Безопасность маршрутизации в беспроводных Ad Hoc сетях. Протоколы SRP, ARAN.

Василий Пряхин

011 группа

Факультет радиотехники и кибернетики

Московский физико-технический институт
(государственный университет)

8 Апреля, 2004

Ad Hoc сети - это множество беспроводных мобильных узлов связи (станций, пользователей), образующих динамическую автономную сеть при помощи полностью мобильной инфраструктуры. Узлы общаются друг с другом без вмешательства централизованных точек доступа или базовых станций, поэтому каждый узел действует и как маршрутизатор, и как конечный пользователь.

Примером может служить соединение нескольких компьютеров беспроводным способом без точки доступа. Нередко такой способ соединения используется на выставках, в конференц-залах.

В Интернете маршрутизаторами в пределах центральных областей сети владеют хорошо известные операторы, и поэтому предполагается некоторая степень доверия к ним. Но это предположение больше не справедливо для Ad Hoc сетей, т.к. ожидается, что все узлы, входящие в сеть, принимают участие в маршрутизации.

И также из-за того, что связи обычно являются беспроводными, любая безопасность, которая могла бы быть получена за счет трудности подключения к сети, потеряна. Более того, по причине того, что топология в таких сетях является чрезвычайно динамичной, традиционные протоколы маршрутизации больше не могут использоваться. Таким образом, Ad Hoc сети имеют намного более *жесткие* задачи и требования для осуществления безопасности, устойчивости и эффективности.

Было предложено несколько протоколов маршрутизации в Ad Hoc сетях, а именно: **AODV**, **DSR**, ZRP, TORA, DSDV, STAR и другие. Но все перечисленные протоколы имеют недостатки и дефекты в системе защиты, и с легкостью могут быть подвергнуты атакам. Цель данной работы заключается в анализе дефектов протоколов маршрутизации в Ad Hoc сетях и обсуждении существующей безопасности протоколов маршрутизации.

Атаки на протоколы маршрутизации в Ad Hoc сетях

Как и в проводных сетях связи, в беспроводных Ad Hoc сетях существует *два вида* атак против протоколов маршрутизации: пассивные атаки и активные атаки.

Пассивные атаки

Пассивные атаки, как правило, подразумевают несанкционированное «подслушивание» пакетов, которые посылают протоколы маршрутизации. В этом случае атакующая сторона не прерывает работу маршрутизирующего протокола, а только пытается узнать ценную информацию, прослушивая трафик маршрутизации.

Главное преимущество атакующей стороны при пассивных атаках заключается в том, что в беспроводной среде связи атаку обычно невозможно обнаружить. И так же сложно защитить от таких атак. Более того, маршрутная информация может раскрыть информацию о взаимодействии между узлами или выявить их адреса. Если маршрут к конкретному узлу сети используется более часто, чем к другим узлам, этот узел может привести к остановке работы всей сети. Другая «интересная» информация, которую можно извлечь из маршрутных данных заключается в расположении узлов. Даже когда было бы невозможно установить точное местоположение узла, можно узнать информацию о сетевой топологии.

Активные атаки

Для осуществления активной атаки недоброжелатель должен уметь проникать в произвольный пакет сети. Цель может заключаться в том, чтобы привлечь (перенаправить) пакеты, предназначенные другим узлам, к атакующей стороне для анализа или просто для нарушения работы сети. Главное отличие по сравнению с пассивными атаками заключается в том, что активные атаки могут быть иногда *обнаружены*. Это делает их менее привлекательными для большинства взломщиков.

Далее мы опишем некоторые типы **активных атак**, которые могут быть легко осуществлены в Ad Hoc сетях.

«Черная дыра»

В этом случае «недружелюбный» узел использует протокол маршрутизации для объявления себя в качестве кратчайшего пути к узлам, чьи пакеты он хочет получить. В протоколе, основанном *на широковещании*, таком как **AODV**, взломщик слушает запросы маршрутов. Когда взломщик получает запрос маршрута к требуемому узлу, он посылает ответ с чрезвычайно коротким маршрутом. Если злополучный ответ достигает узел до того как придет верный ответ, то фальшивый маршрут создан. Если устройство злоумышленника способно встать между общающимися узлами, то оно способно сделать что угодно с пакетами между ними. Оно может решить выкинуть пакет, чтобы организовать dos-атаку или, наоборот, в качестве первого шага заменить его для последующего осуществления атаки типа “man-in-the-middle”.

