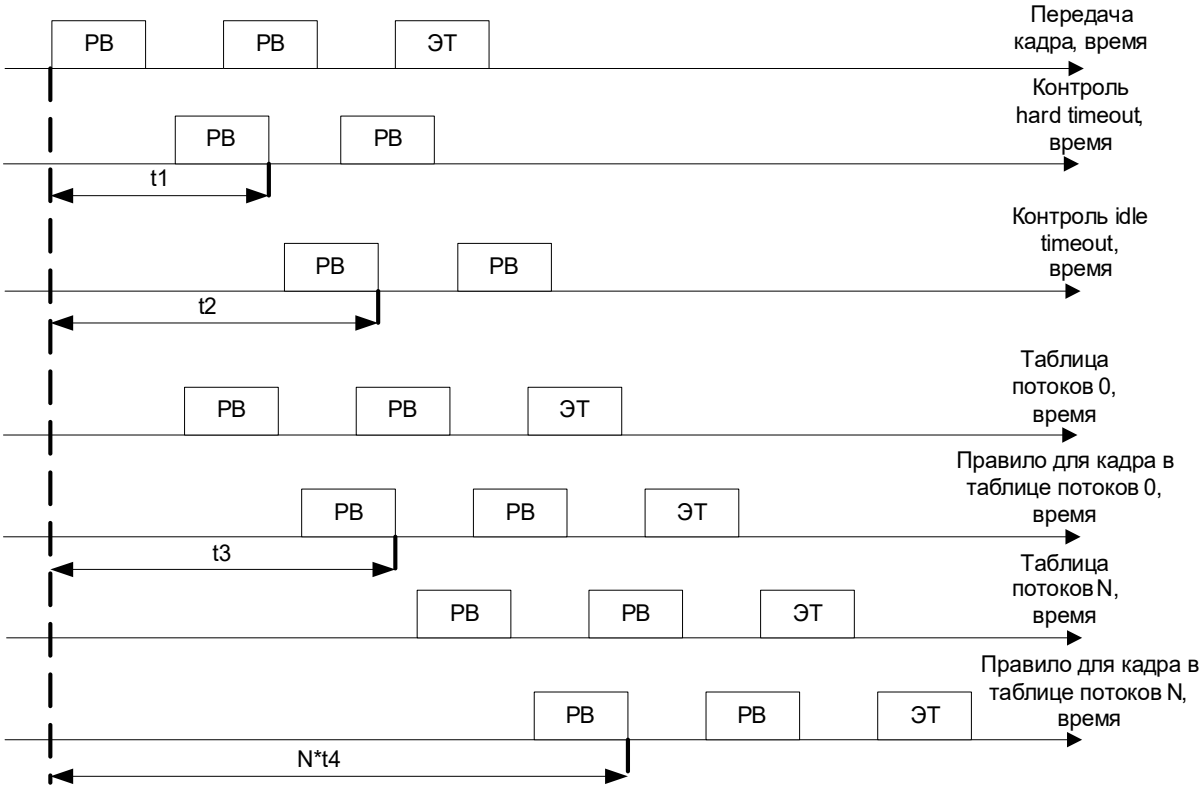

Рис. 12. Временные диаграммы работы модели предложенного метода с контролем таймаутов в ПКС

Fig. 12. Time diagrams of the work of model using the proposed method with the control of timeouts in SDN

Выводы

Предложен метод ранней диагностики потерь трафика реального времени с контролем таймаутов в ПКС и разработан алгоритм его работы. Детально описано функционирование блоков контроля таймаутов на мгновенное удаление кадра из ПКС hard timeout и в зависимости от длины кадра при достижении приемной стороны idle timeout.

Данные блоки управления включены в состав аппаратного защитника контроля таймаутов для трафика реального времени. Описана работа коммутатор OpenFlow с учетом защитника контроля таймаутов. Описано и разработано формирование временных окон таймаутов при передаче кадра.

С помощью предложенного метода ранней диагностики потерь трафика реального времени с контролем таймаутов в ПКС было проведено моделирование ПКС на основе сетей Петри.

