

Санкт-Петербургский государственный университет

Кафедра системного программирования

Группа 22.Б10-мм

Добавление IP-туннеля в Mimetnet

Казбеков Руслан Радикович

Отчёт по учебной практике
в форме «Решение»

Научный руководитель:
старший преподаватель кафедры системного программирования, И. В. Зеленчук

Санкт-Петербург
2024

Оглавление

Введение	3
1. Постановка задачи	4
2. Обзор	5
3. Метод	7
3.1. Создание туннеля	7
3.2. Графический интерфейс	7
3.3. Серверная часть	8
4. Эксперимент	9
Заключение	10
Список литературы	11

Введение

С развитием компьютерных сетей возрастает потребность в специалистах, что приводит к тому, что во многих технических вузах организуются курсы по изучению основ компьютерных сетей.

На Математико-Механическом факультете Санкт-Петербургского государственного университета преподается курс «Компьютерные сети и хранилища данных». Для лучшего усвоения студентами материала был создан Miminet[4] - веб эмулятор компьютерных сетей.

На данный момент Miminet активно развивается и в проект поступают задачи от учебных заведений и компаний. Так университет ЛЭТИ в ходе одной из встреч заметил, что в Miminet отсутствует возможность изучения такой части компьютерных сетей, как туннелирование.

Было предложено добавить в проект возможность создавать туннели, а конкретно, так как проект образовательный, IP-туннели, как наиболее понятные и простые для студентов.

Туннелирование — это процесс, в ходе которого создаётся логическое соединение между двумя конечными точками посредством инкапсуляции различных протоколов. В нашем случае — это инкапсуляция IP пакета в IP пакет. К пакету с полезной нагрузкой добавляется новый заголовок с новыми адресами конечных точек и дополнительной информацией.

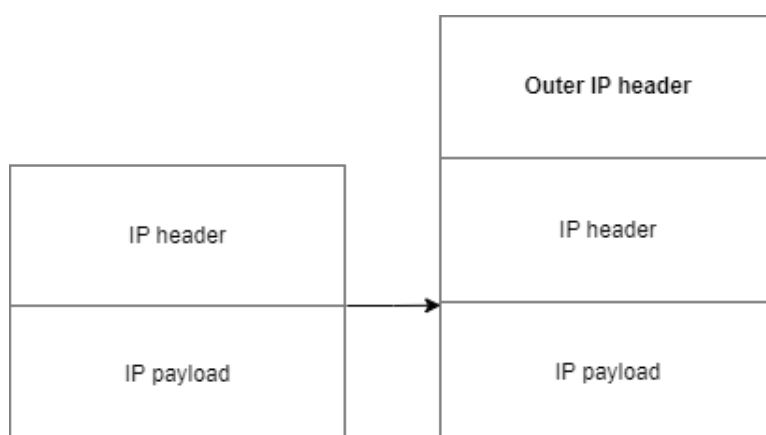


Рис. 1: IP-инкапсуляция

1. Постановка задачи

Целью работы является расширение функциональности Miminet путём добавления возможности создавать IP-IP-туннели. Для её выполнения были поставлены следующие задачи:

1. определить способ создания IP-IP-туннеля;
2. спроектировать и реализовать пользовательский интерфейс;
3. реализовать серверную часть;
4. протестировать реализованную функциональность.

2. Обзор

Были рассмотрены популярные эмуляторы и симуляторы компьютерных сетей, обладающие возможностью создания туннелей.

- Cisco Packet Tracer[1]: Cisco Packet Tracer — это сетевой симулятор, разработанный компанией Cisco Systems, предназначенный для обучения и моделирования сетевых топологий. Этот инструмент позволяет студентам и сетевым инженерам создавать, настраивать и тестировать сетевые среды без необходимости физического оборудования. Packet Tracer поддерживает IPsec VPN туннели. Дополнительное шифрование и настройка туннелей может быть сложной для студентов, не знакомых с консолью и командами сетевого оборудования.
- GNS3[2]: GNS3 (Graphical Network Simulator-3) — это сетевой эмулятор, который позволяет пользователю создавать и моделировать сложные сетевые топологии. Он позволяет комбинировать виртуальные и реальные устройства, используемые для моделирования сложных сетей. Он использует программное обеспечение для моделирования Cisco IOS¹. Этот инструмент часто используется для подготовки к сетевым сертификациям и для реального моделирования сетевых сред. В GNS3 присутствует поддержка различных типов туннелей, включая IPsec, GRE, IPIP и другие. Однако, как и в случае с Cisco Packet Tracer, настройка туннелей в GNS3 также может быть сложной для студентов без опыта работы с Linux и консолью.

В заключение обзора можно сказать, что оба проекта обладают достаточно большой функциональностью по работе с туннелями и позволяют работать с широким спектром возможностей. Однако, так как в них нет полноценной поддержки графического интерфейса работы с туннелями, то их использование требует знаний Linux и опыта работы с консолью, что мешает их использованию в образовательных целях.

