Перехват сеанса. Атака Man-In-The-Middle-подделка DNS
Литвиненко Д. С.

Литвиненко Денис Сергеевич / Litvinenko Denis Sergeyevich – студент,
кафедра информационной безопасности, факультет кибернетики,
Федеральное государственное бюджетное образовательное учреждение высшего образования
Московский технологический университет радиотехники, электроники и автоматики, г. Москва

Аннотация: *статья раскрывает содержание понятий перехват сеанса, атаки типа Man-in-the-Middle–подделка DNS. Раскрывается принцип работы некоторых вариантов подделки DNS, а также рассмотрена защита от нее - система DNSSEC и принцип ее действия.*
Ключевые слова: *DNS – Spoofing, DNS Cache Poisoning, DNS ID Spoofing.*

Термином «перехват сеанса» очень часто бросаются применительно к множеству различных атак. В целом, любая атака, включающая в себя использование в той или иной мере сеанса между устройствами, является атакой с перехватом сеанса. Когда речь идет о сеансе, имеется в виду некоторое соединение между устройствами, происходящее в данный момент. То есть, происходит взаимодействие, в рамках которого формально соединение устанавливается, поддерживается, причем для завершения соединения требуется определенный процесс [2, с. 2].

Принцип, лежащий в основе многих видов перехвата сеанса, заключается в том, что если вы можете перехватить определенные порции данных при установлении сеанса, эти данные могут использоваться в целях выдачи себя за одну из взаимодействующих сторон, чтобы получить доступ к информации о сеансе.

Одним из превалирующих типов сетевых атак, используемых против отдельных пользователей и больших организаций, является атака man-in-the-middle (MITM). Учитывая активную атаку подслушивания, MITM работает путем создания подключений к машинам жертв и пересылки сообщений между ними. В таких случаях одна жертва считает, что взаимодействует напрямую с другой жертвой, когда в реальности все их взаимодействия проходят через узел, осуществляющий атаку. В конечном счете, атакующий узел может не только перехватывать конфиденциальные данные, но и способен внедрять и манипулировать потоком данных для получения дальнейшего контроля над жертвой.

Я рассмотрю одну из формы MITM атак - подделка DNS или DNS-Spoofing.

DNS-Spoofing

Подделка DNS представляет собой MITM прием, используемый для предоставления ложной DNS информации на компьютер, чтобы при попытке просмотра, например, сайта www.bankofamerica.com по IP адресу XXX.XX.XX.XX, этот компьютер был направлен на поддельный www.bankofamerica.com сайт, расположенный по IP адресу YYY.YY.YY.YY, который взломщик создал для того, чтобы украсть интерактивные банковские учетные данные и информацию учетной записи у ничего не подозревающего пользователя. На самом деле это легко осуществимо. Я рассмотрю то, каким образом это работает и как от этого защититься.

Нормальное DNS взаимодействие

Протокол системы доменных имен (Domain Naming System - DNS), как определено в стандарте RFC 1034/1035, «представляет собой иерархическую распределенную базу данных, которая может обеспечивать высокий уровень доступности сервиса DNS. Эта система служит основой для преобразования имен хостов Internet в адреса, автоматической маршрутизации почты SMTP и реализации других базовых функций Internet. Система DNS была расширена в соответствии с [RFC 2535] для поддержки хранения открытых ключей шифрования в DNS и обеспечения возможности аутентификации данных, полученных через DNS с помощью цифровых подписей (сертификатов)» [3].

DNS считают одним из важнейших протоколов, используемых в интернете. Такая точка зрения обоснована тем, что DNS обеспечивает целостность системы. Когда мы вводим веб-адрес, например, http://www.google.com, в свой обозреватель, DNS запрос посылается на DNS сервер, чтобы определить, к какому IP адресу принадлежит это имя.