Было исследовано функциональное и временное поведение модели, проведена верификация метода с использованием сетей Петри. Были получены результаты в виде временных диаграмм, отражающих работу по типам трафика в соответствии с предложенным методом.

Список литературы

1. Никишин К. И. Механизм управления трафиком реального времени в коммутаторе Ethernet // Вестник компьютерных и информационных технологий. 2015. № 10. С. 32–37.

2. Scheduling queues in the Ethernet switch, considering the waiting time of frames / E. Kizilov, N. Konnov, K. Nikishin, D. Pashchenko, D. Trokoz // MATEC Web of Conferences. 2016. Vol. 44. P. 01011-p.1–01011-p. 5.

3. Артемов И. В., Коннов М. Н., Никишин К. И. Анализ эффективности адаптивного алгоритма формирования виртуального таймслота в сетевом коммутаторе // Труды Международного симпозиума «Надежность и качество». Пенза: Изд-во ПГУ, 2020. Т.2. С. 298–302.

4. Описание стандарта IEEE 802.1q [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/IEEE_802.1Q (дата обращения 01.06.2022).

5. McKeown N., Anderson T., Balakrishnan H. et al. Openflow: enabling innovation in campus networks, ACM SIGCOMM Computer Communication Review, 2008, vol. 38, no. 2, pp. 69–74.

6. Kobayashi M., Seetharaman S., Parulkar G., Appenzeller G., Little J., Van Reijendam J., McKeown N. Maturing of OpenFlow and Software-Defined Networking Through Deployments, Computer Networks. 2014. Vol. 61. P. 151–175.

7. Корячко В. П., Перепелкин Д. А. Программно-конфигурируемые сети. М.: Горячая линия – Телеком, 2020. 288 с.

8. Shalimov A. et al. Advanced study of SDN/OpenFlow controllers // Proceedings of the 9th Central & Eastern European Software Engineering Conference in Russia. ACM, 2013.

9. Перепелкин Д. А. Концептуальный подход динамического формирования трафика программно-конфигурируемых телекоммуникационных сетей с балансировкой нагрузки // Информационные технологии. 2015. Т. 21. № 8. С. 602–610.

10. Перепелкин Д. А., Бышов В. С. Балансировка потоков данных в программно-конфигурируемых сетях с обеспечением качества обслуживания сетевых сервисов // Радиотехника. 2016. № 11. С. 111–119.

11. Никульчев Е. В., Паяин С. В., Плужник Е. В. Динамическое управление трафиком программно-конфигурируемых сетей в облачной инфраструктуре // Вестник Рязанского государственного радиотехнического университета. 2013. № 3 (45). С. 54–57.

12. Леохин Ю. Л., Фатхулин Т. Д. Оценка возможности предоставления гарантированной скорости передачи данных в программно-конфигурируемой оптической сети // Вестник Рязанского государственного радиотехнического университета. 2020. № 71. С. 45–59. https://doi.org/ 10.21667/1995-4565-2020-71-45-59.

13. Nikishin K., Konnov N. Schedule Time-Triggered Ethernet // International Conference on Engineering Management of Communication and Technology, EMCTECH 2020. DOI: 10.1109/EMCTECH49634.2020.9261540.

14. Никишин К. И., Коннов Н. Н., Пащенко Д. В. Моделирование систем на базе технологии Time-Triggered Ethernet // Информационные технологии и математическое моделирование: материалы XV Междунар. конф. имени А. Ф. Терпугова. Томск: Изд-во Томск. ун-та, 2016. Ч. 2. С. 117–122.

15. Nikishin K., Konnov N., Pashchenko D. Modelling of systems using Time-Triggered Ethernet // Springer Information Technologies and Mathematical Modelling – Queueing Theory and Applications. 2017. Vol. 638 of the series Communications in Computer and Information Science. P. 303–314.

16. Karakus M., Durresi A. Quality of service (QoS) in software defined networking (SDN): A survey // Journal of Network and Computer Applications. 2017. Vol. 80. P. 200-218.