¹Cisco IOS — <https://www.cisco.com/c/en/us/products/ios-nx-os-software/index.html>

Поэтому было решено разработать и интегрировать эту функциональность в Miminet.

3. Метод

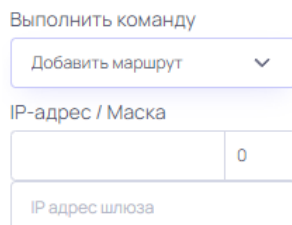
3.1. Создание туннеля

Эмуляция сетей происходит на сервере с системой Linux. С помощью IPMininet[3] и Mininet[5] создаются виртуальные устройства, с которыми можно работать с помощью средств Linux. Для работы с туннелями была выбрана утилита ip[7] из пакета iproute2, как стандарт работы с сетевой подсистемой.

Создание IP-туннеля проводится следующим образом:

1. на устройствах конечных точек создаются интерфейсы туннелей;
2. им присваиваются IP адреса;
3. в таблицу маршрутизации устройств добавляется путь к подсети назначения через созданный интерфейс туннеля.

Последний пункт реализован в Mininet в качестве команды, выполняемой на устройстве. Поэтому было принято решение разделить создание туннеля на создание IP-интерфейса и добавление пути.



Выполнить команду

Добавить маршрут ▼

IP-адрес / Маска

0

IP адрес шлюза

Рис. 2: Интерфейс команды добавления пути

3.2. Графический интерфейс

Так как создание туннеля происходит во время конфигурации топологии сети, то было принято решение выделить его в команду, как и добавление пути.

Выполнить команду

Добавить IPIP-интерфейс ▾

Параметры IPIP-интерфейса

Интерфейс начальной точки ▾

IP конечной точки

IP IPIP-интерфейса

Название IPIP-интерфейса

Рис. 3: Интерфейс команды создания IPIP-интерфейса

В качестве параметров IPIP-интерфейса выступают аргументы, передаваемые серверной части для создания интерфейса:

- адрес начальной точки туннеля, в данном случае выбираемый как интерфейс устройства, у которого был добавлен адрес;
- адрес конечной точки туннеля;
- адрес создаваемого интерфейса;
- название создаваемого интерфейса.

При реализации пользовательского интерфейса были использованы уже интегрированные в проект технологии, такие как JavaScript, jQuery, Bootstrap.

3.3. Серверная часть

Была добавлена новая job — команда, которая выполняется во время конфигурации сети. В ней прописаны консольные команды, которые будут выполнены на job host — устройстве, с которого выполняется команда. Процесс создания интерфейса был описан в разделе 3.1.

4. Эксперимент

Для проверки реализованной функциональности были написаны unit-тесты с использованием библиотеки Pytest[6]. Входные данные для тестов — топология сети, записанная в файле формата JSON. Далее запускается эмуляция сети с заданной топологией, в результате которой получается файл формата JSON с данными о движении пакетов, который, в свою очередь, сравнивается с эталонным. Также каждую неделю проводились собрания с командой, где обсуждались результаты работы и предлагались идеи по улучшению и исправлению функциональности. Эти встречи были важным элементом рабочего процесса, так как позволяли команде обмениваться опытом и находить оптимальные решения для возникающих проблем.

Заключение

В ходе работы были решены следующие задачи:

- определен способ создания IPsec-туннеля;
- спроектирован и реализован пользовательский интерфейс;
- реализована серверная часть;
- проведено тестирование реализованной функциональности;
- код решения² и тестов³ доступен в репозитории на Github;
- функциональность доступна на сайте Miminet⁴.

²<https://github.com/mimi-net/miminet/pull/91>

³<https://github.com/mimi-net/miminet/pull/126>

⁴https://miminet.ru/web_network?guid=359c55a4-1396-467f-9085-daea3127589e

Список литературы

- [1] Cisco Packet Tracer. — URL: <https://www.netacad.com/ru/courses/packet-tracer> (дата обращения: 16 июня 2024 г.).
- [2] GNS3. — URL: <https://www.gns3.com/> (дата обращения: 16 июня 2024 г.).
- [3] IPMininet. — URL: <https://ipmininet.readthedocs.io/en/latest/> (дата обращения: 16 июня 2024 г.).
- [4] Miminet. — URL: <https://miminet.ru/> (дата обращения: 16 июня 2024 г.).
- [5] Mininet. — URL: <https://mininet.org/> (дата обращения: 16 июня 2024 г.).
- [6] Pytest. — URL: <https://docs.pytest.org/> (дата обращения: 16 июня 2024 г.).
- [7] ip. — URL: <https://www.man7.org/linux/man-pages/man8/ip.8.html> (дата обращения: 16 июня 2024 г.).