МИНИСТЕРСТВО ЭНЕРГЕТИКИ, ПРОМЫШЛЕННОСТИ И СВЯЗИ СТАВРОПОЛЬСКОГО КРАЯ

**Государственное бюджетное**

**профессиональное образовательное учреждение**

**«Ставропольский колледж связи**

**имени Героя Советского Союза В.А. Петрова»**

Цикловая комиссия вычислительной техники

УТВЕРЖДАЮ

Заместитель директора по учебной работе

\_\_\_\_\_\_\_\_\_\_\_\_ И.О. Фамилия

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_201\_ г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

Шифр специальности Название специальности

Тема тема тема тема тема тема тема тема тема тема тема

|  |  |
| --- | --- |
|  | Выполнил: Бесленеев Станислав Борисович, группа П141 |
|  | Руководитель: Фамилия И.О. |
|  |  |
|  |  |
| Зав. отделением    Председатель ЦК | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  *(подпись)* И.О. Фамилия  *(подпись)* И.О. Фамилия |
|  |  |
|  |  |
|  |  |

Ставрополь, 2017

**Содержание**

Введение

1 Теоретическое исследование темы

1.1 Актуальность проблемы исчерпания IP адресов

1.2 Технология NAT

1.2.1 Типы NAT

1.3 Разбор протокола STUN

1.3.1 Назначение протокола

1.3.2 Спецификация RF5389

1.3.2.1 Структура протокола

1.3.3 Принцип действия STUN сервера

1.4 Описание метода UDP Hole Punching

2 Реализация программной системы

2.1 Подготовка рабочего пространства

2.2 Написание серверной части

2.1 Написание консольной версии сервера

2.2 Написание сервера с веб интерфейсом

2.2.1 Серверная часть

2.2.2 Клиентская часть

2.3 Написание клиентской части

2.3.1 Модуль для работы с сигнальным сервером

2.3.2 Модуль для работы с UDP Hole Punching

2.4 Взаимодействия клиентской и серверной части

Заключение

Список литературы

Приложение 1

Приложение 2

Приложение 3

**Введение**

С момента появления человека и до наших времен, люди всегда пытались установить связь между собой. Во времена первобытных людей, единственным видом связи был голос. Во времена древнего Египта, Рима, Греции связь осуществлялась с помощью гонцов. Спустя какое-то время люди поняли, что самый эффективный способ передачи информации это - свет. А спустя еще время – электричество.

Открытие электричества, людям позволило передавать информацию на большие расстояния и с большой скоростью. В 1753 году, стали появляться первые телеграфные машинки. Всего за полтора столетия, начиная с момента изобретения телеграфа и до наших дней, человечество освоило такие телекоммуникационные средства, которые позволили ему быть не только информированным, но и мобильным.

На сегодняшний момент эволюция связи человечества остановилась на Интернете. Интернет - всемирная система объединённых компьютерных сетей для хранения и передачи информации. Как и в любой системе передачи информации должен быть адресант (тот, кто посылает сообщение) и адресат (тот, кто получает сообщение). В сети Интернет адресация осуществляется с помощью протокола IP в настоящие время 4-я версия (IPv4) и менее распространённого IPv6.

IP адрес это уникальный идентификатор устройства, который подключен к сети Интернет. Распределением адресов IPv4 занимается американская некоммерческая организация IANA, а также пять региональных интернет-регистраторов, которые ответственны за назначение IP-адресов конечным пользователям на определённых территориях, и локальными интернет-регистраторами, такими как интернет-провайдеры. IPv4 позволяет использовать около 4,22 миллиарда адресов.

Проблема, с которой люди столкнулись 15 апреля 2011 года – это исчерпания адресов IPv4. Одна из пяти региональных интернет-регистраторов ARIN объявила о полном исчерпании IPv4 адресов. С каждым годом проблема начинает касается и других интернет-регистраторов.