Переполнение таблицы маршрутизации

При таком способе атаки вредитель пытается создать маршруты к несуществующим узлам. Цель атаки заключается в создании маршрутов, которые бы предотвратили создание новых маршрутов путем переполнения таблицы маршрутизации протокола. «Упреждающие» алгоритмы маршрутизации пытаются узнать маршрутную информацию даже до того, как это необходимо, в то время как «реагирующий» алгоритм создает маршрут, только если он требуется для передачи данных. Получается, что это свойство «упреждающих» алгоритмов делает их уязвимыми при атаках переполнения маршрутных таблиц. Атакующая сторона может просто посылать избыточные маршруты к маршрутизаторам сети. С другой стороны, «реагирующие» алгоритмы, такие как AODV, не собирают заранее маршрутную информацию.

«Испытание бессонницей»

Обычно, атака является практичной только в Ad Hoc сетях, где функционирование устройств критически зависит от работы источников питания. Источники питания пытаются сохранить запасенную энергию, передавая данные, только когда это необходимо. Недоброжелатель может попытаться использовать источники питания (аккумуляторы) с помощью запросов маршрутной информации, или переадресуя ненужные пакеты другим узлам, используя, на пример, атаки типа «черная дыра». Эта атака особенно подходит против устройств, которые не предлагают какие либо службы в сети или предлагают службы только для тех, кто имеет специальное разрешение. Несмотря на свойства этих служб, узлу приходится участвовать в процессе маршрутизации, если он не хочет оказаться отсоединенным от сети.

Обнаружение местоположения

Атаки данного типа пытаются узнать различную информацию о месторасположении узлов или о структуре сети. Полученная информация могла бы дать данные о том, какие узлы примыкают к интересующему узлу или физическое расположение узлов. Атака может быть такой же простой, как и использование эквивалента команды “trace route” в Unix системе. Маршрутные сообщения посылаются с недостаточным значением передела числа хопов, и адреса устройств, посылающих ICMP сообщения об ошибках, записываются. В конце концов, злоумышленник знает, какие узлы расположены на маршруте к интересующему узлу. Если расположение некоторых промежуточных узлов известно, то можно получить информацию также и о местоположении атакуемой мишени.

Протокол SRP (Secure Routing Protocol)

Предпосылки:

1. Между узлом-источником и местом назначения существует безопасная связь (Security Association=SA). Используя SA, стороны, участвующие в обмене информацией, могут проверять друг друга.

2. Источник и приемник владеют секретным ключом $K_{S.T.}$, который выбирается с помощью SA (Security Association).

Основные идеи SRP

Коды идентификации сообщений (MAC) играют важную роль в SRP. Узел *источник S* начинает процесс изучения маршрутов и посылает запросы маршрута с помощью пары идентификаторов: последовательного номера запроса и произвольного идентификатора запроса. Источник, место назначения и уникальные идентификаторы запроса являются входными параметрами MAC наряду с разделяемым ключом $K_{S.T.}$

При получение запроса маршрута, если он является новым, промежуточные узлы добавляют свои IP адреса к пакету запроса маршрута и передают его дальше. Когда пакет запроса маршрута достигает места назначения (пусть это *узел T*), T проверяет запрос. Потом узел T строит ответный пакет и вычисляет MAC-код, включающий содержимое ответа, и возвращает пакет источнику S по обратному маршруту, вычисленному из соответствующего пакета запроса.

