

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ
Факультет физико-математических и естественных наук
Кафедра прикладной информатики и теории вероятностей

УТВЕРЖДАЮ

Заведующий кафедрой
прикладной информатики и
теории вероятностей
д.т.н., профессор
_____ К. Е. Самуйлов
«__» _____ 20__ г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА
на тему
«Моделирование модуля активного управления трафиком сети передачи
данных»

Выполнил

Студент группы НФИбд-01-18

_____ С. М. Наливайко
«__» _____ 20__ г.

Руководитель

доцент кафедры прикладной информа-
тики и теории вероятностей
к.ф.-м.н., доцент
_____ А. В. Королькова

Содержание

Список используемых сокращений	6
Англоязычные сокращения	6
Русскоязычные сокращения	6
Введение	6
1. Моделирование сетей в среде Mininet и анализ их производительности	9
1.1. Обзор исследований в области моделирования в среде Mininet . .	9
1.2. Обзор исследований в области анализа производительности сетей и сетевых компонентов в среде Mininet	10
2. Аналитическая часть курсовой работы	14
3. Практическая часть курсовой работы	15
Заключение	16
Список литературы	17

Список таблиц

Список иллюстраций

Аннотация

Объектом исследования данной выпускной квалификационной работы является способ моделирования модуля активного управления трафиком сети передачи данных. На сегодняшний день существует множество различных методов моделирования, применяемых для исследований сетей передачи данных. Среда виртуального моделирования Mininet, используемая в работе, позволяет использовать реальные сетевые приложения, сетевые протоколы и ядро Unix/Linux для тестирования и анализа характеристик моделируемых в ней компьютерных сетей и сетевых протоколов. Использование Mininet позволяет производить моделирование сети с минимальными временными затратами и минимальными финансовыми издержками.

В процессе написания работы были рассмотрены способы построения сети передачи данных, исследованы сетевые характеристики, такие как пропускная способность сети, длина очереди пакетов на сетевом интерфейсе устройства, размер окна ТСР на компьютере отправителя, круговая задержка. Создан программный комплекс на языке программирования Python, который позволяет создавать сеть и рассматривать ее сетевые характеристики, не прибегая к изменению программного кода. В качестве примера работы с программой были рассмотрены способы создания и исследования сетей имеющие различные сетевые параметры и топологии.

Список используемых сокращений

Англоязычные сокращения

AQM — Active Queue Management — активное управление очередью

Русскоязычные сокращения

ПМП — Планетарный механизм поворота

Введение

Объектом исследования данной выпускной квалификационной работы является способ моделирования модуля активного управления трафиком сети передачи данных, а также иллюстрация применения данной модели для исследований работы реальных сетей и сетевых приложений.

Актуальность

Актуальность темы обусловлена потребностью организаций в грамотном проектировании и развертывании локальной сети предприятия. Создание сети не может обходиться без предварительного анализа и прототипирования. Для данных задач могут использоваться современные программы, которые позволяют создавать и испытывать сети без реальных сетевых компонентов. Такое решение дешево в построении, а сбор данных сетевых характеристик заметно ускоряется и упрощается.

Так же, исследователям в области сетевых технологий требуется проверять качество работы сетевых протоколов или приложений в определенных условиях. В данной ситуации применение программного комплекса, который поможет

автоматически получить приближенные данные эффективности работы сетевого компонента, поможет сократить временные затраты на развертывание и исследование поведения сети.

Цель работы

Целью работы является создание программного компонента, который позволяет моделировать и измерять сетевые характеристики передачи данных без использования реальных сетевых компонентов.

Задачи

1. Построить модуль активного управления трафиком в Mininet.
2. Измерить и визуализировать характеристики моделируемой сети передачи данных для качественной оценки производительности сети.
3. Оценить влияние различных комбинаций протоколов и сетевых топологий на общую производительность моделируемой сети.
4. Исследовать результат моделирования.

Методы исследования

Методами исследования предметной области являются эксперимент, наблюдение, сравнение и измерение. Каждый из этих методов полезен на практике при построении сетей.