Из-за нехватки IP адресов, интернет провайдеры стали брать дополнительную плату за выделенный IP. А для тех, кто не хочет платить выделяется 1 IP адрес на несколько пользователей. Эта технология называется NAT (Network address translation). Он является основным механизмом, уменьшающий скорость исчерпания адресов.

Из-за исчерпания IP адресов появляется трудность инициирования канала связи с тем, кто находится за NAT. Так как за одним выделенным IP адресом могут быть несколько пользователей. Для решения этой проблемы было создана программная система, которая сможет помочь установить канал связи между пользователями, которые находятся за NAT.

**1 Теоретическая часть**

**1.1 Актуальность проблемы**

Как говорилось ранее, проблема исчерпания IP адресов актуальна для тех кому нужен прямой доступ из Интернета. Такая возможность бывает не у всех, так как у интернет провайдеров ограниченное количество внешних IP адресов.

Например, небольшим провайдером выделено от 127 до 500 адресов. Естественно если каждому давать по адресу на всех не хватит. А за отдельную плату могут выделить отдельный «белый» IP адрес. Для решения этой проблемы провайдеры используют NAT.

Интернет поставщики подключают несколько своих клиентов к одному внешнему адресу. Из-за этого соединиться напрямую к компьютеру, который находится за NAT не получится.

**1.2 Технология NAT**

NAT (Network Address Translation) – механизм был придуман в 1994 году. Чаще всего он нужен для подключения локальной сети к Интернету. Дело в том, что теоретически существует 255\*255\*255\*255=4228250625(4 миллиарда адресов). Даже если бы у каждого жителя планеты был всего один компьютер, адресов бы уже не хватало. А на конец 2015 года в мире было 8,1 млрд. подключенных к Интернету устройств. В два раза больше чем возможное количество адресов. Специалистами было принято решение разделить пространство адресов на публичные (белые) и приватные (частные, серые). К последним относятся три диапазона:

* 10.0.0.0/8
* 172.16.0.0/12
* 192.168.0.0/16

Эти адреса могут использоваться в любой локальной сети или в любой частной сети. Следовательно, они могут совпадать. И для решения конфликта IP адресов приходит на помощь NAT. Простыми словами, он заменяет внутренний IP адрес на внешний.

**1.2.1 Типы NAT**

Существует 3 вида NAT

* Статический
* Динамический
* Many-To-One

Статический NAT – самый простой вид NAT. Он преобразует один внутренний адрес в один внешний. При этом все запросы из вне транслируются во внутреннею сеть. Как будто он напрямую подключен к Интернету.

Динамический NAT работает с пулом внешних адресов и динамически преобразует внутренний адрес в свободный из пула внешний адрес.

Many-To-One – у этого вида есть много названий. Например, PAT (Port Address Translation), NAT Overload. Является самым распространённым видом NAT. Его можно встретить на любом домашнем маршрутизаторе. Из названия понятно, что он транслирует несколько внутренних IP в один внешний. Именно этот вид позволяет решить проблему нехватки адресов и пускать в интернет всех желающих.

Рассмотрим работу NAT Overload на примере. Например, есть 2 хоста в одной локальной сети, имеющие следующие IP адреса

* 192.168.10.101
* 192.168.10.102

Оба хоста запрашивают страницу с сервера 74.125.232.227 (google.com). Подробно видно в таблице 1.

Таблица 1

Запрос к серверу

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес отправителя | Порт отправителя | Адрес назначения | Порт назначения |
| 192.168.10.101 | 5648 | 74.125.232.227 | 443 |
| 192.168.10.102 | 3912 | 74.125.232.227 | 443 |

Когда пакеты доходят до локального маршрутизатора, маршрутизатор проводит декапсуляцию пакета. Он заменяет адрес отправителя на внешний адрес, а порт на свободный любой порт. Далее маршрутизатор создает таблицу маршрутизации, где показано соответствие внутреннего IP:PORT с внешним. В таблице 2 показан пример таблицы маршрутизации.

