ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет Санкт-Петербургская школа
физико-математических и компьютерных наук

Нефедов Андрей Сергеевич

Оценка различий луковичных маршрутизаторов LLARP и I2P и улучшение LLARP

Выпускная квалификационная работа - БАКАЛАВРСКАЯ РАБОТА
по направлению подготовки *01.03.02 Прикладная математика и информатика*
образовательная программа «Прикладная математика и информатика»

Рецензент
канд. физ.-мат. наук
А.Б. Левина

Руководитель
канд. физ.-мат. наук,
О.Ю. Иванова

Консультант
Jeff Becker

Санкт-Петербург 2022

Оглавление

Аннотация

Существует много систем, которые обеспечивают анонимное общение через Интернет. Эти системы используют разные подходы, одним из которых является луковичная маршрутизация. Но даже среди систем, использующих этот подход, есть много деталей реализации, которые влияют на анонимность, удобство и другие характеристики системы.

В этой статье сравниваются два луковичных маршрутизатора I2P и LLARP. Приводится список из 12 характеристик, по которым анализируются достоинства и недостатки этих систем. После этого предлагаются изменения в LLARP, которые исправят некоторые из его недостатков. А именно, описывается возможность использования поля flowlabel из стандарта IPv6 для изоляции ключей, и использование QUIC в качестве транспортного протокола. Обозначается какие именно проблемы эти изменения решают. Затем эти изменения реализуются в коде LLARP, и проводится их проверка и оценка.

Ключевые слова: анонимность, луковичная маршрутизация, шифрование, сетевые протоколы.

Many systems exist which provide anonymous communication over the Internet. These systems use different approaches, one of which is onion routing. But even among the systems using this approach there are many implementation details, which impact anonymity, convenience and other characteristics of the system.

This article compares two onion routers I2P and LLARP. A list of 12 characteristics is presented, according to which the advantages and disadvantages of these systems are analyzed. After that, changes to LLARP are proposed that will correct some of its shortcomings. Namely, the paper describes the possibility of using the flowlabel field from the IPv6 standard for isolating keys, and using QUIC as a transport protocol. It is indicated exactly what problems these changes solve. The changes are then implemented in the LLARP code and consequently are validated and evaluated.

Keywords: anonymity, onion routing, overlay networks, encryption, network protocols.

Введение

Анонимные способы обмена информацией имеют большое количество приложений. Некоторые из их социально важных применений это журналистика, информаторы и диссидентские движения. Анонимность также помогает избежать цензуры.

Сеть Интернет не обеспечивает достаточных гарантий анонимности. Основная проблема заключается в том, что протокол IP, который используется для общения в Интернете, несёт IP -адрес, поэтому каждый компьютер, через который проходит сообщение, знает IP -адреса источника и пункта назначения сообщения. Более того, пункт назначения знает IP -адрес источника и наоборот. Это означает, что даже если в сообщении не несёт информацию о личностях участников, метаданные, используемые для доставки сообщения, все ещё позволяют получить информацию о участниках.

Эта проблема может быть решена различными методами. Одним из них является так называемая луковичная маршрутизация. Существуют различные реализации этой концепции, такие как Tor (The Onion Router, луковичный маршрутизатор), I2P (Invisible Internet Project, проект невидимого интернета) и LLARP (Low-Latency Anonymous Routing Protocol, протокол анонимной маршрутизации с низкой задержкой). Хотя все они используют идею луковичной маршрутизации, они имеют разные детали, которые также влияют на анонимность. Отсутствуют углублённые сравнения этих деталей.

В этой работе произведено сравнение анонимных

луковичных маршрутизаторов LLARP и I2P в ключевых параметрах и приведены улучшения LLARP в отношении некоторых из этих параметров.

Определения ключевых терминов

Луковичная маршрутизация Луковичная маршрутизация – способ передачи сообщений, когда сообщение зашифровано несколько раз, так что каждый маршрутизатор, через который оно проходит, знает только о предыдущем и следующем маршрутизаторе в цепочке передачи. [1]

Луковичный маршрутизатор Луковичный маршрутизатор – программное обеспечение, которое используют клиент, сервер и промежуточные компьютеры. Оно производит передачу сообщений и общение с серверным и клиентским программным обеспечением. [1]

Туннель Туннель – набор компьютеров, используемых для передачи сообщения. Набор выбран клиентским маршрутизатором. Может быть либо однонаправленным, либо двухнаправленным. [2]

Роутер Роутер – один из компьютеров в туннеле.

Цель и задачи

Цель данной работы: произвести сравнение LLARP и I2P, обозначить преимущества и недостатки каждой из сетей,

исправить некоторые из недостатков LLARP.

Задачи:

1. Выделить ключевые характеристики анонимизирующих сетей.
2. Выявить преимущества и недостатки LLARP и I2P по ключевым характеристикам.
3. Предложить улучшения в LLARP.
4. Реализовать улучшения в LLARP:
   (a) Реализовать поддержку изоляции ключей через механизм flowlabels.
   (b) Реализовать поддержку QUIC в качестве транспортного протокола.
5. Проверить и оценить улучшения.