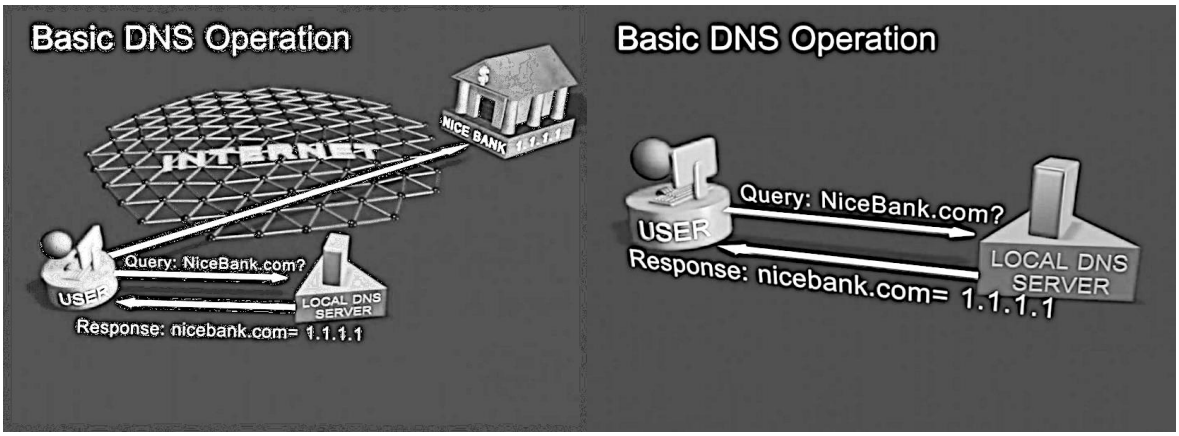

Рис. 1. DNS запрос и ответ

Работа DNS сервера заключается в хранении базы записей IP адресов для сопоставления с DNS именами, передавая эти записи ресурсов клиентам, а также другим DNS серверам. Архитектура DNS серверов в организациях и интернете может быть сложной. Рассмотрим основу DNS взаимодействия, которая показана на рисунке 1.DNS функционирует в формате запрос/ответ. Клиент, желающий преобразовать DNS имя в IP адрес, отправляет запрос на DNS сервер, а сервер отправляет запрошенную информацию в своем ответе. С точки зрения клиентов единственными двумя пакетами являются эти запрос и ответ.

Этот сценарий становится немного сложнее, если рассматривать DNS рекурсию. В силу иерархичной природы DNS структуры интернета, DNS серверам нужна возможность взаимодействовать друг с другом, чтобы обнаруживать ответы для запросов, передаваемых клиентами. В конечном счете, мы можем ожидать от своего внутреннего DNS сервера того, что он знает имя для IP адреса нашего локального сервера, но мы не можем ждать от него того, что он будет знать IP адрес, соотносимый, например, с Google. Именно здесь в игру вступает рекурсия. Рекурсия – это когда один DNS сервер запрашивает информацию у другого DNS сервера от имени клиента, направившего на него запрос. По сути, это превращает такой DNS сервер в клиента, как показано на рисунке 2.

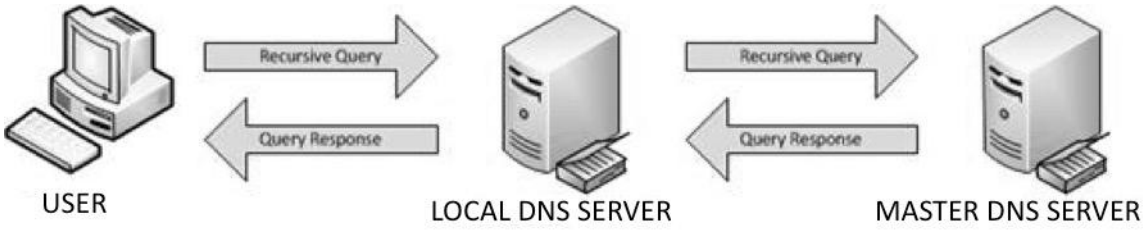

Рис. 2. DNS запрос и ответ с помощью рекурсии

Подделка DNS

Данная атака использует технологию отправки фальшивых ответов на DNS-запросы жертвы.

Атаки на DNS можно условно разделить на две категории.

Первая категория – это атаки на уязвимости в DNS-серверах. С этим подвидом атак связаны следующие опасности:

Во–первых, в результате DNS-атак пользователь рискует не попасть на нужную страницу. При вводе адреса сайта атакованный DNS будет перенаправлять запрос на подставные страницы.

Во–вторых, в результате перехода пользователя на ложный IP–адрес злоумышленник может получить доступ к его личной информации. При этом пользователь даже не будет подозревать, что его информация рассекречена.

Вторая категория – это DDoS-атаки, приводящие к неработоспособности DNS-сервера. При недоступности DNS-сервера пользователь не сможет попасть на нужную ему страницу, так как его браузер не сможет найти IP-адрес, соответствующий введённому адресу сайта.