Таблица 2

Таблица маршрутизации

|  |  |  |  |
| --- | --- | --- | --- |
| Внутренний адрес | Внутренний порт | Внешний адрес | Внешний порт |
| 192.168.10.101 | 5648 | 91.225.32.19 | 11235 |
| 192.168.10.102 | 3912 | 91.225.32.19 | 43852 |

После создания таблицы пакет отправляется адресату. На веб-сервер пришло 2 разных запроса с одного IP адреса, но с разных портов. Сервер после получения запроса, отправляет ответ на внешний адрес и порт хоста как показано в таблице 3

Таблица 3

Ответ от сервера

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес отправителя | Порт отправителя | Адрес назначения | Порт назначения |
| 74.125.232.227 | 443 | 91.225.32.19 | 11235 |
| 74.125.232.227 | 443 | 91.225.32.19 | 43852 |

Как только сообщение доходит до локального маршрутизатора он в соответствии созданной таблице маршрутизации (см. таблица 2) сверяет какому хосту послать дальше пакет.

Теперь немного о видах NAT Overload. Виды классифицируются по методу трансляции пары внутреннего и внешнего IP:PORT. Вот список существующий видов:

* Симметричный NAT (Symmetric NAT)
* Cone NAT, Full Cone NAT
* Address-Restricted cone NAT
* Restricted cone NAT

Симметричный NAT – осуществляет трансляцию, при котором каждое соединение транслируется в случайный уникальную случайную пару IP:PORT. Инициация подключения из публичной сети невозможна.

Full Cone NAT - взаимная трансляция между парами «внутренний адрес: внутренний порт» и «публичный адрес: публичный порт». Любой внешний хост может инициировать соединение с внутренним хостом.

Address-Restricted cone NAT - Постоянная трансляция между парой «внутренний адрес: внутренний порт» и «публичный адрес: публичный порт». Любое соединение, инициированное с внутреннего адреса, позволяет в дальнейшем получать ему пакеты с любого порта того публичного хоста, к которому он отправлял пакет(ы) ранее.

Port-Restricted cone NAT — Трансляция между парой «внутренний адрес: внутренний порт» и «публичный адрес: публичный порт», при которой входящие пакеты проходят на внутренний хост только с одного порта публичного хоста — того, на который внутренний хост уже посылал пакет.

**1.3 Разбор протокола STUN**

**1.3.1 Назначение протокола**

Как говорилось ранее, NAT при инициировании подключения с локальной сети транслирует пару «внутренний адрес: внутренний порт» в пару «внешний адрес: внешний порт». Из-за этого проблематично узнать свой внешний адрес и порт, а это бывает нужно, когда необходимо принять подключение из публичной сети. На помощь приходит протокол STUN.

STUN расшифровывается как (Simple Traversal of User Datagram Protocol (UDP) through Network Address Translators) (упрощенное прохождение UDP пакетов через NAT). Его основной задачей является, помогать устройствам находящиеся за NAT узнать свой публичный адрес и порт. Его второстепенной задачей является обнаружение проброса портов, сделанным шлюзом. STUN не является решением обхода NAT. Скорее, это инструмент, помогающий это достичь.

**1.3.2 Спецификация RF5389**

Структура протокола STUN описана в спецификации RFC 5389. RFC (Request for Comments) —это документ из серии пронумерованных информационных документов Интернета, охватывающих технические спецификации и Стандарты, широко используемые во Всемирной сети. Этот документ описывает работу протокола как на сервере, так и на клиенте. Для продолжения описание протокола необходимо ознакомиться с некоторыми терминами.

STUN агент - это объект, который реализует STUN протокол. Объектом может быть либо STUN-клиент, либо STUN сервер.

STUN клиент - это объект, который отправляет запросы STUN серверу. Клиент STUN также может отправлять свои показания. В этой спецификации термины STUN-клиент и клиент синонимы.

Сервер STUN - это объект, который получает запросы от клиента и отправляет STUN ответы. Сервер STUN также может отправлять свои показания. В этом описании термины STUN-сервер и Сервер являются синонимами.

