

ИССЛЕДОВАНИЕ ПЕРЕДАЧИ ТРАФИКА В ПРОГРАММНО-КОНФИГУРИРУЕМОЙ СЕТИ С ИСПОЛЬЗОВАНИЕМ CISCO PACKET TRACER

К.И. Никишин

Пензенский государственный университет, г. Пенза, Россия

Аннотация: распределенные компьютерные сети упрощают архитектуру сети и администрирование сети, повышая ее отказоустойчивость и возможность процесса резервного копирования. К такому типу сети относится программно-конфигурируемая сеть (ПКС). ПКС является перспективной и востребованной технологией передачи разнородного трафика, при котором для его направления в сети и взаимодействия с базовой аппаратной инфраструктурой используются контроллеры на базе ПО или API-интерфейсы. Цель - исследовать ПКС с помощью имитационного моделирования через специализированные сетевые программы для оценки передачи трафика в сети, пропускной способности и задержки передачи. Описана некоторая часть передачи трафика в ПКС с помощью аппарата сетей Петри пакета CPN Tools. Проведено имитационное моделирование в программе Cisco packet tracer. Была разработана и исследована сеть ПКС с одиночным контроллером в Cisco packet tracer. Описаны особенности конфигурирования сети, формирования IP-адресов, масок и шлюзов. Исследовано перемещение трафика в ПКС, оценены быстродействие пакетов в сети и доля потери пакетов, а также корректность работы оборудования в сети. В разработанной модели Cisco packet tracer представлены диаграммы состояния хостов и сетевых устройств (коммутаторы, маршрутизаторы), показано количество отправленных команд ping к хостам и процент дошедших до них пакетов, для сетевых устройств – процент состояния управления ими

Ключевые слова: программно-конфигурируемая сеть, Ethernet, OpenFlow, разнородный трафик, сети Петри, CPN Tools, Cisco Packet Tracer

Введение

Компьютерная сеть позволяет передавать данные по сети. Передача осуществляется коммутатором Ethernet, маршрутизаторами, роутерами [1]. Быстрая передача трафика возможна только при разграничении и классификации трафика [2]. Это стало возможно благодаря введению «качества обслуживания» (Quality of Service QoS) и стандарту IEEE 802.1 [3-4].

Дальнейшим развитием компьютерных сетей стало возникновение распределенных сетей. Распределенные сети упрощают архитектуру сети, упрощают администрирование сети, повышая отказоустойчивость сети и возможность процесса резервного копирования [5-7].

К распределенной сети можно отнести программно-конфигурируемую сеть (ПКС) [8-10]. ПКС – это новая парадигма в развитии компьютерных сетей, которая появилась сравнительно недавно в университетах Беркли и Стэнфорда, в 2006 году [11-13].

ПКС - это сетевая парадигма, при которой для направления трафика в сети и взаимодействия с базовой аппаратной

инфраструктурой используются контроллеры на базе ПО или API-интерфейсы.

Такая сеть отличается от традиционных сетей, в которых для управления сетевым трафиком используются специальные устройства (маршрутизаторы и коммутаторы Ethernet).

ПКС можно использовать для создания и администрирования виртуальной сети или для управления обычной аппаратной сетью с помощью ПО.

В то время как виртуализация сети обеспечивает возможность разбивать одну физическую сеть на разные виртуальные сети или объединять устройства из разных физических сетей в одну виртуальную сеть, ПКС предлагает новый способ управления маршрутизацией пакетов данных через централизованный сервер.

Выделяют следующие преимущества ПКС:

- более широкие возможности управления с повышенными скоростью и гибкостью;
- настраиваемая сетевая инфраструктура;
- высокий уровень безопасности

Модульная организация ПКС на основе математического аппарата сетей Петри

Открытый стандарт OpenFlow позволяет обрабатывать разный вид трафика и управлять им в ПКС [14-15]. Схематично обработку трафика в ПКС сети можно описать следующим образом: поступающий входной трафик перенаправляется на вход одного из портов коммутатора OpenFlow. Трафик поступает с уровня инфраструктуры на уровень управления.

Основным узлом протокола OpenFlow являются таблицы потоков. Эти таблицы представляют собой сочетание полей из заголовка поступившего кадра согласно стандарту IEEE 802.1, таймаутов, статистики, набор правил потоков.