DNS Cache Poisoning

1. Рассмотрим стандартную DNS Cash Operation.

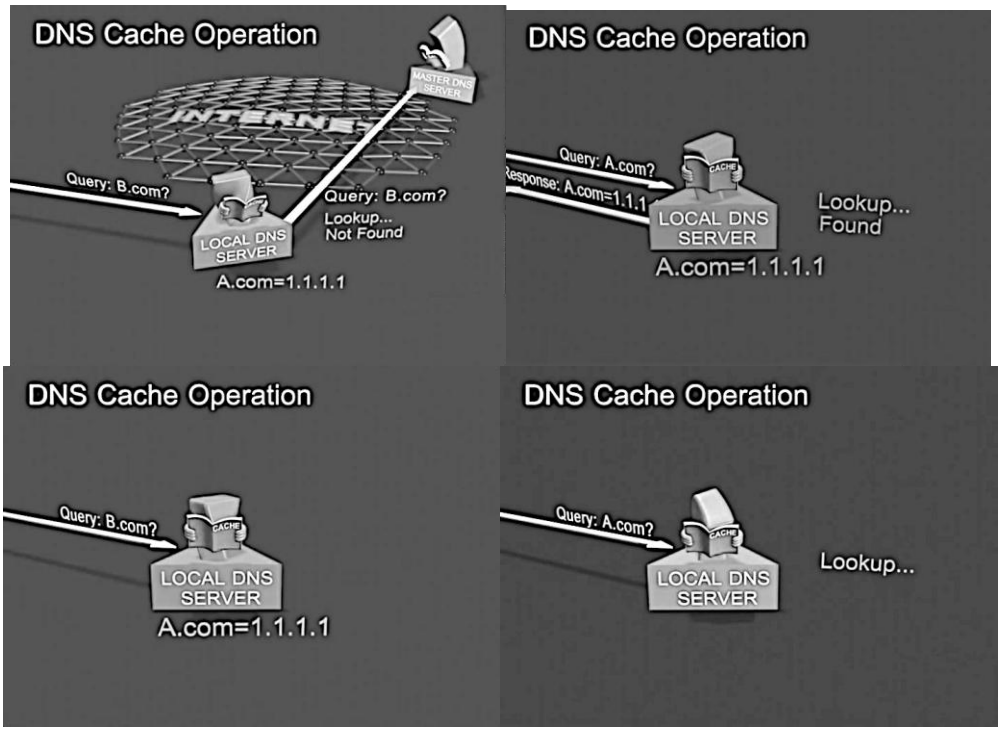

Рис. 3. Запрос на DNS–сервер

Пользователь, желающий преобразовать DNS имя в IP адрес, отправляет запрос на DNS сервер, а сервер отправляет запрошенную информацию в своем ответе, если ее находит.

В противном случае он обращается к другому серверу (Master DNS server).

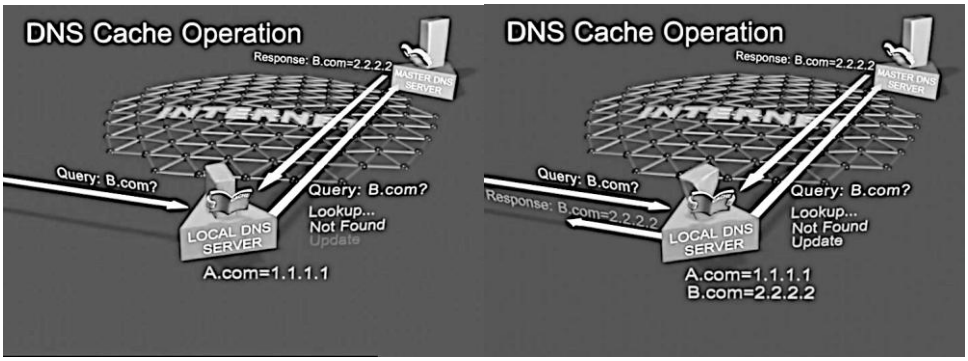

Рис. 4. Обращение за информацией к другому серверу

2. Теперь рассмотрим непосредственно DNS Cache Poisoning

Хакер отправляет запрос локальному серверу на разрешение имени NiceBank.com. (рис. 5).

