Myth I2P

В Сети гуляет несколько популярных мифов о I2P, в которые многие верят. Мы же их развеем.

Миф 1: чем больше участников, тем быстрее работает сеть.

А на самом деле: каждый новый участник должен поддерживать свою базу данных в актуальном состоянии, поэтому сеть, а особенно floodfill’ы просто захлебнутся в потоке таких запросов. В результате часть узлов станет просто недоступной другим узлам.

Миф 2: чем больше доля транзитного трафика, тем выше анонимность.

А на самом деле: I2P оперирует отдельными пакетами, поэтому реальные тоннели поверх обычного интернета, как, например, в VPN, не строятся. Для каждого пакета выбирается подходящий способ доставки, независимо от того, свой ли это пакет или транзитный. Провайдер же видит активность участника как обмен зашифрованными пакетами с различным адресами, выбираемыми достаточно бессистемно. В этом потоке, помимо тоннельных сообщений, присутствуют в большом количестве сообщения, передаваемые напрямую. С другой стороны, узел может видеть часть транзитного трафика, если является концом тоннеля, а не промежуточным узлом, в этом случае извне транзитный тоннель выглядит точно так же, как собственный.

Миф 3: в Tor’е применяется многослойное «луковое» шифрование, а в I2P более прогрессивное «чесночное», в котором сообщение состоит из нескольких «чесночин», предназначенных разным узлам, при этом узел может расшифровать только свою «чесночину», содержимое остальных ему неизвестно.

А на самом деле: изначально оно именно так и планировалось, однако из-за необходимости использования тоннелей парами «исходящий — входящий» пришлось шифровать весь «чеснок» целиком, а не каждую «чесночину» по отдельности. Действительно сообщение, явно именуемое «чесноком», состоит из «чесночин», но поскольку его структура становится видна только после расшифровки, то «чесночины» фактически вырождаются во фрагменты тоннельных сообщений.

Как должно выглядеть реальное «чесночное» шифрование, можно понять из механизма создания тоннелей: сообщение состоит из нескольких записей, из них зашифрованы все, кроме одной, предназначенной данному узлу; он перешифровывает сообщение своим ключом и отсылает дальше. Естественно, следующему узлу предназначается уже другая запись сообщения.

Таким образом, декларируемое «чесночное» шифрование применяется всего лишь в одном сообщении, используемом относительно редко, в основном же потоке данных используется обычное многослойное шифрование: промежуточные узлы шифруют сообщение каждый своим ключом, а владелец расшифровывает, применяя эти ключи последовательно.

Наличие локальной базы данных позволяет участнику выходить в сеть немедленно, не обращаясь к серверам каталогов узлов, как это делается в Tor’е (из-за этого китайское правительство в 2010 году смогло отключить его, блокировав доступ к каталогам). Однако у такой децентрализации есть один существенный недостаток: чтобы получать информацию о новых узлах, в локальной базе данных должны уже присутствовать какие-то узлы. Значит, при первом запуске их придется откуда-то загрузить.

Этот процесс называется «посевом» (reseeding) и заключается в скачивании файлов с небольшого числа жестко прописанных в коде сайтов. Достаточно заблокировать доступ к этим сайтам, и новые узлы не смогут стартовать. Правда, в этом случае для первого запуска можно просто взять список узлов у кого-то другого. Гораздо хуже, если доступ будет не заблокирован, а перенаправлен на сайты с фальшивым списком узлов, — тем самым новый узел рискует попасть в изолированную от остальной сеть, и нет простого способа распознать эту ситуацию.

К чести разработчиков, они понимают масштаб проблемы и работают над тем, чтобы распространять начальный список узлов в виде подписанного их ключом архива по различным каналам.

Сеть I2P состоит из узлов двух видов: маршрутизаторы, имеющие помимо I2P-адресов обычные IP-адреса и видимые в обычном интернете, и узлы, находящиеся позади маршрутизаторов и собственных IP-адресов не имеющие, — они и образуют тот самый «невидимый интернет».

Маршрутизаторы представлены в сетевой базе данных структурой RouterInfo, помимо полного идентификатора содержащей один или несколько внешних IP-адресов и доступных протоколов, а также список возможностей данного маршрутизатора, важнейшей из которых является floodfill. Floodfill-маршрутизаторы служат своего рода «досками объявлений», куда узлы публикуют информацию о себе и куда приходят запросы клиентов.

Во избежание подделки данные подписываются ключом, входящим в адрес. Поскольку информация о маршрутизаторе меняется довольно редко, то соответствующие файлы сохраняются на диске и загружаются в память при старте. У нормально функционирующего I2P-клиента таких файлов должно быть порядка нескольких тысяч.

«Невидимый интернет» представлен структурами данных LeaseSet, содержащих полный идентификатор, дополнительный ключ шифрования и список тоннелей, ведущих к маршрутизатору с данным узлом.

Хотя входящие тоннели имеются и у самих маршрутизаторов, они никогда не формируют LeaseSet’ы: к маршрутизаторам всегда следует обращаться, устанавливая с ними прямые соединения, тоннели же используются только для получения ответов на запросы. Поскольку продолжительность жизни одного тоннеля десять минут, то LeaseSet’ы также существуют недолгое время и поэтому на диске не сохраняются, а при рестарте перезапрашиваются по новой. Тоннели и ключ шифрования из LeaseSet’а являются единственным способом обращения к «невидимому» узлу, то есть, зная адрес, следует сначала запросить его LeaseSet у ближайшего к нему floodfill’а и потом отправить сообщение в один из тоннелей.

