

Мухи Алдин Хассан Мохамед

Одесский национальный политехнический университет

Ткачева Е.Б.

Харьковский национальный университет радиоэлектроники

КОМПЛЕКСНЫЙ МЕТОД ДИНАМИЧЕСКОЙ РЕПЛИКАЦИИ ДАННЫХ В ПРОГРАММНО-КОНФИГУРИРУЕМЫХ СЕТЯХ

В статье предложена модель формирования, предоставления комплексного сервиса и метод динамической репликации, позволяющие повысить доступность сервисов в программно-конфигурируемых сетях. Проведен обзор общеизвестных методов репликации, которые нашли широкое применение в распределенных сетях, приведены их основные преимущества и недостатки. На основании результатов проведенного анализа и с учетом особенностей формирования сервисов предложен метод, который включает в себя решение таких задач: оценки популярности сервиса и выбора момента запуска операции репликации на основании значения порога репликации; вычисления оптимального количества реплик и их размещения на вычислительных узлах; проверки непротиворечивости новой реплики сервиса уже существующим в сети.

Ключевые слова: сети SDN, уровень QoS level, протокол OpenFlow, динамическая репликация, моновалентный сервис, ToS.

Постановка проблемы. Возрастающие бизнес-потребности как отдельных пользователей и организаций обусловили рост популярности и стремительного развития новых сетевых решений: программно-конфигурируемых сетей, сетей с поддержкой функций виртуализации, cloud систем обработки и хранения данных. Анализ рынка инфокоммуникаций [1–4] показал, что провайдеры и крупные IT-организации для обеспечения требуемого качества обслуживания (Quality of Service, QoS) и снижения стоимости предоставляемых услуг все чаще прибегают к использованию концепции программно-конфигурированных сетей (Software-Defined Networking, SDN) [1–3].

В основе концепции SDN лежит идея отделения уровня управления сетью (control plane) от уровня передачи данных (forwarding plane) [2; 3]. Логика принятия решений и функции управления передачей данных переносятся на контроллер – сетевой элемент, обеспечивающий централизованное управление и мониторинг [4; 5]. К основным преимуществам SDN относятся адаптивность, высокая производительность и адаптивность. В соответствии со статистическими данными [6; 7] применение концепции SDN позволяет значительно снизить материальные затраты на такие виды услуг, как модернизация сервисов, управление, администрирование и техническая поддержка.

Однако при переходе от традиционных принципов построения к концепции SDN ряд проблем

остаётся нерешённым: отказы систем хранения данных, избыточность сетевых ресурсов, невысокая отказоустойчивость периферийного оборудования являются основными причинами, вызывающими деградацию QoS в SDN [8]. Наиболее эффективными и распространёнными решениями быстрого устранения подобных сбоев и восстановления требуемого уровня QoS на сегодняшний день являются механизмы кластерной [9] и/или глобальной репликации [10] данных. Применение репликации позволяет повысить производительность и надёжность сети в целом [9–11].

I. Обзор методов репликации в программно-конфигурируемых сетях

На сегодняшний день множество работ посвящено разработке и усовершенствованию механизмов репликации данных в распределённой сетевой инфраструктуре [9–11]. В современных источниках механизмы репликации классифицируются так: статическая и динамическая, централизованная и децентрализованная, синхронная и асинхронная репликация, каждая из которых находит своё применение в зависимости от топологии и масштабов сети, а также территориального размещения вычислительных элементов [11–14]. Анализ существующих решений показал, что широкое практическое применение приобрела методика динамической асинхронной репликации данных. Основным преимуществом методики является сравнительно низкие служебных и поль-

зовательских данных, передаваемых в процессе репликации.

Методам динамической репликации посвящены работы [15–19]. Совершенствование стратегии динамической репликации приведено в работе японских ученых T. Ikeda, M. Ohara и S. Fukumoto [15]. Применение асинхронного режима репликации с возможностью сохранения нескольких экземпляров копий сервиса на вычислительном узле является основной особенностью предложенного авторами метода [15]. Выбор определенного экземпляра реплики вычислительным элементом осуществляется на основе ее популярности. Управление и поддержка нескольких экземпляров сервиса в данном случае является ресурсоемкой задачей – количество сервисов, а, следовательно, и количество реплик неуклонно растет, что приводит к иррациональному использованию сетевых ресурсов.