Апробация

Результаты, полученные в ходе выполнения работы, были представлены на конференции “Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологических систем” (ИТТММ 2022) (Москва, РУДН, 18–22 Апреля 2022 г.).

Публикации

По теме выпускной квалификационной работы были опубликованы в [XXX публикация, пока ее нет].

Структура работы

Работа состоит из введения, трех глав, заключения и списка используемой литературы из [XXX] наименований. Работа содержит [XXX] страниц текста и [XXX] рисунка и [XXX] приложения, содержащих код написанных программ.

Во введении сформулированы цели и задачи, описана актуальность работы и методы, используемые в работе.

В первой главе приводятся основные сведения о методах моделирования модуля активного управления трафиков сети передачи данных. [Дополнение будет как появится часть 1].

Во второй главе описаны программные средства, с помощью которых создается программный комплекс для моделирования модуля активного управления трафиком и исследования сетевых характеристик элементов сети. [Дополнение будет как появится часть 2].

Третья глава посвящена способу взаимодействия с программным модулем и исследованию сетевых характеристик сетей, имеющих различные топологии и сетевые параметры. [Дополнение будет как появится часть 3].

В заключении представлены результаты и выводы по проделанной работе.

1. Моделирование сетей в среде Mininet и анализ их производительности

1.1. Обзор исследований в области моделирования в среде Mininet

Качество работы сетевого приложения определяется, в значительной степени, качествами линии передачи данных, сетевых устройств и работающих в них алгоритмах обработки потока данных. Современные сети передачи данных не могут обеспечить нужный уровень качества обслуживания по ряду причин. Одна из них – слабая взаимосвязь физических сетевых элементов разных уровней. Контроль передачи данных лежит на каждом устройстве отдельно, что, в некоторых случаях, ведет к многочисленным потерям данных и задержкам. Консорциум ONF (Open Networking Foundation) [8] предложил решение данной проблемы: отделить уровень контроля передачи данных от сетевого устройства и перенести все эти заботы на некоторый контроллер. Управление данным контроллером ведется централизованно с помощью протокола OpenFlow [9], что помогает установить нужный уровень предоставления услуг для конкретного приложения. Такое устройство сети консорциум ONF назвал SDN (Software-defined Networking) [14]. Для изучения производительности сетей SDN методом моделирования организация ONF разработала эмулятор сети Mininet.

В работе [11] речь идет об имитационном моделировании сети SDN с помощью средств Mininet и измерении производительности сети. Были затронуты следующие показатели производительности: RTT (Round Trip Time) [12] для каждого направления связи, пропускная способность ветвей и направлений связи, величина задержки на сетевых элементах, загрузка портов OpenFlow Switch, элементы сети с наибольшей задержкой, число обслуженных и потерянных пакетов. Mininet хорошо подходит для задач имитационного моделирования и исследо-

ваний общей эффективности работы сети, однако, для создания сети требуется умение писать программы и знание Mininet API.

Работа [6] представляет собой подробное руководство по взаимодействию с Mininet, как с помощью CLI [2], так и с помощью написания программ на языке программирования Python. Авторы показывают простоту создания сетевых топологий различной сложности, такие как Minimal, Single, Linear, Tree, Reversed. Также, в работе представлена программа, которая создает собственную сеть внутри Mininet и проводит простой анализ ее производительности. Созданная сеть детерминирована и состоит всего из 3 узлов, соединенных коммутатором. Анализ работы сети не имеет смысла, так как сеть создана, скорее, для демонстрации принципов работы с Mininet, а не для исследований. Из данной работы видно, что Mininet отлично подходит для исследований производительности сети и сетевых компонентов, благодаря своей простоте и настройке.

Идея автоматизации процесса создания топологий сети возникает сама собой, после написания двух-трех программ. Авторы работы [5] описывают способ такой автоматизации. Способ основан на создании конфигурационного файла, в котором описаны основные правила адресации, создания сетевых устройств и соединений между ними. Чтение конфигурационного файла происходит в программе, написанной на языке программирования Python. Данный способ заметно ускоряет развертывание сети, избавляя исследователя от постоянных модификаций программного кода.