Детальное описание протокола

1. Структура сообщения SRP

IP Header	
Basic Routing Protocol Packet	
Type	Reserved
Query Identifier	
Query Sequence Number	
SRP MAC	

SRP запрос включает IP заголовок, основной пакет протокола маршрутизации и SRP заголовок. *Поля SRP* заголовка следующие:

- Поле Типа: длина 1 байт, оно используется, чтобы отличить типы SRP сообщений, таких как сообщение-запрос или сообщение-ответ.
- Последовательный номер запроса (Q_{seq}): 32-х битный последовательный номер, монотонно растет. Он используется для каждого пункта назначения T, чтобы выполнять безопасное общение и защитить устаревшие запросы маршрута. Последовательный номер инициируется во время установления безопасной связи (SA), при этом не разрешается, чтобы он был известен посторонним узлам.
- Идентификатор запроса (Q_{ID}): 32-х битный произвольный номер, который используется промежуточными узлами, для того чтобы распознать каждый исходящий запрос маршрута. Q_{ID} является выходом генератора безопасного псевдо произвольного числа, поэтому его результат статистически неразличим и непредсказуем взломщиком с ограниченными вычислительными ресурсами.

- Код идентификации сообщения (MAC): поле длины 96 бит, которое генерируется снабженным ключами hash-алгоритмом. Алгоритм вычисляет искаженный выход one-way- или hash-функции (т.е. SHA-1 или MD5). Входом one-way функции является целый IP заголовок, пакет запроса основного маршрута протокола и, самое главное, разделяемый ключ $K_{S.T.}$

2. Запрос маршрута

Узел-источник S инициирует поиск маршрута, посылая пакет запроса маршрута, определяемый парой идентификаторов: *последовательным номером* запроса и *произвольным идентификатором* запроса. Источник, приемник и уникальные идентификаторы запроса являются входными параметрами для вычислений кода идентификации сообщения (MAC) вместе с $K_{S,T}$, где $K_{S,T}$ - *секретный ключ*, известный только источнику S и приемнику T.

3. Обработка и распространение запроса

Промежуточные узлы извлекают Q_{ID} и также извлекают адреса источника и места назначения, для того чтобы создать запись в таблице запроса. Запросы, содержащие Q_{ID}, совпадающий с одной из записью для одной и той же пары конечных узлов, уничтожаются. В противном случае, промежуточные узлы ретранслируют запрос маршрута. Промежуточные узлы измеряют частоту полученных запросов от своих соседей, для того чтобы регулировать процесс распространения запросов. С одной стороны, все узлы саморегулируют создание новых запросов маршрута с целью поддерживать избыточность трафика низкой. С другой стороны, узлы-злоумышленники, вероятно, действуют эгоистично и избегают выдержки паузы перед генерированием нового запроса маршрута или генерируют запросы с наивысшей возможной скоростью, потребляя сетевые *ресурсы* и уменьшая *производительность* протокола маршрутизации.

4. Ответ на запрос маршрута

Приемник T в первую очередь проверяет полученный пакет запроса на то, сгенерирован ли пакет именно тем узлом, с которым установлена безопасная сессия (SA). Во-вторых, узел T сравнивает Q_{seq} с S_{max}, который является максимальным последовательным номером запроса, полученного от S, за период жизни SA. Если $Q_{seq}<S_{max}$, то запрос уничтожается как устаревший. Иначе, T вычисляет с помощью ключей hash-функцию поля запроса. Если результат совпадает со значением MAC заголовка SRP, то целостность этого запроса проверена вместе с подлинностью его происхождения. Далее, приемник генерирует несколько ответов на действительные запросы, столько, сколько есть у него соседей, чтобы не позволить возможному узлу-злоумышленнику контролировать несколько ответов. MAC охватывает основной ответ запроса маршрута и оставшийся заголовок SRP, защищает целостность ответа на его пути к источнику и дает доказательство источнику S о том, что запрос на самом деле достиг места назначения.

5. Проверка достоверности ответа на запрос маршрута

Когда узел-источник S получает ответ, S проверяет адреса источника и приемника, Q_{ID} и Q_{seq} и уничтожает ответ, если он не соответствует текущему ожидаемому запросу. Иначе, он сравнивает маршрут ответа от источника с обратным маршрутом внутри пакета ответа, а также поля SRP заголовка и $K_{s.t.}$ В случае успешной проверки, S уверен в том, что запрос и ответ не были испорчены во время передачи. Таким образом, информация является подлинной.