В работе [16] основное внимание уделено вопросам формирования оптимального количества копий и их распределения между вычислительными устройствами. Однако алгоритм репликации, предложенный в этой работе, не позволяет учитывать вероятность и причины отказа в обслуживании. Так, обращения к вычислительному элементу, от которого получен отказ в обслуживании, могут повторяться снова и снова, приводя к потерям пакетов и деградации качества обслуживания.

В работе [17] предложена методика хранения и формирования реплик, которая позволяет обеспечивать динамическое распределение сетевых ресурсов между несколькими пользователями или приложениями путем анализа популярности и стоимости использования услуг. Распределение сетевых ресурсов между приложениями выполняется динамически и зависит от типа сервиса и требований QoS. Эффективность данного подхода в процессе предоставления высококачественного контента ограничена из-за длительности времени принятия решения о необходимости репликации и низкой точности распределения сетевых ресурсов.

В работе [18] рассмотрены механизмы асинхронной репликации, а также предложена комплексная методика, которая позволяет уменьшить объемы передаваемых данных и нивелировать противоречия между репликами. Однако возникновение большого количества запросов, обращенных к одному и тому же вычислительному устройству одновременно может привести к его перегрузке и, как следствие, отказу в обслуживании. Кроме этого, оригинал услуги или ее основ-

ная копия всегда должны быть доступны, что накладывает ряд ограничений на использование предложенного метода.

Two-level Distributed Hash Table метод предложен в [19]. В предлагаемом подходе определяется необходимость формирования реплики на основе информации о ее популярности. Решение о размещении реплики принимается на основе двух критериев: основной локации запросов пользователей на предоставление сервиса и физических характеристик вычислительных узлов.

Приведенные выше подходы нашли широкое применение в процессе управления распределенной сетевой инфраструктурой. Однако их непосредственное применение в SDN затруднительно в силу ряда причин, связанных с высокими затратами сетевых ресурсов в процессе распределения и хранения реплик [15], отсутствием методов переадресации запросов на вычислительный узел, содержащий реплику [17], отсутствием учета производительности вычислительных элементов, на которых размещена реплика, [16] и механизмов балансировки сетевой нагрузки в процессе распределения и использования реплик услуги [18; 19].

Таким образом, в условиях применения репликации сервисов как метода повышения их доступности и обеспечения требуемого качества обслуживания в SDN известные решения должны быть пересмотрены в сторону повышения их эффективности, что обуславливает актуальность задачи управления количеством и расположением реплик, в частности подзадач определения момента принятия решения о формировании реплик сервиса, определения необходимого минимального количества реплик и выбора их месторасположения, а также установления момента принятия решения о непротиворечивости новой реплики сервиса в контексте их применения в SDN.

II. Модель процесса предоставления комплексных сервисов

Как было отмечено, контроллер является ключевым элементом системы управления в сетях, построенных на основе концепции SDN. Производительность сети и качество предоставляемых оконечному пользователю сервисов всецело зависят от функциональных особенностей контроллера и принципов его взаимодействия с рядом других вспомогательных модулей управления, таких как планировщик задач, балансировщик нагрузки, модуль оркестрации сервисов, брокер репликации и прочих [3], которые поставляют контроллеру актуальную информацию о состоянии подчиненных ему фрагментов сети (рис.1).