Адрес транспорта - комбинация IP-адреса и номера порта (Например, номер порта UDP или TCP).

Рефлексивный транспортный адрес – это транспортный адрес, полученный клиентом показывающий, как клиент виден в публичной сети.

Сопоставленный адрес – это тоже самое что рефлексивный адрес. Он сохранился только по историческим причинам и из-за наименования атрибутов MAPPED-ADDRESS и XOR-MAPPED-ADDRESS.

Long-Term Credential – это долгосрочный логин связанный с определенным паролем. Необходим для предоставление доступа клиенту к серверу.

Long-Term Password – это долгосрочный пароль

Short-Term Credential – это временный логин и пароль. Обычно выдается на время до 5 минут.

Short-Term Password – это временный пароль

STUN Indication (STUN Индикация) – Сообщения на которые не ожидается ответ.

RTO: Retransmission TimeOut, определяет начальный период времени между передачей запроса и первой повторной передачей этого запроса.

**1.3.2.1 Структура протокола**

Сообщения STUN кодируются в двоичном формате, используя формат, ориентированный на сеть. Все сообщения должны начинаться с 20-байтового заголовка, за которым следует атрибуты.

Заголовок STUN содержит тип сообщения STUN, Magic Cookie, идентификатор транзакции и длину сообщения. Подробней видно в таблице 4

Таблица 4

Структура заголовка сообщения STUN

|  |  |  |
| --- | --- | --- |
| 00 | Тип сообщения STUN | Длина сообщения |
| Magic Cookie | | |
| ID Транзакции (96 бит) | | |

Наиболее значимые первые 2 бита каждого сообщения STUN. Они должны быть нулями. Это можно использовать для дифференциации пакетов STUN из других протоколов, когда STUN мультиплексируется с другими протоколами на одном и том же порту.

Тип сообщения определяет класс сообщения (запрос, ответ успеха, ответ отказа или указание) и метод сообщения (основная функция). Хотя всего существует 4 типа класса сообщения.

В протоколе существует только два типа транзакций: транзакции запроса/ответа и операции индикации. Классы ответов разбиты на ошибки и успешные ответы.

Поле Magic Cookie должно содержать значение 0x2112A442. Magic Cookie позволяет различать пакеты STUN из пакетов других протоколов, когда STUN мультиплексируется с другими протоколами на одном и том же порту.

**1.3.3 Принцип действия STUN сервера**

STUN сервер – это сервер находящийся в публичной сети, который позволяет узнать свой публичный IP: PORT транслируемый локальным NAT.

Работу сервера осуществляется следующими шагами:

1. Сервер слушает порт 3478 на входящие сообщения
2. При получении сообщения, сервер в соответствии со спецификацией RF5389 расшифровывает входящее сообщение
3. Если сообщение было составлено корректно, сервер составляет ответ, в который он включает IP: PORT отправителя
4. Ответ посылается обратно.
5. В зависимости от вида NAT сообщение может быть получено или нет.

**1.4 Описание метода UDP Hole Punching**

User Datagram Protocol Hole Punching обычно используется в приложениях использующие NAT, для проброса UDP пакетов через NAT. Эта технология обычно нужна для клиент-клиент сетевых приложений.

UDP Hole Punching часто используется в торрентах, peer-2-peer соединениях, Direct Client-to-Client (DCC), VoIP (голос через интернет), Skype и много других приложения или протоколах где необходимо прямое соединение с клиентами.