6. Ответы промежуточных узлов

Узел-злоумышленник может подделать пакеты данных или ответы на запрос маршрута. Когда такие маршруты используются или посылаются как ответы,

«неподозревающие» узлы записывают такие недействительные маршруты и могут использовать их в будущем. Поэтому для требуемой устойчивости к атакам, запись маршрута обычно не используется, и промежуточные узлы не обязаны отвечать на запросы маршрута. Если промежуточный узел N имеет активный маршрут к приемнику T, и между источником S и приемником N существует безопасная сессия (SA), то приемник N может сгенерировать ответ. И это единственная ситуация, при которой запрос маршрута не достигает места назначения.

Маршрутизация с идентификацией в Ad Hoc сетях Протокол ARAN (Authenticated Routing for Adhoc Networks)

Протокол **ARAN** использует идентификацию, целостность сообщения и безотказность в среде ad hoc сетей. Структура безопасности в этом протоколе предназначена для управления специальными событиями, общими для ad hoc сетей: быстро меняющейся топологией, маломощным оборудованием и гетерогенными требованиями сетевых приложений. Соответственно, ARAN состоит из двух отдельных стадий. Первая стадия является простой и требует мало дополнительных вычислений от узлов, в отличие от традиционных протоколов в ad hoc сетях. Узлы, которые выполняют необязательную вторую стадию, увеличивают безопасность передаваемых маршрутов, но требуют при этом дополнительных затрат (например, ресурсов источников питания). ARAN использует *криптографические сертификаты* для идентификации и безотказности. Они состоят из процесса предварительной сертификации, обязательной стадии идентификации конечных узлов и необязательной второй стадии, которая обеспечивает безопасные кратчайшие пути. Необязательная стадия значительно более дорогая, чем идентификация конечных передающих узлов.

Ниже в таблице приведены используемые переменные и условные обозначения:

K_{A+}	Открытый ключ узла A
K_{A-}	Закрытый ключ узла A
$\{d\}K_{A+}$	Зашифрованные данные d с помощью ключа K_{A+}
$cert_A$	Сертификат, принадлежащий узлу A
t	Время, отмеченное узлом A
e	Время окончания сертификата
N_a	Номер, отмеченный узлом A (a nonce)
IP_A	IP адрес узла A
RDP	Идентификатор пакета запроса маршрута
REP	Идентификатор ответного пакета
SPC	Идентификатор пакета подтверждения кратчайшего пути
RSP	Идентификатор пакета записанного кратчайшего пути
ERP	Идентификатор ошибочного пакета

Описание протокола

Протокол ARAN состоит из 12 шагов:

1) T -> A:$cert_A = [IP_A , K_{A+} , t, e]K_{T-}$
2) A -> broadcast: $[RDP, IP_X , cert_A , N_A , t]K_{A-}$
3) B -> broadcast: $[[RDP, IP_X , cert_A , N_A, t] K_{A-}]K_{B-} , cert_B$

4) C -> broadcast: [[RDP, IP_X , $cert_A$, N_A , t]K_{A-}]K_{C-} , $cert_C$
5) X -> D: [REP, IP_a ,$cert_x$, N_A , t]K_{X-}
6) D -> C: [[REP, IP_a , $cert_x$, N_A , t]K_{X-}]K_{D-} , $cert_c$
7) C -> B: [[REP, IP_a , $cert_x$, N_A , t]K_{X-}]K_{C-} , $cert_c$
8) A -> broadcast: SPC, IPX , $cert_x$,{[IP_X , $cert_A$, N_A , t]K_{A-}}K_{X+}
9) B -> broadcast: IP_X , $cert_x$, SPC, IP_X , $cert_x$,{[{[IP_X , $cert_A$, N_A , t]K_{A-} }K_{X+}]K_{B-} , $cert_B$}K_{X+}
10) X -> D: [RSP, IP_A , $cert_x$, N_A , route]K_{X-}
11) B-> C: [ERR, IP_A , IP_X , $cert_c$, N_b , t]K_B
12) T -> broadcast: [revoke, $cert_r$]K_{T-}

Во время Шага 1 ARAN требует использования *доверенного сервера сертификатов* T. Перед входом в ad hoc сеть каждый узел запрашивает сертификат от T.