Для получения ответа требуется сформировать собственный LeaseSet, который можно отправить вместе с сообщением или же опубликовать на ближайшем floodfill'е. Невозможность установить, на каком маршрутизаторе располагается тот или иной LeaseSet, является краеугольным камнем технологии обеспечения анонимности в сети I2P. Соответственно, большинство атак злоумышленников направлены на решение противоположной задачи. С этой целью в I2P для передачи информации используется сильная криптография, скрывающая данные от особо любопытных провайдеров разных уровней, а удачно применяемые электронные подписи делают сеть устойчивой к атакам типа man-in-the-middle.

Для обеспечения анонимности внутри I2P применяются тоннели, представляющие собой цепочки маршрутизаторов, через которые передаются сообщения. Тоннели бывают исходящие и входящие. Исходящие предназначены для сокрытия местоположения отправителя, а входящие — получателя. Потому LeaseSet'ы и представляют собой список входных узлов и идентификаторов входящих тоннелей, информация об исходящих тоннелях не публикуется.

Местоположение второго конца тоннеля держится в секрете. Для получения ответов клиент посылает серверу собственный LeaseSet. Каким путем проложен тоннель и, соответственно, на каком узле находится его второй конец, известно только создателю тоннеля. Все промежуточные участники тоннеля знают лишь следующий узел, которому следует передать перешифрованное сообщение.

Но это в теории — на практике же промежуточные узлы также знают, откуда пришло сообщение, потому что сообщения между узлами передаются по обычному интернету и узнать IP-адрес отправителя не составляет труда. Далее, при достаточном размере базы можно найти и RouterInfo. Таким образом, если промежуточный узел тоннеля принадлежит злоумышленнику, то он немедленно узнает и двух своих соседей, что компрометирует одно- или двухшаговые тоннели, поскольку позволяет отследить всю цепочку. Теоретически можно увеличить длину тоннелей вплоть до восьми узлов, практически же каждый дополнительный узел резко замедляет скорость работы и надежность, поскольку присутствие узла онлайн на все время существования тоннеля не гарантировано.

Поэтому в настоящий момент в I2P используются трехшаговые тоннели. Таким образом, для успешной деанонимизации узла злоумышленнику следует узнать маршрут любого из тоннелей в любой момент времени — для этого достаточно, чтобы два узла одного тоннеля были доступны злоумышленнику. При нынешнем размере сети в несколько тысяч узлов такой сценарий вполне по силам крупным структурам.

Если в деанонимизации серверов ранее описанный перехват reseeding'а мало поможет, поскольку серверы выбирают узлы входящих тоннелей сами, то для выявления клиентов, посещающих «неблагонадежные» ресурсы, данный метод идеален: все узлы, в том числе выходные, используемые клиентом для построения его исходящих тоннелей, будут априори принадлежать злоумышленнику. Тем самым сразу станет известно, откуда пришло сообщение, предназначенное какому-нибудь входящему тоннелю сервера.

Для обеспечения анонимности с обеих сторон тоннели используются парами: исходящий тоннель отправителя и входящий тоннель получателя. Поскольку тоннели создаются независимо друг от друга, то выходной и входной маршрутизаторы в месте соединения тоннелей видят незашифрованные передаваемые данные.

Поэтому поверх тоннельного используется дополнительный уровень шифрования — специальное «чесночное» сообщение, полностью зашифрованное и предназначенное для конечных узлов в цепочке. Проблема заключается в том, что расшифровкой таких сообщений занимается маршрутизатор узла, а не сам узел.

Таким образом, ключ шифрования, присутствующий в полном идентификаторе, не используется, вместо этого в LeaseSet’е присутствует предназначенный для шифрования отдельный ключ, сгенерированный маршрутизатором, на котором располагается данный LeaseSet. При этом ключ должен быть одним и тем же для всех расположенных на маршрутизаторе узлов, даже если каждый LeaseSet использует свой собственный набор тоннелей. Иначе и нельзя, поскольку «чесночное» сообщение должно быть расшифровано до того, как станет понятно, кому предназначена та или иная «чесночина». В результате изначально здравая идея «чесночной» передачи данных обрела столь уродливую форму при передаче через пару тоннелей.

Таким образом, ключ шифрования, публикуемый в LeaseSet’е, является уникальным идентификатором соответствующего маршрутизатора. Достаточно скомпрометировать любой из узлов, чтобы также скомпрометировать все остальные, в том числе и клиентские. Для проведения данной атаки злоумышленнику следует запустить один или несколько floodfill’ов, куда узлы будут публиковать свои LeaseSet’ы.

Суммируя вышесказанное, приходим выводу: *анонимность I2P в нынешнем состоянии носит лишь базовый характер, позволяя укрыться только от пассивного наблюдения, вроде сбора маркетологической информации. Безусловно, проведение данных типов атак требует серьезных ресурсов, вроде высокоскоростных серверов и специализированного софта, но если кому-то сильно понадобится, то он сможет раскрыть анонимность довольно быстро. Увеличение числа узлов в сети могло бы решить данную проблему, однако при нынешней организации сети это приведет к ее фактическому коллапсу. В то же самое время I2P прекрасно подходит для построения «неубиваемых» ресурсов, доступ к которым невозможно ограничить в принципе.*