Существуют различные виды таймаутов, осуществляющих жесткий и мягкий контроль времени поступления кадра. Правило для кадра, потока представляет собой выполнение определенных действий с ним. Выделяют следующие правила: направить непосредственно в определенный порт или в очередь, или удалить пакет, или обратно переслать контроллеру. Ведется статистика в таблицах потоков об успешной и неуспешной передаче, подсчет количества переданных кадров в определенном направлении (выходной порт, очереди, контроллер), количество ошибок в ПКС.

Таблица потоков принимает решение о дальнейшей передаче кадра, потока в ПКС. Кадр может быть удален из сети и формируется сигнал, оповещение контроллеру о повторной передаче.

Первоначально кадр поступает в таблицу потоков 0, если не найдено соответствие полей кадра по немаскированным полям в таблице (чаще всего MAC-адреса и IP-адреса), принимается решение о передаче в следующую таблицу потоков или об удалении кадра из сети.

В связи с этим возникла необходимость исследовать передачу разнородного трафика в ПКС. Для этих целей подходит имитационное моделирование. Одним из способов имитационного моделирования является формализация с помощью математического аппарата.

Выделяют различные математические аппараты: цифровые детерминированные и недетерминированные автоматы, сети Петри, семантические сети, использование предикатов,

GPSS, фреймы. Был выбран математический аппарат сетей Петри и пакет моделирования CPN Tools.

Главное преимущество аппарата сетей Петри [16] – построение сложных с модульной иерархией систем для исследования динамических параметров системы. Таким образом, для исследования различных традиционных и распределенных компьютерных сетей, облачной инфраструктуры, сетевых протоколов и алгоритмов подходит наилучшим образом данный математический аппарат.

Пакет CPN Tools основывается на цветных временных иерархических сетях Петри, позволяет проводить верификацию моделей, программировать на языке CPN ML [17]. Все эти возможности позволяют полноценно исследовать телекоммуникационную модель.

В данной статье не акцентируется полноценно внимание на исследовании передачи трафика с использованием аппарата сетей Петри, а только приводится часть ПКС. Более подробное исследование ПКС представлено в статьях [18-19].

Как было сказано ранее, основным элементом в ПКС являются таблицы потоков, поэтому ниже на рис. 1 показана часть сети Петри, моделирующая передачу кадра в таблицу потоков.

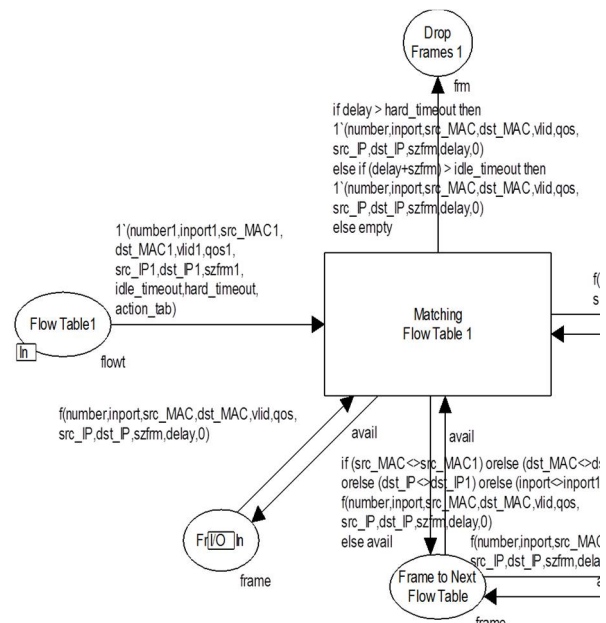


Рис. 1. Часть сети Петри, моделирующая передачу кадра в таблицу потоков

Через позицию Flow Table 1 задается структура, соответствующая физической таблице потоков. Через позицию Frame In

задается поступающий кадр на порт коммутатора. Часть сети Петри, моделирующая классификацию разнородного трафика, представлена на рис. 2.

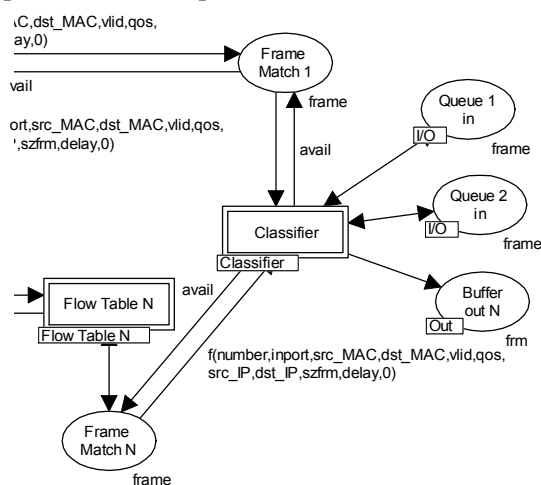


Рис. 2. Часть сети Петри, моделирующая классификацию трафика

В позиции Drop Frames 1 происходит удаление кадров в случае, если не выполняются действия с тайментами (жесткий таймаут `hard_timeout` и таймаут передачи кадра `idle_timeout`).