Структура работы

В главе 1 приведён обзор на существующие статьи о луковичных маршрутизаторах.

В главе 2 приводятся характеристики луковичных маршрутизаторов и оценка LLARP и I2P с точки зрения этих характеристик.

Глава 3 посвящена реализации изолирования ключей в LLARP.

Глава 4 посвящена реализации использования QUIC в качестве протокола транспортного уровня в LLARP.

В последней главе обобщаются полученные результаты.

1. Обзор литературы

1.1. Существующие сравнения

Одним из первых примеров анонимных способов обмена сообщениями были миксирующие сети Чаума [3]. Он предложил отправлять сообщение через несколько компьютеров в сети которые называются ”миксеры”, где источник и пункт назначения будут скрыты с многоуровневым использованием криптографии открытого ключа. В этой реализации миксеры задерживают и переупорядочивают сообщения, которые они получают, чтобы внешний наблюдателю было труднее понять, какое входящее сообщение соответствует какому исходящему.

Затем появились практические реализации этих идей. Их можно разделить на две категории: протоколы с высокой задержкой, и протоколы с низкой задержкой. Первые были разработаны, чтобы обеспечить защиту от атак по времени, в то время как последние гораздо более подходят для в реальном применений. Примеры конструкций с высокой задержкой включают Mix-master, Babel, и Mixminion. Tor, I2P и LLARP – луковичные маршрутизаторы с низкой задержкой.

Другие луковичные маршрутизаторы были довольно глубоко исследованы, например Tor в [4]. Данная статья предоставляет детальное описание архитектуры Tor, но не выполняет глубокое сравнение с другими системами. Тем не менее, она помогает понять некоторые из дизайнерских решений, принятых LLARP, поскольку на него влияли как I2P, так и Tor.

[2] представляет введение в I2P вместе со списком его сильных и слабых сторон. Он также содержит краткое сравнение с Tor. Основное отмеченное отличие состоит в том, что I2P не предназначен для доступа к обычным Интернету, в то время как Tor предоставляет эту возможность в качестве основной функции. Статья также содержит краткое руководство по работе с I2P.

Tor и I2P сравниваются в [5]. Сравнение не перечисляет некоторые различия, которые важны при сравнении LLARP с I2P. Оно также содержит сравнение организационных характеристик, таких как финансирование и количество разработчиков, которые не являются предметом текущей статьи. Для характеристик, которые сравниваются, выводы не являются углублёнными и не имеют комментариев, и поэтому они менее надёжны.

[6] также содержит сравнение между Tor и I2P. Сравнение рассказывает о сравниваемых деталях и их влиянии на анонимность и другие характеристики. Эта статья выбирает более широкий набор характеристик для сравнения, которые более актуальны для различий между I2P и LLARP.

Oxen (ранее Lokinet, основной проект, использующий LLARP) опубликовал sстатью [7] с описанием принципов работы сети. В то время как он фокусируется на дизайне Oxen, а не LLARP, он содержит небольшое описание функций Tor, I2P и LLARP. Но это описание не является углублённым сравнением.

Многие документы обсуждают некоторые проблемы с определёнными луковичными маршрутизаторами (например, [8]). Хотя они указывают на проблемы в луковичных

маршрутизаторах и решения этих проблем, они не анализируют различия между системами. Но они обеспечивают лучшее понимание деталей реализации. [8] предоставляет хорошее описание внутренней работы I2P, в частности, его алгоритмы выбора нод сети.

[9] представляет подход к поиску реальных IP-адресов серверов в сети I2P.

[10] предлагает использовать SSL в I2P. Эта статья также кратко сравнивает Tor и I2P.

[11] предоставляет набор данных для измерения анонимности сетей на основе луковичной маршрутизации и внедряет новые подходы для количественной оценки уровня анонимности. В нем также обсуждаются детали сетей I2P, Tor и JonDonym.

Что касается прямых сравнений I2P и LLARP, есть только некоторые неформальные (например, [12]). К сожалению, эти сравнения не являются всеобъемлющими и упоминают только основные идеи, стоящие за проектами. [12] в основном фокусируется на подходе, который является альтернативой луковичной маршрутизации.

1.2. Выводы

Текущих статей по данной тематике не достаточно чтобы понять преимущества и недостатки LLARP по сравнению с другими луковичными маршрутизаторами, в частности по сравнению с I2P.

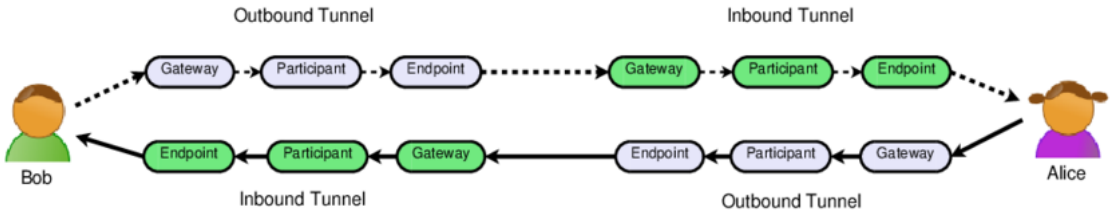
2. Характеристики луковичных маршрутизаторов

Были выделены следующие характеристики:

1. Устройство туннелей
2. Хранение информации о состоянии сети
3. Кто может быть роутером?
4. Кто может хранить метаданные?
5. Поддержка коротких адресов
6. Интерфейс взаимодействия с программами
7. Криптографические примитивы
8. Противодействие Sybil-атакам
9. Балансировка нагрузки
10. Доступ к сети Интернет
11. Транспортный уровень
12. Формат сообщений

Рассмотрим их подробнее.