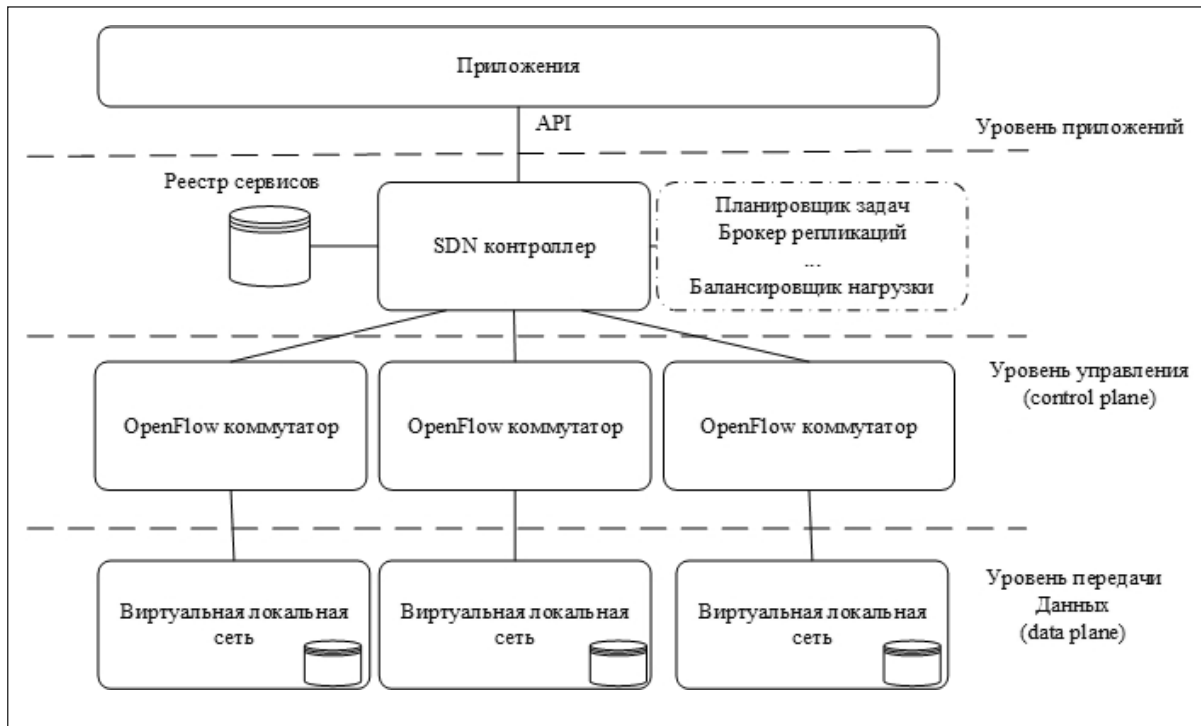


Рис. 1. Архитектура программно-конфигурируемой сети

Процесс предоставления сервисов представляет собой следующую последовательность действий: потребитель сервиса запрашивает определенный набор услуг, требования потребителя посредством OpenFlow коммутатора передаются на контроллер. Контроллер, на основании информации о текущем количестве услуг и их местоположении, формирует ответ, в котором указывает данные для доступа к требуемому сервису. В случае ввода нового сервиса информация о нем передается непосредственно контроллеру, который заносит данные об услуге в каталог сервисов [3; 19; 20].

Процесс предоставления сервисов представляет собой следующую последовательность действий: потребитель сервиса запрашивает определенный набор услуг, требования потребителя посредством OpenFlow коммутатора передаются на контроллер. Контроллер на основании информации о текущем количестве услуг и их местоположении формирует ответ, в котором указывает данные для доступа к требуемому сервису. В случае ввода нового сервиса информация о нем передается непосредственно контроллеру, который в свою очередь, заносит данные об услуге в каталог сервисов [3; 19; 20]. Детализированный сценарий формирования и схема предоставления комплексного сервиса в SDN сети приведены на рис. 2.

Предположим, что множество потребителей сервиса определено как множество

$C = \{c_1, c_2, \dots, c_m\}$, множество предоставляемых сервисов определено как $SC = \{SC_1, SC_2, \dots, SC_n\}$, где SC_i – комплексный сервис, который формируется посредством композиции атомарных сервисов sa_{ki} : $SC_i = \{sa_{i1}, sa_{i2}, \dots, sa_{ki}\}$.