В таблице потоков находится узел классификации, осуществляющий как раз определенное действие с кадром, потоком. Классификатор в соответствии с правилом в таблице потоков определяет следующий маршрут кадра: отправить в очередь коммутатора в соответствии с его приоритетом или сразу же в выходной порт коммутатора.

Таким образом, с помощью аппарата сетей Петри была исследована передача трафика в ПКС.

Как было сказано ранее, моделирование с помощью какого-либо математического аппарата это одна сторона имитационного моделирования. Другой стороной является применение специализированных сетевых программ для эмуляции заданной сети [20]. Для этих целей была выбрана программа Cisco Packet Tracer для исследования уже непосредственно на физических и промышленных телекоммуникационных устройствах и оборудовании.

Разработка и исследование ПКС в Cisco Packet Tracer

Разработанная ПКС с одиночным контроллером представлена на рис. 3.

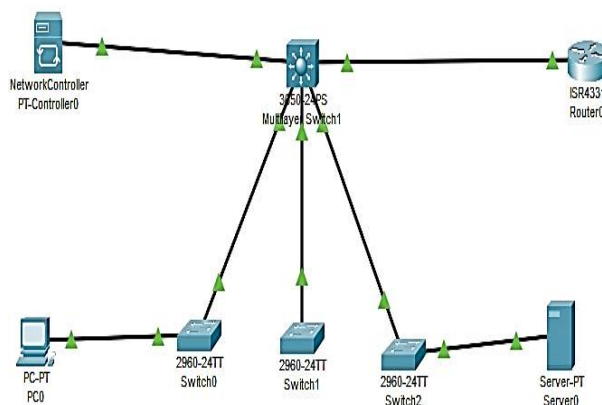


Рис. 3. ПКС с одиночным контроллером

Вначале производится настройка компонентов ПКС: персональные компьютеры PC, коммутаторы Switch 3 штуки, сервер Server, моноиерархический коммутатор Multilayer Switch и контроллер ПКС Network Controller. Для ПК прописываются параметры IP – 192.168.0.15, маска – 255.255.255.0 и шлюз – 192.168.0.254. С помощью ПК можно контролировать состояние сети через контроллер.

В файле конфигурации Switch прописываются интерфейсы `vlan1` на коммутаторе, команду `IP address` для назначения IP-адреса и маски подсети в режиме конфигурации интерфейса. Далее открываются порты командой `no shutdown` и проделывается то же самое для оставшихся двух коммутаторов.

В файле конфигурации для роутера прописывается IP адрес – 192.168.0.254 и маска – 255.255.255.0, аналогичным образом как на рис. 4. Для сервера задается шлюз – 192.168.0.254, IP – 192.168.0.253 и маска – 255.255.255.0. Для контроллера ПКС задается шлюз – 192.168.0.254, IP – 192.168.0.1 и маска – 255.255.255.0.

Для достоверности и работоспособности ПКС отправляется команда `ping` контроллеру и осуществляется контроль на присутствие или отсутствие потери пакетов и на задержку в ПКС (рис. 4).

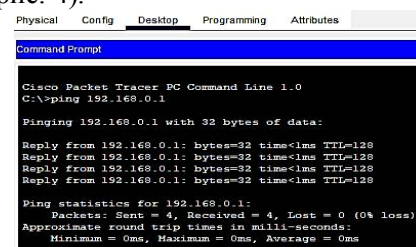


Рис. 4. Проверка отклика контроллера ПКС

Для определения валидации и работоспособности сети убеждаемся в движении трафика во всей ПКС. На рис. 5 показаны пакеты, дошедшие до своего места назначения через иконку зеленое письмо.