2.1. Устройство туннелей



Для анонимизации сообщения проходят через несколько роутеров по туннелю. I2P и LLARP в данном случае используют два разных подхода.

В I2P туннели однонаправленны. Это значит что сообщения по ним идут только в одну сторону. Соответственно для коммуникации между клиентом и сервером нужно два туннеля: от клиента к серверу, и наоборот.

В LLARP же туннели двунаправленные: туннель один, сообщения по нему идут как от клиента к серверу, так и обратно.

Данные подходы имеют свои плюсы и минусы.

В односторонних туннелях есть проблема с пониманием что туннель перестал работать. Это может происходить в случае если один из роутеров в туннели был выключен. В таком случае, если это произошло в туннеле от клиента к серверу клиент не может сразу узнать что связь перервана. Единственная возможность это заметить что по туннелю от сервера не приходят подтверждения сообщений, но для этого нужно чтобы истекло какое-то время, порядка нескольких секунд.

Предположительно, в двухсторонних туннелях есть проблемы с безопасностью. Считается, что из-за того что роутеры в туннелях пропускают сообщения в обе стороны,

они могут производить атаки по времени или обмениваться с другими роутерами информацией о следующих/предыдущих роутерах в туннелях. Подробных исследований о том что такой способ менее безопасен чем односторонние туннели, а точнее насколько атаки на двусторонние туннели успешнее чем аналогичные на односторонние туннели, найдено не было.

Итого по туннелям:

- I2P - односторонние. Разделяются на исходящие и входящие. Также разделаются на исследовательские и клиентские
  - Недостаток - трудно заметить когда один из роутеров в туннеле выключился. Как результат частые разрывы соединения после долгого время ожидания ответа.
- LLARP - двусторонние.
  - Проще заметить что нужно поменять туннель.

2.2. Хранение информации о состоянии сети - роутеры

Чтобы строить туннели нужно знать какие роутеры есть в сети.

В I2P для хранения этой информации используется распределённая хэш-таблица (DHT), которая называется netDb.

В LLARP роутеры передают друг другу информацию о роутерах которые они знают. Эта модель называется gossiping.

- I2P - DHT (netDb, основана на Kademlia)
- LLARP - роутеры обмениваются информацией о других роутерах по модели gossiping

2.3. Хранение информации о состоянии сети - пути к сервисам

Чтобы запустить сервер внутри луковичной сети нужно как-то опубликовать информацию о половинах туннелей, которые ведут из/в сервер. Используя эту информацию можно будет строить туннели.

В I2P эта информация раниться в той же netDb. Недавно была добавлена возможность шифровать содержимое этой информации симметричным ключом, чтобы было труднее перечислять все сервера к сети.

В LLARP для хранения этой же информации тоже используется DHT. При этом с самого начала она была зашифрована, но не нужен отдельный симметричный ключ – информация шифруется с использованием публичного ключа закодированного в адресе.

Итого

- I2P - DHT (netDb, основана на Kademlia)
  - Обычно не зашифрована. Недавно добавлена возможность шифровать симметричным ключом.

- LLARP - DHT (тоже Kademlia)
  - Информация зашифрована адресом сервиса и подписана производным ключом.

2.4. Кто может быть роутером?

В луковичных сетях есть политики о том, какие компьютеры становятся роутерами.

В I2P каждый клиент является роутером.

В LLARP клиенты не могут быть роутерами. Для того чтобы стать роутером нужно запустить сервер в специальном режиме. Более того, в официальной сети за оперирование роутера выдаётся награда в криптовалюте. Но оперирование роутером также требует вложений: для того чтобы роутер участвовал в сети он должен заморозить часть средств в криптовалюте. Это помогает противодействовать Sybil-атакам.

Итого

- I2P - каждый клиент (с возможностью отказаться).
  - Преимущество: демократизация.
  - Недостаток: относительная простота Sybil-атаки.
- LLARP - выделенные ноды, получающие награду в Oxen и проверяемые на надёжность.
  - Преимущество: труднее проводить Sybil-атаки.

2.5. Кто может хранить метаданные?

Также важно где именно хранятся метаданные.

В случае I2P метаданные хранят ноды которые называются floodfills. Нода автоматически становится floodfill если у неё достаточная пропускная способность чтобы справиться с соответствующей нагрузкой. При этом можно принудительно сделать ноду floodfill, или отказатся от этого, всё это можно изменить в настройках. По статистике floodfill становятся примерно 6% роутеров сети.

В LLARP не нужны критерии. Клиенты не хранят никакую информацию, зато все роутеры хранят. Это возможно потому что предполагается, что роутеры достаточно заинтересованы в том чтобы предоставлять достаточно ресурсов для этого.