1.2. Обзор исследований в области анализа производительности сетей и сетевых компонентов в среде Mininet

Простое моделирование сети не дает информации об ее производительности. Нужно провести качественный анализ работы сетевых компонентов, рассчитать пропускную способность соединения, количество потерянных и доставленных

пакетов и т. д. Работа [13] является хорошим началом для изучения методов анализа производительности сети. Авторы исследуют задержки в сети при передаче данных и сравнивают показатели в HDN и SDN. HDN (Hardware Defined Network) — привычные нам сети, где управление передачи данных лежит на устройствах сетевого и канального уровней. Работа показала, что SDN справляется с работой лучше и средняя задержка передачи данных в ней ниже (3.891 мс против 8.277 мс). Однако, стоит заметить, что здесь речь идет о передаче простого ICMP пакета, а не потока TCP/UDP трафика.

Работа [10] является более интересным примером анализа производительности сети в Mininet. В ней рассматриваются вопросы оценки производительности механизмов для эффективной работы с перегрузки в SDN. Авторы оценивают общую производительность сети, сравнивая ее с производительностью сети, которая использует LLDP (Link Layer Discovery Protocol) [7]. Данный протокол канального уровня позволяет сетевому оборудованию оповещать оборудование, работающее в локальной сети, о своём существовании и передавать ему свои характеристики. Протокол отлично подходит для новой концепции построения сетей SDN, которую мы рассматривали ранее, так как позволяет SDN-контроллеру знать характеристики элементов сети и в зависимости от этого управлять потоками трафика. Оценка производительности основывается на трех пунктах:

- Уровень потери пакетов;
- Уровень доставки пакетов;
- Общая пропускная способность.

Данные оценки определяются для каждой итерации исследования. Всего таких итераций 4, и они имеют следующие сетевые топологии:

1. 1 SDN-контроллер, 4 коммутатора, 1 хост, 4 соединения;
2. 1 SDN-контроллер, 15 коммутаторов, 16 хостов, 30 соединений;
3. 1 SDN-контроллер, 40 коммутаторов, 81 хост, 120 соединений;
4. 1 SDN-контроллер, 85 коммутаторов, 256 хостов, 340 соединений.

В ходе работы были построен график, которые отображают зависимость се-

тевой характеристики от сетевой топологии для обычной сети и сети, которая использует LLDP. На графиках видно, что сеть второго типа показывает лучший уровень производительности для каждой сетевой характеристики передачи данных: ниже уровень потерь пакетов, выше уровень доставки пакетов, выше общая пропускная способность.

Mininet отлично подходит для проведения исследований поведенческих особенностей сетевых компонентов. Однако, можно также провести исследование работоспособности и производительности сетевых протоколов и приложений. Авторы статьи [1] исследуют производительность алгоритма для эффективной работы с перегрузками BBRv2. Анализ работы алгоритма, презентацию и исходный код можно найти в [18]. BBRv1 – нестандартный алгоритм управления перегрузками разработанный в Google, который не использует потерю пакетов как маркер для снижения скорости отправки. Алгоритм BBRv1 [17], в сравнении с предшествующими алгоритмами, выдает большую пропускную способность для потоков данных при равных условиях. Одним из минусов алгоритма является его слабая совместимость с более старыми алгоритмами, используемыми в сети, и алгоритм BBRv2 создан как раз для исправления данной брешы. В исследовании средствами Mininet создается сеть, имеющая 100 узлов отправителей и 100 узлов получателей. Отправители и получатели связаны между собой сетью из 3-х коммутаторов, каждый из которых имеет свою задачу: - коммутатор 1: является точкой входа для узлов-отправителей, эмулирует потери и задержки данных с помощью средств NetEm [15], соединен с коммутатором 2; - коммутатор 2: эмулирует «узкое горлышко» между отправителями и получателями, ограничивая скорость передачи данных до 1 Гбит/с. Ограничивание передачи осуществляется с помощью дисциплины очередей TBF [16]. - коммутатор 3: соединяет коммутатор 2 и хосты получатели.