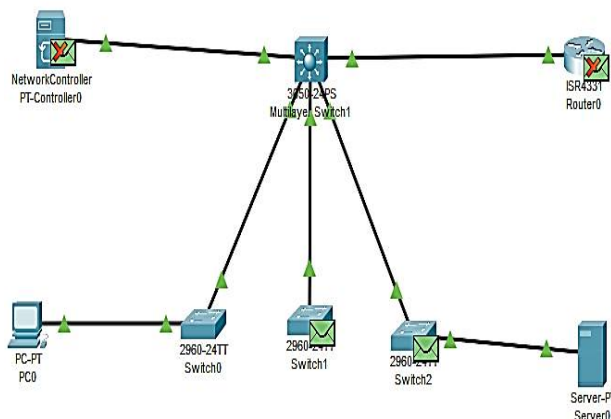


Рис. 5. Движение маршрутов трафика в ПКС

На рис. 6 представлены замеры времени перемещения пакетов в ПКС.

Vis.	Time(sec)	Last Device	At Device	Type
0.000	-	-	PC0	ICMP
0.000	-	-	PC0	ARP
0.001	0.001	PC0	Switch0	ARP
0.002	0.002	Switch0	Multilayer Switch1	ARP
0.003	0.003	Multilayer Switch1	Switch1	ARP
0.003	0.003	Multilayer Switch1	Switch2	ARP
0.003	0.003	Multilayer Switch1	Router0	ARP
0.003	0.003	Multilayer Switch1	PT-Controller0	ARP
0.004	0.004	Switch2	Server0	ARP

Рис. 6. Время перемещения пакетов в ПКС

Далее через рабочий стол PC заходим в браузер и прописываем команды, предварительно вписав логин и пароль, которые прописывались в файле конфигурации коммутаторов и маршрутизатора.

Вводится в браузере PC ссылка с нашим IP-адресом <https://192.168.0.1>. При необходимости можно ввести и другую ссылку.

Для отображения полной статистики нужно добавить несколько устройств. Для начала во вкладке “credentials” прописывается логин и пароль администратора ПКС, а также пользовательский пароль. Затем во вкладке “discovery” вводятся параметры для поиска устройств, а именно имя устройства, через которое ищутся остальные устройства (SW1),

IP-адрес этого устройства и шаблон данных для входа.

Присутствует вкладка сетевых устройств ПКС, на которой показывается название узла, его тип, IP-адрес, время его работы, последние обновления устройства и состояние управление им.

В результате работы ПКС программа Cisco Packet Tracer предоставляет диаграммы состояния хостов (PC и Server) и сетевых устройств (коммутаторы, маршрутизаторы) на вкладке сетевого контроллера. Можно увидеть отправку команд ping к хостам и процент дошедших до них пакетов, для сетевых устройств – процент состояния управления ими (рис. 7).

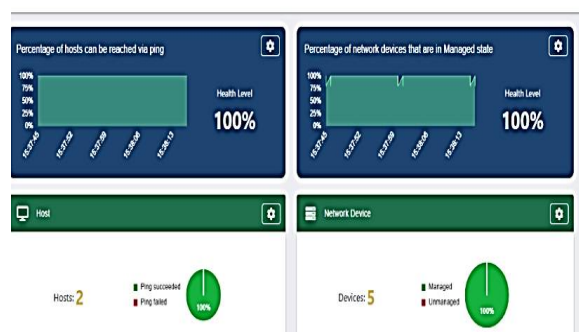


Рис. 7. Вкладка сетевого контроллера

Можно просмотреть более полное состояние устройств сети во вкладке “assurance”. В ней сетевые устройства разделены на более простые устройства для упрощения контроля сетевого состояния.

Заключение

ПКС является перспективной и востребованной технологией передачи разнородного трафика, при которой для его направления в сети и взаимодействия с базовой аппаратной инфраструктурой используются контроллеры на базе ПО или API-интерфейсы.

Все поставленные задачи и цель по исследованию ПКС были выполнены с помощью имитационного моделирования. Одним из способов имитационного моделирования является формализация с помощью математического аппарата.

Описана часть передачи трафика в ПКС с помощью аппарата сетей Петри пакета CPN Tools. Было проведено имитационное моделирование в одной из специализированных сетевых программ для

эмуляции заданной сети, для этих целей лучшим образом подошла программа Cisco Packet Tracer.

Была разработана, исследована сеть ПКС с одиночным контроллером в Cisco Packet Tracer. Описаны особенности конфигурирования сети, формирования IP-адресов, масок и шлюзов. Исследовано перемещение трафика в ПКС, оценены быстродействие пакетов в сети и доля потери пакетов, а также корректность работы оборудования в сети.