Первая стадия

Шаги 1 – 7 составляют первую стадию протокола ARAN, которая заключается в идентификации конечных узлов. Цель этой стадии заключается в том, что источник проверяет, является ли предполагаемый приемник достижимым.

Шаг 2: Узел-источник A начинает процесс установления маршрута к месту назначения X путем широковещания своим соседям пакета изучения маршрута (RDP). Каждый узел записывает соседа, от которого было получено сообщение.

Шаг 3: Потом узел перенаправляет сообщение каждому из своих соседей, записывая содержимое сообщения. Такая запись предотвращает атаки имитации соединения.

Шаг 4: После получения широковещательного пакета сообщения, узел C (сосед B) подтверждает подпись с данным сертификатом.

Шаг-5: Потом узел C снова посылает широковещательный пакет RDP своим соседям, удаляя при этом подпись узла B. В конце концов сообщение достигает места назначения – узла X, который отвечает на первый пришедший от источника RDP. Нет гарантии того, что первый полученный RDP прошел по кратчайшему пути от источника. RDP пакет, который идет по кратчайшему пути, не будет получен первым, только если он попадет в затор. В этом случае, тем не менее, не перегруженный и не кратчайший путь, вероятно, является предпочтительным перегруженному кратчайшему пути из-за меньшей задержки. Приемник посылает обратный ответ - REP-пакет назад, вдоль обратного пути к источнику. Пусть первый узел, который получает RDP посланный X-ом, есть узел D.

Шаг-6: Узлы, которые получают REP, пересылают этот пакет обратно предшественнику, от которого был получен первоначальный RDP. Все REP подписываются посылающим узлом. Допустим, следующим хопом узла D к источнику является узел C.

Шаг-7: Узел C проверяет подпись узла D, удаляет подпись и потом подписывает содержимое сообщения перед дальнейшей отправкой RDP пакета узлу B.

Вторая стадия

Шаг-8: Необязательная вторая стадия протокола обеспечивает кратчайший путь, но является *дорогостоящей*. Источник сначала должен выполнить первую стадию протокола ARAN. Тем не менее, вариант маршрута уже есть в конце первой стадии, и передачу данных можно осуществлять *одновременно* со второй стадией. Источник, используя широковещание, начинает слать сообщения подтверждения кратчайшего пути (SPC-пакеты) своим соседям (используются те же переменные, что и в первой стадии).

Шаг-9: Соседний узел, который получает сообщение, например узел B, также широковещанием посылает пакет дальше после того, как включает в него свои

собственные криптографические данные. Узел B подписывает зашифрованную часть данных полученного SPC, включает свой собственный сертификат и перезашифровывает с открытым ключом узла X.

Шаг-10: Когда приемник X получает SPC, он проверяет, все ли подписи действительны. Узел X отвечает на первый полученный SPC и также на любой SPC с кратчайшим записанным путем. Потом узел X посылает записанный кратчайший путь в виде RSP сообщения источнику, используя предшествующий узел D.

Шаг-11: ARAN является протоколом *по востребованию*. Узлы продолжают следить до тех пор, пока маршруты являются активными. Когда трафика нет на существующем маршруте в течение времени жизни маршрута, маршрут просто деактивируется в таблице маршрутизации. Данные, полученные на неактивном маршруте, заставляют узлы генерировать сообщения об ошибках (ERR-сообщения), которые посылаются источнику по обратному пути. Узлы также используют ERR-сообщения для того, чтобы сообщить о связях в активных маршрутах, которые были разорваны вследствие перемещения узла. Все ERR-сообщения должны быть подписаны. Для маршрута между источником A и приемником X узел B генерирует ERR-сообщение для своего соседа C.