Итого:

- I2P - так называемые floodfills. Клиенты становятся ими автоматически если у них достаточная пропускная способность. Составляют примерно 6% сети.
- LLARP - все ноды (не клиенты).

2.6. Короткие адреса

В луковичных системах обычна используются адреса, в которых закодирована информация о публичном ключе сервера/клиента. Благодаря этому эти адреса гарантируют что при подключении другая сторона владеет соответствующим ключом, предоставляя всем уникальную проверяемую “личность”. Из-за этого получающиеся адреса – довольно

длинные последовательности случайных букв, например ah9ohxnitmr7nxc1ki6b3aba5qk85mr4s794io5dtwxhit6kxato.loki. Людям удобнее использовать короткие запоминающиеся адреса. Данная проблема называется “Треугольник Зуко” [13], и довольно активно изучается.

I2P использует подход с локальной адресной книгой. Пользователь может добавлять в неё имена вручную, или может подписаться на чужие адресные книги. При этом нарушается требование безопасности: адресная книга на которую подписан человек может содержать вредоносные записи, которые отличаются от записей в других книгах. Таким образом нет возможности глобально договориться у кого какой адрес.

В LLARP для решения этих проблем используется блокчейн. Имена можно покупать на нём за специальную валюту Oxen. Считается что блокчейн решает трилемму треугольника Зуко.

Итого:

- I2P - локальная адресная книга. Встроена в клиент, можно добавлять свои.
  - Преимущества: простота, полный контроль пользователем.
  - Недостатки: изначальные подписки должны быть встроены в клиент. Необходимость следить за подписками. Возможность атак имперсонирования со стороны подписок.
- LLARP - адреса покупаются на блокчейне Oxen.

– Преимущества: делает атаки имперсонирования гораздо дороже.

– Недостатки: отсутствие контроля со стороны пользователя, регистрация домена платная.

2.7. Интерфейс взаимодействия с программами

В данной секции пойдёт речь о взаимодействии между луковичным маршрутизатором и прикладными программами, такими как почтовый клиент, веб браузер, веб сервер. По данной характеристике LLARP и I2P имеют абсолютно разные подходы.

В I2P предполагается, что программы знаю о существовании маршрутизатора, знают что они работают не просто с сетью интернет. При этом для взаимодействия маршрутизатора с прикладными программами используются несколько разных протоколов: BOB, SAM, I2CP, Java Stream Library.

Протокол BOB с недавнего времени не используется.

SAM – довольно простой текстовый протокол, подходящий для использования в программах на разных языках программирования.

I2CP – низкоуровневый протокол, основанный на сериализация объектов Java.

Java Stream Library – библиотека для языка Java, предоставляющая интерфейс, сопоставимый с обычными TCP сокетами.

Все эти протоколы объединяет необходимость встраивания

в прикладное приложение, а так же возможность управлять ключами: то есть управлять тем, какую “личность” видит другой конец соединения.

В LLARP же наоборот, предполагается, что процесс встраивания кода в приложения слишком неудобен, поэтому существующие программы должны работать с маршрутизатором без изменений. Для этого используются два интерфейса. Первый – TUN интерфейс, который для приложения представляется обычным сетевым интерфейсом. Единственное отличие – все адреса на нём приватные и временные. Второй интерфейс – DNS сервер, который умеет отвечать на DNS запросы про внутренние адреса сети (.loki, .snode, а также специальные адреса, например localhost.loki), а так же на Reverse DNS запросы. При каждом запросе к новому DNS адресу маршрутизатор выделяет для этого адреса приватный IP-адрес на TUN интерфейсе, и дальнейшие коммуникации с данным IP-адресом будут перенаправлены через сеть LLARP. Как можно заметить, при этом нет механизмов контроля ключей. И правда, маршрутизатор LLARP в текущий момент создаёт одну “личность” на весь сеанс.

Подведём итоги об интерфейсах взаимодействия с прикладными программами.

- I2P - SAM, I2CP, BOB (до недавнего времени), Stream Library (Java)
  - Преимущества: большой контроль приложения над сессиями и ключами.

  - Недостатки: требуют интеграции в приложение.

- LLARP - TUN+DNS
  - Преимущества: легко интегрируется с практически любыми программами реализующими протоколы поверх IP.
  - Недостатки: приложения не могут управлять сессиями и ключами, к примеру чтобы разделить разные аккаунты.

2.8. Криптографические примитивы

В данной секции пойдёт речь о криптографических примитивах, используемых для шифрования и аутентификации сетевых пакетов, а также для поддержания и проверки “личностей” роутеров, клиентов и серверов.

Проект I2P появился в начале 2000-х, и поэтому некоторые алгоритмы которые используются в нём и сейчас довольно давно известны. А именно, для шифрования в нём используется AES, а для криптографии с открытым ключом – алгоритм ElGamal., для подписей – DSA AES популярен и сейчас (хотя и более популярны совмещённые схемы так называемого AEAD), а вот ElGamal несколько утратил популярность из-за очень медленной скорости операций. DSA также утратил популярность по тем же причинам. На текущий момент DSA был заменён на EdDSA. Также в процессе переход с ElGamal + AES на современные X25519 и ChaCha20/Poly1305.