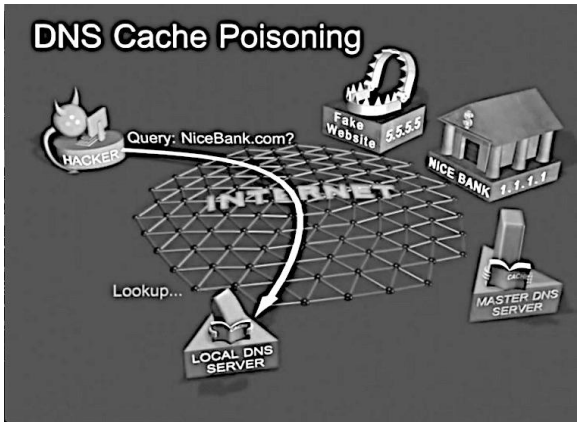

Рис. 5 .Запрос на разрешение имени

Локальный сервер не находит запись и отправляет запрос на другой сервер с соответствующим ID (рис. 6).

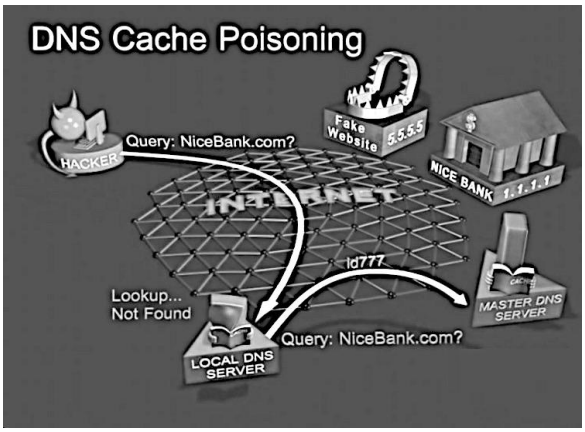

Рис. 6. Запрос другого сервера

Хакеру необходимо ответить до настоящего DNS-сервера. Он шлет группу фальсифицированных DNS-ответов (передавая в качестве IP-адреса адрес злоумышленника -- 5.5.5.5).

Ему надо отправить как можно больше ответов, чтобы увеличить шансы нахождения нужного ID (Рис. 7).

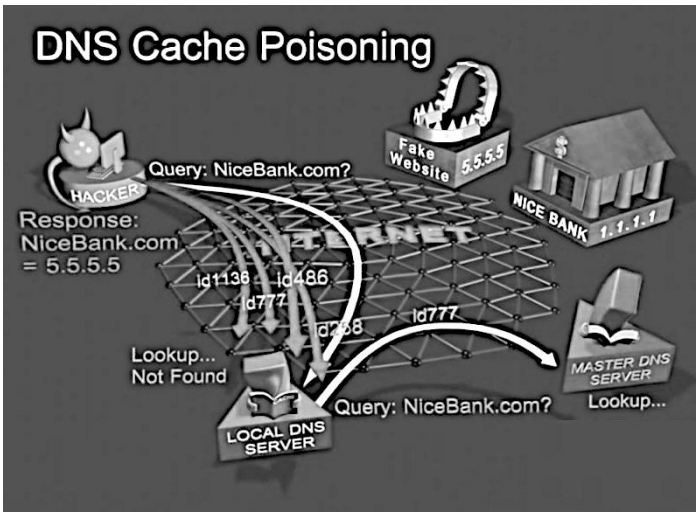

Рис. 7. Нападение

И как только он достигнет своей цели, ответ с настоящего сервера не будет иметь смысла. В локальном сервере прописалась соответствующая информация (Рис. 8).

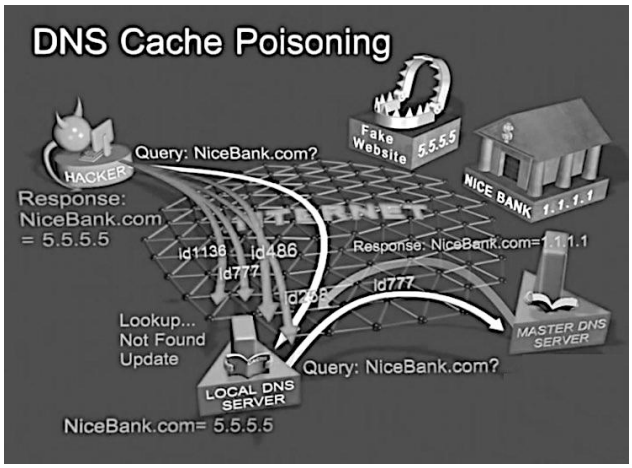

Рис. 8. Достижение цели

Теперь если пользователь будет запрашивать соответствующий адрес (Рис. 9).