Результирующее решение о формировании и предоставлении комплексного сервиса осуществляется на основании информации об атомарных сервисах, входящих в его состав и их месторасположении в сети.

SDN обладает множеством локальных каталогов сервисов (рис. 1). Каждый локальный каталог сервисов $KC_y[sa]$, закреплен за определенной VLAN или LAN, сформированной посредством объединения нескольких OpenFlow коммутаторов [1; 3]. Каждый $KC_y[sa]$ имеет свою область обслуживания, границы которой определяются поставщиком услуг в процессе конфигурирования сети.

В случае использовании такого подхода информация о каждом i -м сервисе SC_i , которая хранится в локальном каталоге сервисов $KC_y[sa]$, содержащемся в каждой зоне обслуживания, характеризуется множеством следующего вида:

$$KC_y[sa] = \{l(sa), c(sa), t(sa), p(sa), r(sa)\}, \quad (1)$$

где $l(sa)$ – локация атомарного сервиса, $c(sa)$ – стоимость сервиса, $t(sa)$ – время отклика, $p(sa)$ – производительности вычислительного узла, на котором размещен сервис, $r(sa)$ – надежность.



Рис. 2. Сценарій надання сервісів в програмно-конфігурованих мережах

В глобальному каталозі сервісів міститься повна інформація про сервісах, надаваних в даній мережній архітектурі. Структура глобального каталогу представляє собою упорядкований масив виду:

$$KCG[sa] = \{KC_{i1}(sa), KC_{i2}(sa), \dots, KC_{ij}(sa)\}. \quad (2)$$

Результуюче значення при формуванні комплексного сервісу в SDN має функція інтеграції атомарних сервісів, яка дозволяє визначати значення/вагові коефіцієнти кожного атомарного сервісу, що входить до складу комплексного. Функція інтеграції атомарних сервісів має наступний вигляд:

$$IF(SC_i) = K_1(sa_{i1}) + K_2(sa_{i2}) + K_3(sa_{i3}) + \dots + K_k(sa_{ik}), \quad (3)$$

де K_1, K_2, \dots, K_k – вагові коефіцієнти атомарних сервісів. Результуюче рішення про склад комплексного сервісу визначається виходячи з вагових коефіцієнтів атомарних сервісів.

SDN контролер аналізує склад комплексного сервісу, визначає композицію і місцезнаходження атомарних сервісів в залежності від ряду факторів: навантаженості мережних ресурсів, політики обслуговування постачальника послуг, пріоритету і об'ємів надаваної послуги і т.д.

Основними вимогами, при формуванні комплексного сервісу є такі [21]:

- наявність кінцевого множини атомарних сервісів;
- QoS показники і вартість результуючого комплексного сервісу повинні відповідати вимогам до якості обслуговування, затвердженим в SLA:

$$QoS(SC_i) = K_1(sa_{i1}) + K_2(sa_{i2}) + \dots + K_k(sa_{ik}) = \sum_{k=1}^n K_k \cdot sa_{ki} \rightarrow \max; \quad (4)$$

$$QoS(SC_i) \geq QoS_{SLA}(SC_i).$$

$$C_s(SC_i) = c_1 \cdot sa_{i1} + c_2 \cdot sa_{i2} + \dots + c_k \cdot sa_{ik} = \sum_{k=1}^n c_k \cdot sa_{ki} \rightarrow \min; \quad (5)$$

$$C_s(SC_i) \leq C_{sSLA}(SC_i),$$

де $QoS(SC_i)$ – результуюче значення якості комплексного сервісу; $QoS_{SLA}(SC_i)$ – якість комплексного сервісу, узгоджене в SLA, c_1, c_2, \dots, c_k – вартість атомарних сервісів, $C_s(SC_i)$ – результуюче значення вартості комплексної послуги, $C_{sSLA}(SC_i)$ – значення вартості, відповідне SLA.

III. Метод динамічної реплікації сервісів в програмно-конфігурованих мережах

Представлений метод реплікації включає в себе наступні етапи:

Прийняття рішення про необхідність реплікації сервісів

Прийняття рішення про необхідність реплікації сервісів здійснюється на основі результатів аналізу його популярності і ступеня деградації QoS з ростом його популярності. Як відзначено в роботах [16, 18] значення порога реплікації впливає на частоту формування і розподілу копій сервісу.