Литература

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. 4-е изд. СПб.: Питер, 2010. 943 с.
2. Никишин К.И. Механизм управления трафиком реального времени в коммутаторе Ethernet // Вестник компьютерных и информационных технологий. 2015. № 10. С. 32–37.
3. Описание стандарта IEEE 802.1q [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/IEEE_802.1Q.
4. 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.
5. Корячко В.П., Перепелкин Д.А. Программно-конфигурируемые сети: учебник для вузов. М.: Горячая линия – Телеком, 2020. 288 с.
6. Advanced study of SDN/OpenFlow controllers/ A. Shalimov et al. // Proceedings of the 9th Central & Eastern European Software Engineering Conference in Russia. ACM, 2013.
7. 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.
8. Перепелкин Д.А. Концептуальный подход динамического формирования трафика программно-конфигурируемых телекоммуникационных сетей с балансировкой нагрузки // Информационные технологии. 2015. Т. 21. № 8. С. 602–610.
9. 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.
10. Программная инфраструктура и визуальная среда распределенной обработки потоков данных в программно-конфигурируемых сетях / В.П. Корячко, Д.А. Перепелкин, М.А. Иванчикова, В.С. Бышов, И.Ю. Цыганов // Вестник Рязанского государственного радиотехнического университета. 2018. № 65. С. 44–54. DOI: 10.21667/1995-4565-2018-65-3-44-54.
11. Openflow: enabling innovation in campus networks/ N. McKeown, T. Anderson, H. Balakrishnan et al. // ACM SIGCOMM Computer Communication Review. 2008. Vol. 38. No. 2. P. 69–74.
12. Maturing of OpenFlow and Software-Defined Networking Through Deployments/ M. Kobayashi, S. Seetharaman, G. Parulkar, G. Appenzeller, J. Little, J. Van Reijendam, N. McKeown // Computer Networks. 2014. Vol. 61. P. 151–175.
13. Ушакова М.В., Ушаков Ю.А. Исследование сети виртуальной инфраструктуры центра обработки данных с гибридной программно-конфигурируемой коммутацией // Вестник Рязанского государственного радиотехнического университета. 2021. № 75. С. 34–43. DOI: 10.21667/1995-4565-2021-75-34-43.
14. Никульчев Е.В., Паян 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.
15. 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. DOI: 10.21667/1995-4565-2020-71-45-59.
16. Wael Hosny Fouad Aly. A novel controller placement using Petri-nets for SDNs// Wseas Transactions on Mathematics. 2020. Vol. 19. P. 598–605.
17. 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.
18. Никишин К.И. Моделирование и верификация топологий программно-конфигурируемых сетей // Вестник Рязанского государственного радиотехнического университета. 2022. № 80. С. 67–74.
19. Никишин К.И. Моделирование контроллера и верификация процесса передачи данных в программно-конфигурируемых сетях // Вестник Рязанского государственного радиотехнического университета. 2022. № 80. С. 75–83.
20. Никишин К.И. Моделирование беспроводной сенсорной сети с использованием OMNET++ // Вестник Рязанского государственного радиотехнического университета. 2021. № 78. С. 46–54.

Поступила 27.06.2022; принята к публикации 10.10.2022

Информация об авторах

Никишин Кирилл Игоревич – канд. техн. наук, старший преподаватель кафедры вычислительной техники, Пензенский государственный университет (440026, Россия, г. Пенза, ул. Красная, д. 40), e-mail: nkipnz@mail.ru

RESEARCH OF TRAFFIC TRANSMISSION IN A SOFTWARE DEFINED NETWORK USING CISCO PACKET TRACER

K.I. Nikishin

Penza State University, Penza, Russia

Abstract: distributed computer networks simplify the network architecture and network administration, increase network fault tolerance and the possibility of a backup process. This type of network includes a software defined network (SDN). SDN is a promising and in-demand technology for the transmission of heterogeneous traffic, in which software-based controllers or

APIs are used to direct it to the network and interact with the underlying hardware infrastructure. The goal is research SDN using simulation modeling through specialized network programs to assess the transmission of traffic in the network, bandwidth and latency. Some part of the traffic transfer to the SDN using the device of the CPN Tools package Petri nets is described. Simulation modeling was carried out in the Cisco packet tracer program. A network of SDN with a single controller in Cisco packet tracer was developed and investigated. The features of network configuration, formation of IP addresses, masks and gateways are described. The movement of traffic to the PC was investigated, the speed of packets in the network and the proportion of packet loss, as well as the correctness of the operation of equipment in the network were evaluated. The developed Cisco packet tracer model presents diagrams of the status of hosts and network devices (switches, routers), shows the number of ping commands sent to hosts and the percentage of packets that reached them, for network devices – the percentage of their management status