17. Ren H., Li X., Geng J. A SDN‐based dynamic traffic scheduling algorithm, in IEEE International Conference on Cyber‐Enabled Distributed Computing and Knowledge Discovery (CyberC), Chengdu, China, 2016. https://doi.org/10.1109/CyberC.2016.103.

18. Maniu R., Dumitru L. A. Exploring the possibilities of a self‐regulating SDN controller // Scientific Bulletin «Mircea cel Batran» Naval Academy. 2015. Vol. 18. No. 1. P. 58-61.

19. Никишин К. И., Коннов Н. Н. Генератор трафика Ethernet на основе цветных сетей Петри // Модели, системы, сети в экономике, технике, природе и обществе. 2016. № 1 (17). С. 299–307.

20. Аладьев Ю. Ю., Никишин К. И. Моделирование сетевого трафика в пакете CPN Tools // Новые информационные технологии и системы: сб. науч. тр. XIV Междунар. науч.-техн. конф. Пенза: Изд-во ПГУ, 2017. С. 128–132.

References

1. Nikishin K. I. Mekhanizm upravleniya trafikom real'nogo vremeni v kommutatore Ethernet [The mechanism of management real-time traffic in the switch Ethernet]. *Vestnik komp'yuternykh i informatsionnykh tekhnologii = Herald of Computer and Information Technologies*, 2015, no. 10, pp. 32–37.

2. Kizilov E., Konnov N., Nikishin K., Pashchenko D., Trokoz D. Scheduling queues in the Ethernet switch, considering the waiting time of frames. *MATEC Web of Conferences*, 2016, vol. 44, pp. 01011-p.1–01011-p. 5.

3. Artemov I. V., Konnov M. N., Nikishin K. I. Analiz effektivnosti adaptivnogo algoritma formirovaniya virtual'nogo taimslota v setevom kommutatore [Analysis of the effectiveness of an adaptive algorithm for forming a virtual timeslot in a network switch]. *Trudy Mezhdunarodnogo simpoziuma «Nadezhnost' i kachestvo»* [Proceedings of the International Symposium "Reliability and quality"]. Penza, 2020, vol.2, pp. 298-302.

4. *Opisanie standarta IEEE 802.1q* [Description of the IEEE 802.1q standard]. Available at: https://ru.wikipedia.org/wiki/IEEE_802.1Q (accessed 01.06.2022).

5. McKeown N., Anderson T., Balakrishnan H. et al. Openflow: enabling innovation in campus networks, ACM SIGCOMM Computer Communication Review, 2008, vol. 38, no. 2, pp. 69–74.

6. Kobayashi M., Seetharaman S., Parulkar G., Appenzeller G., Little J., Van Reijendam J., McKeown N. Maturing of OpenFlow and Software-Defined Networking Through Deployments, Computer Networks, 2014, vol. 61, pp. 151–175.

7. Koryachko V. P., Perepelkin D. A. *Programmno-konfiguriruemye seti* [Software defined networks]. Moscow, Goryachaya liniya – Telekom Publ., 2020. 288 p.

8. Shalimov A. et al. Advanced study of SDN/OpenFlow controllers. Proceedings of the 9th Central & Eastern European Software Engineering Conference in Russia. ACM, 2013.

9. Perepelkin D. A. Kontseptual'nyi podkhod dinamicheskogo formirovaniya trafika programmno-konfiguriruemykh telekommunikatsionnykh setei s balansirovkoi nagruzki [Conceptual approach of dynamic traffic generation of software defined telecommunication networks with load balancing]. *Informatsionnye tekhnologii = Information Technologies,* 2015. vol. 21, no. 8, pp. 602-610.

10. Perepelkin D. A., Byshov V. S. Balansirovka potokov dannykh v programmno-konfiguriruemykh setyakh s obespecheniem kachestva obsluzhivaniya setevykh ser-visov [Balancing data flows in software defined networks with ensuring the quality of service of network services]. *Radiotekhnika = Radio Engineering*, 2016, no. 11, pp. 111-119.