Вибір значення порога реплікації залежить від ряду факторів: об'єму даних, які повинні бути передані при реплікації, значення ToS (Type of Service) і поточних фізичних характеристик мережі, що впливають на доступність послуги. Тоді порогове значення реплікації може бути представлено як:

$$Th(SC_i) = f(val(SC_i), ToS(SC_i), A(SC_i)), \quad (7)$$

где $val(SC_i)$ – объем данных, переданных по сети в процессе предоставления сервиса, $ToS(SC_i)$ – приоритет сервиса, $A(SC_i)$ – значение доступности сервиса, $A(SC_i)$ носит переменное значение, зависящее от таких факторов, как текущая популярность сервиса и значение производительности вычислительного узла, на котором расположен сервис.

Взаимосвязь между порогом репликации и производительностью вычислительного узла, на котором размещен сервис, определяется следующим образом:

$$Th(SC_i) = K_R \cdot p(SC_i), \quad (8)$$

где K_R – коэффициент, определяющий требования к репликации. Если значение $K_R < 1$, то репликация выполняется с целью предотвращения отказов в обслуживании. При $K_R > 1$ основной целью репликации является разгрузка вычислительного узла при значительном увеличении потока заявок.

Если количество запросов на предоставление i -го сервиса, поступивших из одного фрагмента сети, превышает порог репликации:

$$cnt(Req(SourceIP), KC_j[SC_i]) > Th(SC_i) \quad (9)$$

и приводит к ухудшению качества сервиса, то брокер принимает решение о необходимости запуска процесса репликации.

Определение количества реплик и их расположение в сети

С целью обеспечения гарантированного качества услуг в процессе репликации данные переносятся в тот фрагмент сети, где популярность услуги превышает граничный предел.

Учитывая тот факт, что запрашиваемый пользователем сервис не всегда может находиться в рамках области обслуживания локального каталога сервисов $KC_j[SC_i]$, а также то, что популярность удаленного сервиса может со временем увеличиться, в распределенных сетях, в частности SDN применяется два вида репликации: локальная и удаленная [18; 20].

Необходимость в локальной репликации возникает тогда, когда к определенному сервису в сети резко возрастает количество запросов от потребителей этой сети и это приводит к появлению отказов. Таким образом, создание дополнительной реплики при локальной зоне позволяет повысить доступность и производительность запрашиваемого сервиса, путем перераспределения запросов между репликациями.

Необходимость в удаленной репликации возникает в том случае, когда количество запросов

из одного участка сети к определенному удаленному ресурсу значительно возрастет. С целью уменьшения трафика передаваемого по сети, а также сокращения времени доступа к ресурсу запускается процесс репликации. Основное преимущество использования удаленной репликации состоит в том, что доступ к сервисам может выполняться локально, без поглощения сетевого трафика и задержек.

Количество реплик i -го сервиса, распределенных между фрагментами сети, определяется следующим образом:

$$n_R(SC_i) = \left\lfloor \frac{F_{R(SourceIP)}(SC_i)}{\sum F_R(SC_i)} \times mSC_i \right\rfloor, \quad (10)$$

где $F_{R(SourceIP)}(SC_i)$ – фактор репликации, соответствующий определенному фрагменту сети, $\sum F_R(SC_i)$ – суммарный фактор репликации, mSC_i – общее количество созданных копий сервиса.

Проверка непротиворечивости реплик

При выполнении синхронной репликации в качестве протокола обеспечения непротиворечивости предложено использовать протокол первичного архивирования [21]. В случае внесения изменений в первичный экземпляр сервиса информация о его обновлении пересылается в локальный каталог сервисов, принадлежащий данной зоне, и глобальный каталог, находящийся на уровне управления. SDN контроллер обеспечивает проверку содержимого глобального каталога и в случае, если обнаружены копии услуги, посредством протокола OpenFlow выполняется обновление обнаруженных копий.