LLARP же появлися в 2018 году, поэтому изначально в нём использовались все современные наработки: XChaCha20/Poly1305 AEAD, HMAC на основе blake2b, подписи ed25519, протокол Диффи-Хеллмана на основе curve25519 и sntrup4591761. В качестве основной библиотеки криптографии используется libsodium.

Остановимся подробнее на использовании sntrup4591761. Это алгоритм шифрования с публичным ключом в текущий момент считается PQ-стойким (т.е. безопасным с учётом текущих знаний о возможностях квантового компьютера). Ключи в данном алгоритме большие, а операции над ними — долгие. Но сами ключи подписаны ed25519, то есть не PQ-безопасными подписями. Таким образом атака со стороны квантового компьютера всё ещё возможна. В данном случае логично либо убрать тяжёлые вычисления, либо во всех частя использовать PQ-безопасные алгоритмы. Данная проблема была принята разработчиками LLARP, но её исправление требует изменения протокола.

Таким образом LLARP в отличие от I2P не использует “старые” алгоритмы, но имеет проблемы с производительностью из-за неуместного использования PQ-безопасных алгоритмов. I2P же понемногу тоже переходит на новые алгоритмы.

2.9. Противодействие Sybil-атакам

- I2P - экспериментальное сканирование пиров на подозрительную активность.
- LLARP - основывается на экономической дороговизне

владения большим количеством нод.

2.10. Балансировка нагрузки

В сети интернет большие сайты используют несколько датацентров для обработки запросов. При этом нужно балансировать трафик: для каждого запроса выбирать на какой датацентр он пойдёт. В случае луковичных сетей для этого нужна отдельная поддержка.

В I2P за путь к серверу отвечает структура под названием LeaseSet. Она хранит пути до сервера. В I2P была добавлена поддержка MetaLeaseSet – специальной структуры, которая объеденяет несколько LeaseSet и предоставляет информацию для выбора одного из них. Но эта функциональность должна быть реализована н

В LLARP нет такой структуры, но реализован другой механизм. В DNS записях. Этот механизм уже используется в сети Интернет. Но у него есть проблемы. Этот механизм должен использоваться клиентским приложением, то есть он не будет работать автоматически. Самым популярным клиентским приложением являются браузеры. Соответственно ожидается что они будут поддерживать эту технологию. Обсуждения поддержки этой технологии среди браузеров ведутся давно, но по итогу этой поддержки нет. Основная причина – дополнительные запросы в DNS приведут к задержкам при загрузке страниц.

Итого:

- I2P - недавно реализован механизм MetaLeaseSet,

позволяющий хостить сайт на большом количестве серверов.

  - Преимущества: прозрачно для программ.
  - Недостатки: требует поддержки роутера.

- LLARP - возможна с помощью SRV записей в DNS.

  - Преимущества: существующий, стандартизированный механизм.
  - Недостатки: некоторые программы, например браузеры, не учитывают SRV записи, так как это требует отдельный DNS запрос.

2.11. Доступ к сети Интернет

При реализации луковичных сетей встаёт вопрос о том, будет ли из сети доступ в обычную сеть Интернет. Данная функциональность обычно реализуется с помощью выходных нод: особых серверов в луковичной сети, которые переправляют пакеты в сеть Интернет.

В I2P есть поддержка данной функциональности, но она требует ручной настройки.

В LLARP также есть поддержка, причём информация о выходных нодах есть в сети. В текущий момент идёт проработка способов поощрения операторов выходных нод.

Итого по доступу в сеть интернет

- I2P - не основная цель. В поставку по умолчанию

включена одна нода, но её пропускная способность ограничена.

- LLARP - есть поддержка, в текущий момент прорабатываются способы поощрения операторов выходных нод. Пока что ноды должны быть добавлены вручную.

2.12. Транспортный уровень

При построении туннеля нужен отдельный протокол, по которому общаются роутеры друг с другом.

В I2P таких протоколов несколько.

NTCP (и его продолжение NTCP2) базируется на протоколе транспортного уровня TCP.

SSU (и его продолжение SSU2) расшифровывается как Secure Semireliable UDP, и базируется на протоколе транспортного уровня UDP. При этом он реализует подтверждение получения сообщений.

В LLARP такой протокол один, он называется IWP, что расшифровывается как Internet Wire Protocol. Он базируется на протоколе транспортного уровня UDP, поддерживает подтверждение получения сообщений, а также умеет разбивать сообщение на несколько частей. Все пакеты шифруются. Основная цель этого – противодействие Deep Packet Inspection, то есть обнаружению того что в пакетах содержится трафик LLARP. Но при этом при обмене сообщениями всё равно можно отличить пакеты LLARP от других.

Оба проекта реализуют свои алгоритмы с нуля и не пользуются какими-то уже существующими решениями.