UDP Hole Punching является методом установления двунаправленного UDP соединения между узлами сети в частных сетях, использующие трансляторы сетевых адресов. Метод не применим во всех случаях и со всеми типами NAT, так как эксплуатационные характеристики NAT не стандартизированы. Для инициации соединения требуется третья сторона- сервер, который виден обоим компьютерам. Обычно это STUN сервера

Для того что бы напрямую установить связь между 2-мя хостами необходимо сделать несколько шагов:

* Узнать внешний IP и порт удаленной машины. Для этого воспользуемся STUN – сетевым протоколом, который позволяет определить внешний IP-адрес.
* Передать эту информацию другому хосту
* Установить соединение и использовать его далее для обмена данными

Схематично представлено на рисунке 1

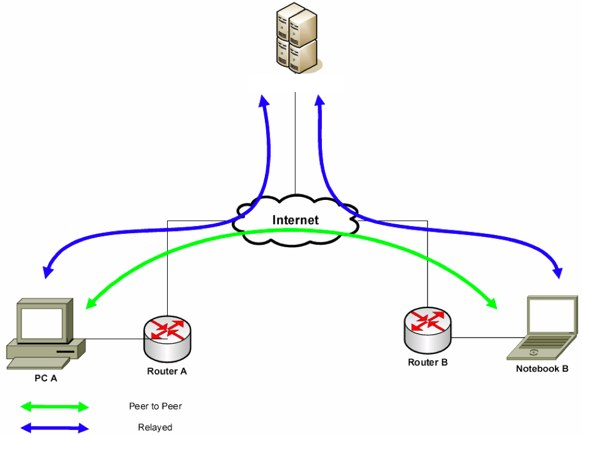


Рис. 1. Упрощённая схема UDP Hole Punching

Предположим A и B являются хостами, которые хотят установить двунаправленное соединение между собой и находящиеся в приватной сети. NA и NB являются NAT устройствами с глобальными адресами EIPA и EIPB. S является публичным сервером с адресом известным A и B.

* A и B начинают UDP соединение с сервером S. NAT устройства NA и NB транслируют UDP адрес и задает временный внешний порт EPA и EPB соответственно.
* Сервер S исследует UDP пакет и извлекает исходящий адрес A и B (внешний порт EPA и EPB)
* Сервер S посылает EIPA: EPA клиенту B и EIPB: EPB клиенту А. Теперь оба клиента знают внешние адреса для инициации подключения.
* Клиент А отправляет пакет на адрес EIPB: EPB
* NA исследует пакет создает соответствующую запись в таблице маршрутизации {Source-IP-A, EPA, EIPB, EPB}
* Клиент B отправляет пакет на адрес EIPA: EPA
* NB исследует пакет создает соответствующую запись в таблице маршрутизации {Source-IP-B, EPB, EIPA, EPA}
* Когда первый пакет B достигает NAT, в зависимости от состояния таблицы маршрутизации NA устройства, пакет может быть отброшен (если запись не была создана) или пропущен дальше в локальную сеть к клиенту А
* Когда первый пакет А достигает NAT, в зависимости от состояния таблицы маршрутизации NB устройства, пакет может быть отброшен (если запись не была создана) или пропущен дальше в локальную сеть к клиенту B
* Как только пакет А дошел до пакета B и как только пакет B дошел до А, можно сказать что связь установлена. Теперь А и B могут обмениваться сообщениями, несмотря на то, что они находятся за NAT устройствами.

Подробный процесс инициирования соединения между 2-ми хостами можно увидеть на рисунке 2

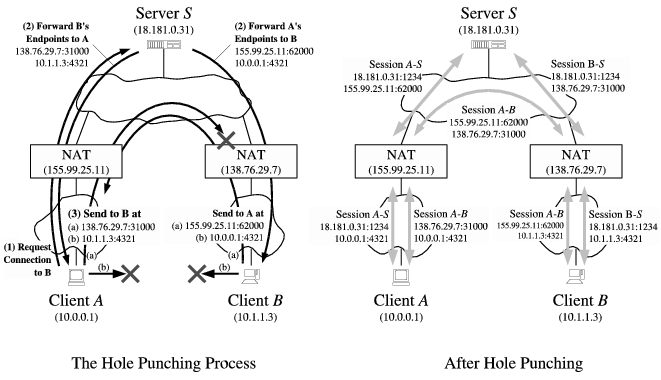


Рис. 2. Подробный процесс UPD Hole Punching

**2 Реализация программной системы**