На основании решения приведенных выше задач (2) – (8) может предложен комплексный метод динамической репликации сервисов в SDN (рис. 3).

Приведенный на рис. 3 комплексный метод динамической репликации включает выполнение трех этапов: своевременное принятие решения о необходимости формирования реплики сервиса и проведения операции репликации; определение количества и выбор месторасположения реплик; обеспечение непротиворечивости новой реплики с существующими сервисами. Применение данного метода позволяет системе управления гибко адаптировать сетевые ресурсы в соответствии с требованиями пользователей, что способствует оперативному предоставлению сервисов и повышению уровня качества обслуживания в SDN.

Выводы. Внедрение концепции программно-конфигурируемых сетей позволяет повысить уровень качества предоставляемых сервисов. Главным образом, повышение достигается за счет перераспределения функций: функции передачи

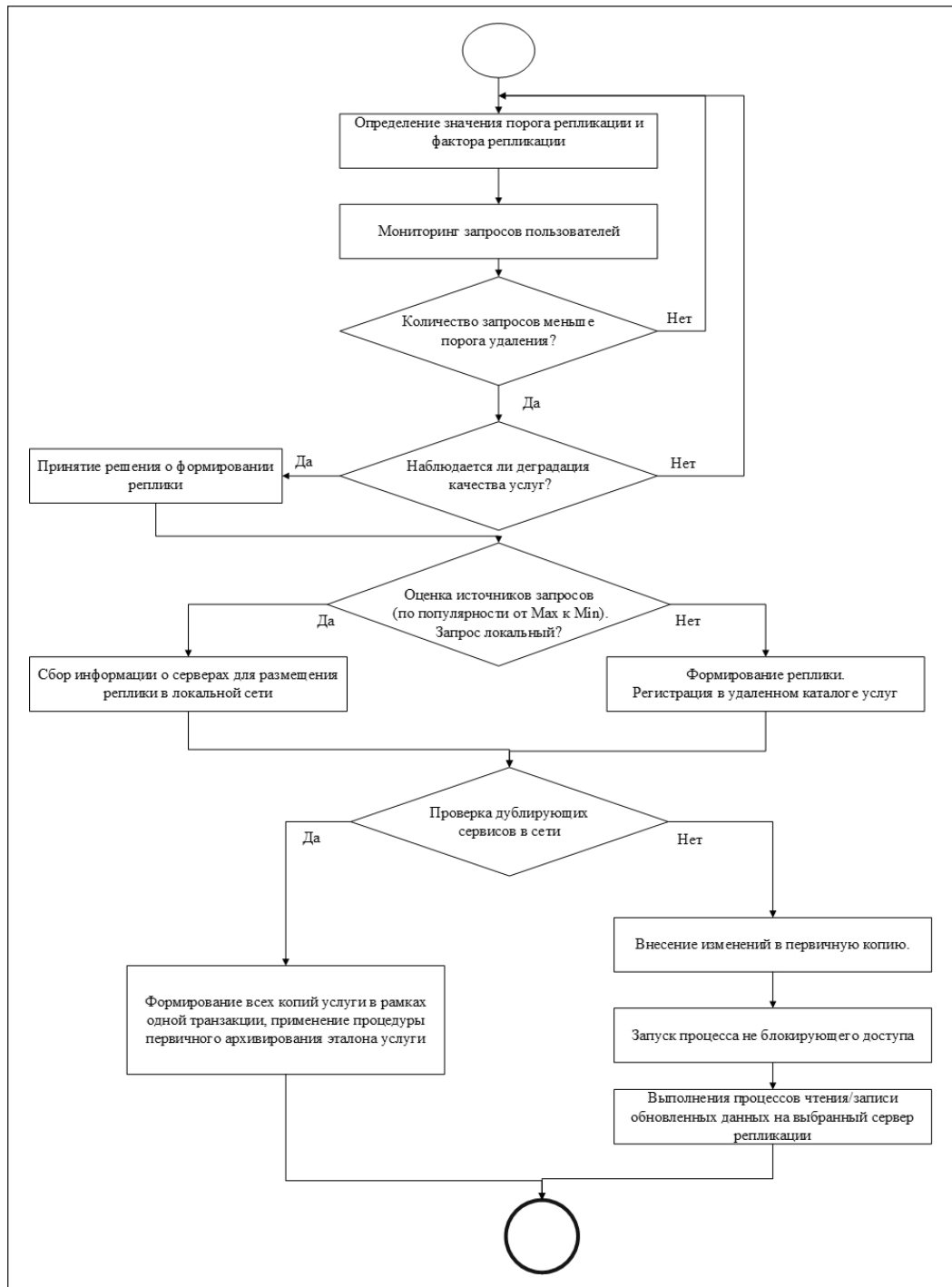


Рис. 3. Метод динамической репликации сервисов

данных выносятся на периферийное и окончное оборудование, в то время как функции управления закрепляются за контроллером.

Однако при использовании концепции SDN ощутимое улучшение качества сервисов затруднительно в силу следующих причин: отказы систем хранения данных, избыточность сетевых ресурсов, невысокая отказоустойчивость периферийного оборудования и т.д.

В статье предлагается подход, позволяющий повысить качество предоставляемых сервисов путем внедрения разработанного метода динамической репликации. Предлагаемый метод состоит из следующих этапов: выбора момента репликации, расчета количества и месторасположения реплик, проверка непротиворечивости реплики существующим. В качестве критериев принятия решения о необходимости репликации

выступают порог репликации, популярность сервиса, производительность вычислительных устройств и локация запросов. Применение предложенной модели формирования и предо-

ставления сервисов, а также динамического метода репликации позволяет своевременно определять причины деградации качества сервисов и устранять их.

Список литературы:

1. SDN architecture. Open Networking Foundation. 2016. 68 p. URL: https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR_SDN_ARCH_1.0_06062014.pdf (viewed: 12.09.2018).
2. Feamster N., Rexford J., Zegura E. The Road to SDN: An Intellectual History of Programmable Networks. ACM Queue. 2013. P. 20–40.
3. Egawa T. SDN standardization Landscape from ITU-T Study Group. ITU Workshop on SDN, Geneva, Switzerland. 2013. 4 June. 22 p.
4. Costas N. Atanasova I. K. OpenFlow and SDN. Technical Report. Technical Report CESGA-2014-001. 2014. 64 p.
5. Issam S. Mathematical models for analysis Software-Defined Network. Information Technologies & Knowledge. 2015. Vol. 9. No.2. P.111–123.
6. Lawrence S. Software Engineering: Theory and Practice. Upper Saddle River. NJ: Prentice Hall, 2001. 630 p.