Итого

- I2P - NTCP(2) (NIO-based TCP), SSU(2) (Secure Semireliable UDP)
  - Оба написаны с нуля.
  - SSU – протокол поверх UDP без перепосылки сообщений.
- LLARP - IWP (Internet Wire Protocol)
  - Неаутентифицированое шифрование для обхода Deep Packet Inspection.
  - Написан с нуля.
  - Поддерживает подтверждение доставки сообщений.
  - Для посылки сообщений размером до 8 KiB разделяет их на блоки по 1 KiB.
  - Нет Congestion Control.

2.13. Формат сообщений

Сообщения в сети несут довольно большое количество метаинформации, например о используемых криптографических ключах, состоянии сет, возможностях серверов. Для организации этой информации в пакетах используются разные форматы.

I2P изначально разрабатывался на языке программирования Java, из-за чего формат сообщений в протоколах основан на сериализации объектов Java. Когда появилась реализация на другом языке, а именно на C++, это оказалось довольно болшой преградой, так как сериализацию нужно было реализовывать вручную. Также это неудобно с точки зрения отладки – сериализация Java это внутренний формат JVM и не предполагалось, что с ним напрямую будут работать люди.

LLARP использует формат, который используется в протоколе BitTorrent, он называется bencode. Это бинарный формат, поэтому он не занимает много места. Но при этом он напрямую может быть преобразован, например, в JSON, из-за чего отладка становится удобнее.

Итого по формату сообщений

- I2P - сериализация Java.
  - Преимущества: удобна когда единственная реализация на Java.
  - Недостатки: не удобно если есть альтернативная реализация на чём-то другом (например C++)
- LLARP - bencode
  - Преимущества: есть библиотеки для разных языков, а также конвертеры в/из JSON.

2.14. Выводы по главе

I2P и LLARP были исследованы с точки зрения 12 характеристик. Были выявлены достоинства и недостатки используемых в них подходов.

3. Реализация изолирования ключей

3.1. Предпосылки

Как было замечено в разделе про интерфейс с клиентскими программами, LLARP использует стандартные средства – сетевой интерфейс и DNS сервер. При этом у клиентских приложений нет возможности выбирать когда используются разные ключи. Опишем ситуации, когда это является проблемой.

3.2. Примеры проблем

Боб пользуется протоколом IRC для общения. Сервер IRC к которому подключается Боб находится в сети LLARP. При этом у Боба есть несколько аккаунтов на этом сервере IRC. К примеру, один из них рабочий, а другой – личный.

Для управления двумя аккаунтами клиентская программа IRC использует два TCP соединения с сервером. По сути, Бобу нужно чтобы эти два соединения не были связаны между собой.

Боб мог бы запустить клиентскую программу на двух разных компьютерах, и тогда они и вправду выступали в качестве двух независимых участников сети LLARP.

Но из-за того, что у Боба запущен один процесс луковичного маршрутизатора, а этот маршрутизатор использует одни и те же ключи для всех соединений, то сервер может очень просто понять что соединения идут с одного компьютера, так как приватные ip адреса у них будут одни и те же.

В данном случае происходит корреляция между двумя аккаунтами пользователя. Но это не единственная проблема.

Проблема может также возникнуть есть пользователь в каком-то случае идентифицирует себя, например при подключении к IRC, а в другом без идентификации читает другие сайты в сети LLARP. При этом если владельцы сайтов совпадают или общаются с владельцами IRC сервера, то они могу сопоставить его активность на сайте с аккаунтом в IRC.

При этом даже не обязательно чтобы все сервера находились на одном компьютере и совпадал приватный IP-адрес. В LLARP встроена возможность получить .loki адрес по IP-адресу сделав Reverse DNS запрос к DNS серверу LLARP.

Изначальная цель этой возможности была в том, чтобы сервер мог создавать соединения к клиенту. Но в данном случае эта функциональность может быть использована чтобы проследить и связать между собой действия пользователя.

3.3. Архитектура решения

Для решения этой проблемы в этой статье предлагается использовать flowlabels – часть протокола IPv6, которая редко используется (а если и используется, то например для выбора маршрутов в датацентрах).

Так как LLARP создаёт TUN-интерфейс и работает напрямую с IP-пакетами, то можно очень удобно получить flowlabel из пакета.

Предлагается хранить внутри LLARP соответствие между flowlabel и ключами, создавая их при необходимости, и таким

образом давать возможность приложениям создавать новые “личности” и переиспользовать старые.

3.4. Преимущества для неадаптированных программ

Тут нужно заметить, что для одного из случаев использования этой функциональности не нужно даже изменять код клиентской программы.

Ядро Linux по-умолчанию присваивает каждому сокету случайный flowlabel.

Это означает, что, например, в случае с IRC клиентом без какого-либо изменения кода этого клиента для двух соединений с IRC сервером будут использованы разные flowlabel, что позволит разграничить “личности”.

3.5. Преимущества для адаптированных программ

Так как создание новых личностей довольно затратно (об этом позднее), имеет смысл иногда их переиспользовать. Например при переподключении к серверу не имеет смысл для того же аккаунта создавать новые ключи, можно поставить в соответствие аккаунту flowlabel и переиспользовать его.