11. Nikulchev E. V., Payin S. V., Pluzhnik E. V. Dinamicheskoe upravlenie trafikom programmno-konfiguriruemykh setei v oblachnoi infrastrukture [Dynamic traffic management of software defined networks in cloud infrastructure]. *Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta = Vestnik of Ryazan State Radio Engineering University*, 2013, no. 3 (45), pp. 54-57.

12. Leokhin Yu. L., Fathulin T. D. Otsenka vozmozhnosti predostavleniya garantirovannoi skorosti peredachi dannykh v programmno-konfiguriruemoi opticheskoi seti [Evaluation of the possibility of providing a guaranteed data transfer rate in a software defined optical network]. *Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta = Vestnik of Ryazan State Radio Engineering University*, 2020, no. 71, pp. 45-59. https://doi.org/10.21667/1995-4565-2020-71-45-59.

13. Nikishin K., Konnov N. Schedule Time-Triggered Ethernet. International Conference on Engineering Management of Communication and Technology, EMCTECH 2020. DOI: 10.1109/EMCTECH49634.2020.9261540.

14. Nikishin K. I., Konov N. N., Pashchenko D. V. Modelirovanie sistem na baze tekhnologii Time-Triggered Ethernet [Modelling of systems using Time-Triggered Ethernet]. *Informatsionnye tekhnologii i matematicheskoe modelirovanie: materialy XV Mezhdunar. konf. imeni A. F. Terpugova* [Information technologies and mathematical modeling: materials of the XV International Conf. named after A. F. Terpugov]. Tomsk, Tomsk University Publ., 2016, part 2. pp. 117-122. (In Russ.)

15. Nikishin K., Konnov N., Pashchenko D. Modelling of systems using Time-Triggered Ethernet. Springer Information Technologies and Mathematical Modelling – Queueing Theory and Applications, 2017, vol. 638 of the series Communications in Computer and Information Science, pp. 303–314.

16. Karakus M., Durresi A. Quality of service (QoS) in software defined networking (SDN): A survey. *Journal of Network and Computer Applications*, 2017, vol. 80, pp. 200-218.

17. Ren H., Li X., Geng J. A SDN‑based dynamic traffic scheduling algorithm, in IEEE International Conference on Cyber‑Enabled Distributed Computing and Knowledge Discovery (CyberC), Chengdu, China, 2016. https://doi.org/ 10.1109/CyberC.2016.103.

18. Maniu R. and Dumitru L. A. Exploring the possibilities of a self‑regulating SDN controller. *Scientific Bulletin «Mircea cel Batran» Naval Academy*, 2015, vol. 18, no. 1, pp. 58-61.

19. Nikishin K. I., Konnov N. N. Generator trafika Ethernet na osnove tsvetnykh setei Petri [The traffic generator of switch Ethernet using colored Petri nets]. *Modeli, sistemy, seti v ekonomike, tekhnike, prirode i obshchestve = Models, Systems, Networks in Economics, Technology, Nature and Society*, 2016, no. 1 (17), pp. 299–307.

20. Aladyev Yu. Yu., Nikishin K. I. Modelirovanie setevogo trafika v pakete CPN Tools [Modeling of network traffic in the CPN Tools package]. *Novye informatsionnye tekhnologii i sistemy: sb. nauch. tr. KhIV Mezh-dunar. nauch.-tekhn. konf.* [New information technologies and systems. Collection of scientific articles of XIV International scientific-technical. conf.]. Penza, PSU Publ., 2017, pp. 128-132.

Информация об авторе / Information about the Author

Никишин Кирилл Игоревич, кандидат технических наук, старший преподаватель кафедры вычислительной техники, Пензенский государственный университет, г. Пенза, Российская Федерация, e-mail: nkipnz@mail.ru, ORCID: https://orcid.org/0000-0001-7966-7833

Kirill I. Nikishin, Cand. of Sci. (Engineering),, Senior Lecturer of of Computer Engineering Department, Penza State University, Penza, Russian Federation, e-mail: nkipnz@mail.ru, ORCID: https://orcid.org/0000-0001-7966-7833