7. Infonetics Research, Inc. SDN and NFV Strategies: Global Service Provider Survey. 2014. 39 p. URL: <http://alu.us.neolane.net/res/img/286758382c7e061c52883e873cee02e6.pdf> (viewed: 12.09.2018).
8. Akyildiz I. F. A roadmap for traffic engineering in SDN-OpenFlow networks. Computer Networks. 2014. Vol. 71. No.4. P. 1–30.
9. Stantchev V. Effects of Replication on Web Service Performance in WebSphere // ICSI Technical Report TR-08-003. 2008. 72 p.
10. Yousefi A. Request Replication: An Alternative to QoS Aware Service Selection. Proceedings of the 2011 IEEE International Conference on Service-Oriented Computing and Applications, SOCA. 2011. P. 1–14.
11. Khan S.U., Ahmad I. Comparison and analysis of ten static heuristics-based internet data replication techniques. Parallel Distrib. Comput. 2008. No. 68(2). P. 113–136.
12. Malik S. U., Madani S. A., Khan S. U. Performance analysis of data intensive cloud systems based on data management and replication: a survey. Distributed and Parallel Databases. 2016. Vol.34, No.2. P. 179–215.
13. Chang R. S. Chang H. P. A dynamic data replication strategy using access-weights in data grids. Journal of Supercomputing. 2008. Vol.45. No.3. P. 277–295.
14. Kia H.S. Khan S.U. Server replication in multicast networks. In: 10th IEEE International Conference on Frontiers of Information Technology (FIT). Islamabad, Pakistan. 2012. P. 337–341.
15. Ikeda T., Ohara M., Fukumoto S. A distributed data replication protocol for file versioning with optimal node assignments. In: Proceedings of IEEE International Pacific Rim International Symposium on Dependable Computing 2010. 2010. P. 117–125.
16. Wei Q., Veeravalli B., Gong B. and all CDRM: A cost-effective dynamic replication management scheme for cloud storage cluster. In: IEEE International Conference on Cluster Computing. 2010. P. 188–197.
17. Bonvin N., Papaioannou T.G., Aberer K. A self-organized, fault tolerant and scalable replication scheme for cloud storage. In: Proceedings of the Symposium on Cloud Computing, Indianapolis, USA. 2010. P. 205–216.
18. Khan S., Ahmad I. Heuristic-based replication schemas for fast information retrieval over the internet. In: Proceedings of 17th International Conference on Parallel and Distributed Computing Systems. 2004. P. 278–283.
19. OpenFlow Switch Consortium and Others. OpenFlow Switch Specification Version 1.2.0. OpenFlow Switch Consortium and Others. 2011. URL: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/> (viewed: 12.09.2018).
20. Yang Y., Tang S., Xu Y. and all. An Approach to QoS-Aware Service Selection in Dynamic Web Service Composition . In: 3rd IEEE International Conference on Networking and Services (ICNS). 2007. P.18–23.
21. Дейт К. Дж. Введение в системы баз данных. Москва: Издательский дом «Вильямс», 2005. 1328 с.