При этом если программа использует flowlabel 0, то ядро автоматически выбирает случайный flowlabel. Чтобы воспользоваться основным набором ключей предоставлена возможность использовать особый flowlabel 1001.

3.6. Преимущества API ядра Linux

API ядра Linux по управлению flowlabel для сокетов предоставляет несколько удобных функций.

Уже упоминалась возможность присваивать случайный flowlabel всем новым соединениям.

Также в Linux существуют различные режимы выдачи flowlabels. При этом ядро гарантирует насколько уникален данный flowlabel: данный flowlabel может быть использован только текущим соединением/в текущем процессе/текущим пользователем.

3.7. Проблемы использования flowlabels

Хотя стандарт IPv6 появился довольно давно (примерно 20 лет назад) API ядра Linux для управления flowlabels мало задокументированно, примеров в интернете мало, а те которые есть – нерабочие. Это связано с тем что данная функциональность протокола IPv6 мало используется

Для успешного использования этого API был изучен исходный код ядра Linux и различных утилит, которые предоставляют возможность указывать flowlabel, например ping и traceroute.

3.8. Проблемы использования IPv6

LLARP выключает использование IPv6 для упрощения реализации выходных нод. Код для поддержки IPv6 был в нерабочем состоянии.

Была добавлена поддержка использования IPv6 вместо IPv4, а также исправлен весь код связанный с IPv6.

3.9. Исследования задержек из-за создания новых ключей

Так как при использовании нового flowlabel обязательно создаётся новое соединение с сервером (а значит новый обмен ключей), то первое соединение с данным flowlabel будет начинаться дольше.

В случае с основным набором ключей данная проблема менее заметна: соединение устанавливается уже в момент когда происходит DNS-запрос адреса сервера.

Было исследовано насколько долго новое соединение создаётся.

Для этого была запущена серия экспериментов, где несколько раз создаётся соединение с заданным flowlabel. При этом интересно отношение времени первого соединения к среднему времени последующих.

В результате экспериментов было замечено, что последующие соединения занимают примерно одинаковое время (маленькое относительное среднее квадратичное отклонение времени), а первый запрос стабильно дольше.

Отношение времени первого обмена сообщениями к среднему времени последующего обмена сообщениями:

- При использовании основных ключей: 0.989 ($\sigma = 0.0463$)
- При создании дополнительных ключей: 2.417 ($\sigma = 0.243$)

С учётом среднего времени на соединение 1.5 секунд это означает, что в среднем установка нового соединения занимает $(2.4 - 1) * 1.5 = 2.1$ секунд.

Также был проведён эксперимент, где последовательно каждый раз создавались новые ключи. Что интересно, в этом случае отношение времени к среднему времени соединения было ещё выше, что говорит о том что краткосрочные соединение создавать невыгодно.

Отношение времени последовательного создания новых ключей к постоянному использованию ключа по-умолчанию составило 5.423, ($\sigma = 1.477$).

3.10. Полученные артефакты разработки

- Патчи к Lokinet которые реализуют изоляцию.
- Клиентская программа на C++, которая демонстрирует использование flowlabels в приложении.
- Серверная программа на Python, которая показывает ключи клиентов которые присоединяются и отправляет ключ обратно клиенту.
- Программа на Python которая демонстрирует ответы от сервера при использовании разных flowlabels, а также подсчитывает время необходимое для сессии с сервером.
- Исследование увеличения задержки создания новых соединений.

3.11. Выводы по главе

Учитывая недостатки LLARP по характеристике “Интерфейс взаимодействия с программами” был разработан новый протокол, который исправляет недостатки, не требует нового канала взаимодействия с программами, удобен, а так же приносит улучшения даже в случае если не меняется код прикладных программ. Было продемонстрировано использование этого протокола. и произведены замеры задержек из-за встраивания нового протокола.

Также на примерах были продемонстрированы проблемы которые этот протокол решает.

4. Реализация использования QUIC в качестве протокола транспортного уровня

4.1. Предпосылки

Текущий транспортный протокол реализован с нуля и не использует state of the art в реализации протоколов используемых для туннелирования (в частности не имеет полноценный Congestion Control).

4.2. QUIC

QUIC [14] – это высокопроизводительный протокол нового поколения. Он реализован поверх UDP и содержит в себе механизмы, аналогичны тем которые содержатся в TCP, но реализованные не в пространстве ядра, а в userspace.

4.3. Проблемы QUIC и их решение

Спецификация QUIC предусматривает перепосылку недоставленных пакетов, что может плохо влиять на поведение пакетов в случае туннелирование другого протокола с перепосылкой сообщений (например TCP)

Поэтому было использовано недавно перешедшее в состояние RFC расширение QUIC под названием “An Unreliable Datagram Extension to QUIC” [15]. Используемая реализация QUIC под названием ngtcp2 его поддерживает.

4.4. Преимущества QUIC

Использование QUIC в качестве транспортного протокола даёт несколько преимуществ.

Основное – это использование передовых технологий подтверждения пакетов и congestion control.

Также важно, что QUIC реализует шифрование, аналогичное TLS.

При этом в отличие от IWP, QUIC активно используется в сети интернет (с протоколом следующего поколения HTTP/3), поэтому он не подвержен блокировке с использованием DPI.