В ходе работы были исследованы и сравнены алгоритмы CUBIC [3], BBRv1, BBRv2 на такие сетевые характеристики: пропускная способность, индекс справедливости [4], сосуществование. Результаты показывают, что BBRv2 обеспечи-

вает лучшее сосуществование с потоками, использующие алгоритм CUBIC, по сравнению со своим предшественником. Кроме того, BBRv2 способен обеспечить более справедливую долю пропускной способности по сравнению с BBRv1, когда сетевые условия, такие как пропускная способность и задержка, динамически изменяются.

2. Аналитическая часть курсовой работы

3. Практическая часть курсовой работы

Заключение

Здесь могло бы быть ваше заключение

Список литературы

1. A Performance Evaluation of TCP BBRv2 Alpha / J. Gomez [и др.] // 2020 43rd International Conference on Telecommunications and Signal Processing (TSP). — IEEE, 2020.
2. Command-line interface. — URL: https://en.wikipedia.org/wiki/Command-line_interface.
3. CUBIC for Fast Long-Distance Networks : RFC / I. Rhee [и др.] ; RFC Editor. — 02.2018. — № 8312. — URL: <https://www.rfc-editor.org/rfc/rfc8312.txt>.
4. *Floyd S.* Metrics for the Evaluation of Congestion Control Mechanisms : RFC / RFC Editor. — 03.2008. — № 5166. — URL: <https://www.rfc-editor.org/rfc/rfc5166.txt>.
5. Implementation of simplified custom topology framework in Mininet / C. Pal [и др.] // 2014 Asia-Pacific Conference on Computer Aided System Engineering (APCASE). — 2014.
6. *Kaur K., Singh J., Ghumman N.* Mininet as Software Defined Networking Testing Platform // International Conference on COMMUNICATION, COMPUTING & SYSTEMS (ICCCS-2014). — EXCEL INDIA PUBLISHERS, 2014. — С. 139—142.
7. Link Layer Discovery Protocol. — URL: https://en.wikipedia.org/wiki/Link_Layer_Discovery_Protocol.
8. Open Networking Foundation. — URL: <https://opennetworking.org>.
9. OpenFlow. — URL: <https://opennetworking.org/sdn-resources/customer-case-studies/openflow/>.
10. Performance Analysis of Congestion Control Mechanism in Software Defined Network (SDN) / M. Z. A. Rahman [и др.] //. — 2017.

11. RESEARCH OF SDN NETWORK PERFORMANCE PARAMETERS USING MININET NETWORK EMULATOR / O. I. Romanov [и др.] // Information and Telecommunication Sciences. — 2021.
12. Round Trip Time. — URL: [https://developer.mozilla.org/en-US/docs/Glossary/Round_Trip_Time_\(RTT\)](https://developer.mozilla.org/en-US/docs/Glossary/Round_Trip_Time_(RTT)).
13. *Shivayogimath C. N., Reddy N. V. U.* Performance Analysis of a Software Defined Network Using Mininet // Artificial Intelligence and Evolutionary Computations in Engineering Systems: Proceedings of ICAIECES 2015 / под ред. S. S. Dash [и др.]. — Springer, 2016. — С. 391—398.
14. Software-defined networking. — URL: https://en.wikipedia.org/wiki/Software-defined_networking.
15. tc-netem(8) — Linux manual page. — URL: <https://man7.org/linux/man-pages/man8/tc-netem.8.html>.
16. tc-tbf(8) — Linux manual page. — URL: <https://man7.org/linux/man-pages/man8/tc-tbf.8.html>.
17. TCP BBR Quick-Start: Building and Running TCP BBR on Google Compute Engine. — URL: <https://github.com/google/bbr/blob/master/Documentation/bbr-quick-start.md>.
18. TCP BBR v2 Alpha/Preview Release. — URL: <https://github.com/google/bbr/blob/v2alpha/README.md>.