КОМПЛЕКСНИЙ МЕТОД ДИНАМІЧНОЇ РЕПЛІКАЦІЇ ДАНИХ У ПРОГРАМНО-КОНФІГУРОВАНИХ МЕРЕЖАХ

У статті запропоновано модель формування та надання комплексного сервісу і метод динамічної реплікації, що дозволяють підвищити доступність сервісів у програмно-конфігурованих мережах. Проведено огляд загальновідомих методів реплікації, які знайшли широке застосування в розподілених мережах, наведено основні переваги та недоліки. На підставі результатів проведеного аналізу та з

урахуванням особливостей формування сервісів у програмно-конфігурованих мережах запропоновано метод, який включає в себе вирішення таких завдань: оцінки популярності сервісу та вибору моменту запуску операції реплікації на підставі значення порога реплікації; обчислення оптимальної кількості реплік і розміщення на обчислювальних вузлах; перевірки несуперечності нової репліки сервісу із тими, що вже існують у мережі.

Ключові слова: мережі SDL, рівень QoS level, протокол OpenFlow, динамічна реплікація, моновалентний сервіс, ToS.

AN INTEGRATED METHOD FOR DYNAMICALLY REPLICATING DATA IN SOFTWARE-CONFIGURABLE NETWORKS

The work is devoted to the increasing quality of services in software-Defined Network by modifying the resource allocation during service provision and services' replication techniques. The well-known replication techniques are analyzed in the work. The result of the analysis shows that existing replication techniques that are used in distributed networks should be revised with the purpose to improve their effectiveness and as a result, increase the availability of services. With the aim to increase the effectiveness of replication techniques in software-defined networking the solution of next tasks are proposed: determining the decision about start replication procedure based on simultaneously increasing of service popularity and degradation of quality of service; determine the required minimum number of replicas and their allocation in the network based on intensive of incoming customers' requests; checking the consistency of new replica of service with almost existing in the network. The complex method of dynamic service replication that is based on this particular solution is proposed. The model of complex services provision is also proposed in the work. The proposed model allows to analytically analyze the value of each monatomic service in complex solution. The case of using the proposed model of the monatomic services that included in complex service can be preplaced by the service with better quality characteristics.

Key words: software-defined networking, QoS level, OpenFlow, dynamic service replication, monatomic service, type of service.