Шаг-12: В случае, если требуется аннулировать сертификат, сервер T доверенных сертификатов посылает широковещательное сообщение ad hoc группе, которое объявляет об аннулировании.

Недостатки

Протокол использует криптографию с *открытым ключом*. Это слишком дорого, особенно для Ad Hoc сетей, у которых мало ресурсов. Распределение открытого ключа является проблемой, потому что в Ad-Hoc сетях все узлы перемещаются. Как выбирать доверенный источник сертификатов CA остается неясным.

Поэтому был разработан обобщенный протокол SAR (Security-Aware Ad-Hoc Routing) для количественного измерения безопасности исследования маршрутов и распространения с «уровнями доверия» и свойствами безопасности в качестве метрики. Этот протокол работает поверх AODV протокола.

Заключение

Маршрутизация является основой сетевой инфраструктуры. Она контролирует и управляет потоком сообщений в сети. Чтобы установить и поддерживать усовершенствованную сетевую топологию, маршрутизаторы обмениваются сообщениями о состоянии соединения, его стоимости и метрике. Возможно, что злоумышленник может подслушать, перехватить или изменить эти сообщения и даже внедрить вредные сообщения в процессе маршрутизации. Злоумышленник также может овладеть маршрутизатором и полностью контролировать его, для того чтобы повредить сеть изнутри. Этот вид внутренней атаки наиболее разрушительный. В результате компрометация сетевой инфраструктуры может привести к отказу служб, раскрытию или модификации чувствительной маршрутной информации, раскрытию сетевого трафика или ошибоночному использованию сетевых ресурсов.

Общий метод борьбы против этих атак это *цифровая подпись*: узел, порождающий маршрутное сообщение, подписывает сообщение, а приемник проверяет подпись. Таким образом, может быть защищена подлинность и целостность сообщения. Для беспроводных Ad Hoc сетей недостаток поддержки фиксированной инфраструктуры и частые изменения в сетевой топологии делают проблемы безопасной маршрутизации более сложными. В Ad Hoc сетях существует нехватка ресурсов питания. Криптография на основе открытых ключей является слишком дорогой. Но проблема заключается в том, как безопасная связь SA между источником

и приемником может быть установлена перед тем, как будет проложен маршрут между ними.

Протокол OADV был расширен с помощью добавления двух атрибутов безопасности RREQ и RREP и добавления уровня доверия каждому узлу.

Еще была предложена схема, основанная на DSR, в которой каждому узлу приписывается оценка стоимости, и работают методы «сторожевого пса» и «путевого ворчуна», которые управляют работой других узлов и выбирают маршрут.

Описанные протоколы маршрутизации SRP, AODV, ARAN реализованы производителями оборудования для беспроводных ad hoc сетей (JOLT Ltd, Surplus Communications, TTI Wireless и другие) и используются в современных сетях связи. Инфраструктура маршрутизации остается важным компонентом сети и имеет слабые места в системе защиты. Поэтому требуются новые методы криптографии для обеспечения безопасной работы.

При подготовке эссе использовались следующие материалы:

- **“Secure routing in mobile wireless ad hoc networks”**
  Siddhartha Gupte, Mukesh Singhal
  4 Department of Computer Science, University of Kentucky, Lexington, KY 40508, USA
  24 June 2003
  http://cs.engr.uky.edu/~singhal/CS685-papers/adhoc-net.pdf

- **“Routing Data Authentication in Wireless Ad Hoc Networks”**
  Mark Torgerson, Cryptography and Information Systems Surety Department
  Brian Van Leeuwen, Networked Systems Survivability and Assurance
  SAND2001-3119, Unlimited Release, Printed October 2001
  www.cs.tcd.ie/~htewari/papers/sandia01.pdf

- **“Secure Routing in Wired Networks and Wireless Ad Hoc Networks**”
  Huaizhi Li, Zhenliu Chen, Xiangyang Qin, Chengdong Li, Hui Tan
  Department of Computer Science, University of Kentucky, April, 2002
  http://cs.engr.uky.edu/~singhal/term-papers/routing.pdf