Key words: software defined network, Ethernet, OpenFlow, heterogeneous traffic, Petri nets, CPN Tools, Cisco Packet Tracer

References

1. Olifer V.G., Olifer N.A. "Computer networks. Principles, technologies, protocols." ("Komp'yuternye seti. Printsipy, tekhnologii, protokoly"), St. Petersburg: Piter, 2010, 943 p.
2. Nikishin K.I. "Real-time traffic control mechanism in an Ethernet switch", *Bulletin of Computer and Information Technologies (Vestnik komp'yuternykh i informatsionnykh tekhnologiy)*, 2015, no. 10, pp. 32–37.
3. Description of the IEEE 802.1q standard, available at: https://ru.wikipedia.org/wiki/IEEE_802.1Q.
4. 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. Koryachko V.P., Perepelkin D.A. "Software defined networks" ("Programmno-konfiguriruyemye seti"), Moscow: Goryachaya liniya – Telekom, 2020, 288 p.
6. Shalimov A. et al. "Advanced study of SDN/OpenFlow controllers", *Proc. of the 9th Central & Eastern European Software Engineering Conference in Russia*. ACM, 2013.
7. Nikishin K., Konnov N. "Schedule time-triggered ethernet", *Int. Conf. on Engineering Management of Communication and Technology*, EMCTECH 2020. DOI: 10.1109/EMCTECH49634.2020.9261540.
8. Perepelkin D.A. "Conceptual approach of dynamic traffic shaping of software-defined telecommunication networks with load balancing", *Information technologies (Informatsionnye tekhnologii)*, 2015, vol. 21, no. 8, pp. 602–610.
9. Perepelkin D.A., Byshov V.S. "Balancing data flows in software-defined networks with ensuring the quality of service for network services", *Radio Engineering (Radiotekhnika)*, 2016, no. 11, pp. 111–119.
10. Koryachko V.P., Perepelkin D.A., Ivanchikova M.A., Byshov V.S., Tsyganov I.Yu. "Software infrastructure and visual environment for distributed processing of data flows in software-defined networks", *Bulletin of Ryazan State Radio Engineering University (Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta)*, 2018, no. 65, pp. 44–54. DOI: 10.21667/1995-4565-2018-65-3-44-54.
11. 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.
12. 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.
13. Ushakova M.V., Ushakov Yu.A. "Study of the virtual infrastructure network of a data center with hybrid software-defined switching", *Bulletin of Ryazan State Radio Engineering University (Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta)*, 2021, no. 75, pp. 34–43. DOI: 10.21667/1995-4565-2021-75-34-43.
14. Nikul'chev E.V., Payain S.V., Pluzhnik E.V. "Dynamic traffic management of software-defined networks in cloud infrastructure", *Bulletin of Ryazan State Radio Engineering University (Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta)*, 2013, no. 3 (45), pp. 54–57.
15. Leokhin Yu.L., Fathulin T.D. "Evaluation of the possibility of providing a guaranteed data transfer rate in a software-defined optical network", *Bulletin of Ryazan State Radio Engineering University (Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta)*, 2020, no. 71, pp. 45–59. DOI: 10.21667/1995-4565-2020-71-45-59.
16. Wael Hosny Fouad Aly "A novel controller placement using Petri-nets for SDNs", *Wseas Transactions on Mathematics*, 2020, vol. 19, pp. 598–605.
17. Nikishin K.I., Konnov N.N. "Ethernet traffic generator based on colored Petri nets", *Models, systems, networks in economics, technology, nature and society (Modeli, sistemy, seti v ekonomike, tekhnike, prirode i obshchestve)*, 2016, no. 1 (17), pp. 299–307.
18. Nikishin K.I. "Modeling and verification of topologies of software-defined networks", *Bulletin of Ryazan State Radio Engineering University (Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta)*, 2022, no. 80, pp. 67–74.
19. Nikishin K.I. "Controller modeling and verification of the data transfer process in software-defined networks", *Bulletin of Ryazan State Radio Engineering University (Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta)*, 2022, no. 80, pp. 75–83.
20. Nikishin K.I. "Modeling a wireless sensor network using OMNET++", *Bulletin of Ryazan State Radio Engineering University (Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta)*, 2021, no. 78, pp. 46–54.

Submitted 27.06.2022; revised 10.10.2022

Information about the authors

Kirill I. Nikishin, Cand. Sci. (Technical), Assistant Professor, Penza State University (40 Krasnaya str., Penza 440026, Russia), e-mail: nkipnz@mail.ru