Также важным фактом является то что QUIC поддерживает 0-RTT – технологию быстрого старта нового соединения при условии сохранения метаинформации (ключей и тому подобное) с предыдущего соединения.

4.5. Поддержка нескольких транспортных протоколов в LLARP

Реализация поддержки нескольких транспортных протоколов в LLARP когда-то была начата, но соответствующий код никогда не был интегрирован в основой (и даже не компилировался). Для исходящих соединений вообще не поддерживалось более одного протокола.

Была написана интеграция выбора протокола транспортного уровня. Теперь при настройке интерфейса/IP-адреса/порта можно добавить название протокола. Аналогично можно его указать для настройки исходящих соединений.

Раньше:

```
0.0.0.0=1090
*=30000
```

Теперь:

```
0.0.0.0#iwp=1090
0.0.0.0#quic=1091
*#iwp=30000
*#quic=30001
```

При этом старый файл конфигурации будет продолжать работать: если протокол не указан используется IWP.

4.6. Выводы по главе

Была реализована поддержка QUIC в качестве протокола транспортного уровня в LLARP, что позволяет использовать последние наработки в области шифрования и congestion control, а также позволит использовать соединения без предварительного обмена сообщениями.

5. Заключение

Анонимный обмен сообщениями играет важную роль в демократическом обществе, on используются журналистами, диссидентами и информаторами. Многие системы появились, чтобы достичь его с помощью различных методов, от сети смешанных до последнего анонимного лукового маршрутизатора LLARP. Тем не менее, не хватает информации о методах, которые используют эти инструменты, особенно более новые, такие как LLARP, и компромиссы, которые имеют эти методы.

В этой статье произведено сравнение между анонимными маршрутизаторами LLARP и I2P по 12 характеристикам. Проведён анализ характеристик этих систем и определены их преимущества и недостатки. Наконец, провидены и исследованы улучшения LLARP, которые исправляют некоторые из его недостатков.

Список литературы

[1] Goldschlag David, Reed Michael, Syverson Paul. Onion Routing for Anonymous and Private Internet Connections // Communications of the ACM. — 1999. — 02. — Vol. 42.

[2] I2P - The Invisible Internet Project : Web Technology Report / Media Technology, Leiden University ; Executor: Felipe Astolfi, Jelger Kroese, Jeroen Van Oorschot : 2015.

[3] Chaum David L. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms // Commun. ACM. — 1981. — Vol. 24, no. 2. — P. 84–90. — URL: https://doi.org/10.1145/358549.358563.

[4] Dingledine Roger, Mathewson Nick, Syverson Paul. Tor: The Second-Generation Onion Router // Proceedings of the 13th Conference on USENIX Security Symposium - Volume 13. — SSYM'04. — USA : USENIX Association, 2004. — P. 21.

[5] Ali A., Khan M., Saddique M. et al. TOR vs I2P: A comparative study // 2016 IEEE International Conference on Industrial Technology (ICIT). — 2016. — P. 1748–1751.

[6] Conrad Bernd, Shirazi Fatemeh. A Survey on Tor and I2P // Proceedings of the 9th International Conference on Internet Monitoring and Protection (ICIMP 2014). — 2014.

[7] Jefferys Kee, Harman Simon, Ross Johnathan, McLean Paul. Loki. — 2018. — URL: https://loki.network/wp-content/uploads/2018/10/LokiWhitepaperV3_1.pdf.

[8] zzz (Pseudonym), Schimmer Lars. Peer Profiling and Selection in the I2P Anonymous Network // Proceedings of PET-CON 2009.1. — 2009. — March. — P. 59–70.

[9] Crenshaw Adrian. Darknets and hidden servers: Identifying the true IP/network identity of I2P service hosts // Proceedings of Black Hat 2011. — 2011. — January. — URL: http://www.irongeek.com/i.php?page=security/darknets-i2p-identifying-hidden-servers.

[10] Vashi Dipal, Khilari Girish. Performance Improvement in I2P using SSL // International Journal of Science, Engineering and Technology Research. — 2015. — May. — Vol. 4, no. 5. — P. 1454–1456.

[11] Shahbar Khalid. Analysis of Multilayer-Encryption Anonymity Networks : Ph. D. thesis / Khalid Shahbar. — 2017. — October.

[12] Piotrowska Ania M. VPNs, Tor, I2P — how does Nym compare? — 2020. — URL: https://medium.com/nymtech/vpns-tor-i2p-how-does-nym-compare-8576824617b8.

[13] Wilcox-O'Hearn Zooko. Names: Decentralized, Secure, Human-Meaningful: Choose Two. — 2001. — URL: https://web.archive.org/web/20011020191610/http://zooko.com/distnames.html.

[14] Iyengar J., Thomson M. QUIC IETF Draft 34. — 2021. — URL: https://datatracker.ietf.org/doc/html/draft-ietf-quic-transport-34.

[15] Pauly Tommy, Kinnear Eric, Schinazi David. An Unreliable Datagram Extension to QUIC. — 2022. — URL: https://datatracker.ietf.org/doc/rfc9221/.