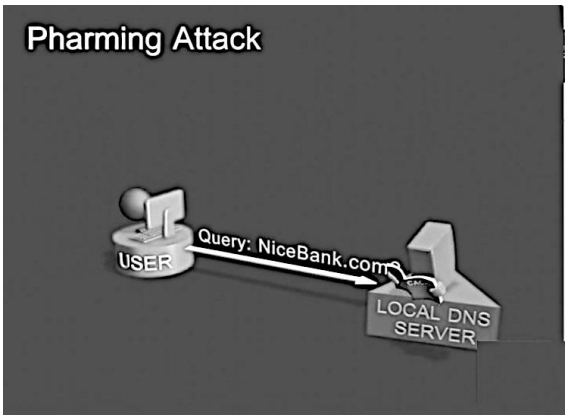

Рис. 9. Запрос адреса

ему будет сообщен адрес машины злоумышленника (Рис. 10).

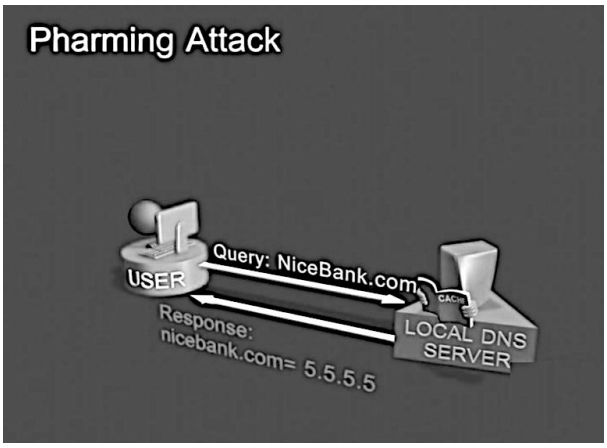

Рис. 10. Адрес злоумышленника

На ней может быть размещена копия настоящего сайта, с помощью которого злоумышленник может красть конфиденциальную информацию (Рис. 11).

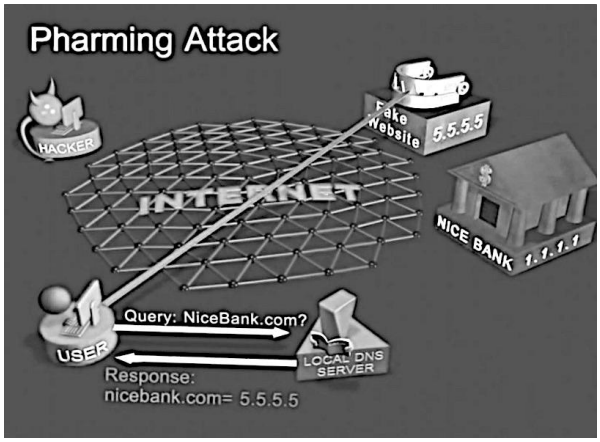

Рис. 11. Совершенная атака

Таким образом, DNS spoofing заключается в подмене DNS информации, содержащейся в ответе на DNS запрос. Целью атаки является изменение записи в кэше целевого DNS сервера, связывающего DNS имя с ложным IP адресом. Для того чтобы провести такую атаку, необходимо каким-то образом предугадать правильный ID запроса. Если атака проводится внутри локальной сети, то не составляет

труда прослушивать трафик. В противном случае все гораздо сложнее. Существует несколько способов:

1. Произвольно проверить все возможные значения 16 битового поля ID (возможных значений составляет 65535)..

2. Послать 200-300 запросов для того чтобы увеличить шансы в нахождении хорошего ID.

В любом случае, необходимо ответить до настоящего DNS-сервера. Этого можно достичь, например, выполнив против сервера атаку типа «отказ в обслуживании» [5].

В случае подмены **DNS ID (DNS ID Spoofing)** операция проходит аналогично, но различается некоторыми моментами.

Пусть у нас есть целевой DNS сервер ns.target.com, www.spoofed.com — DNS имя для подмены (например, имя какого-нибудь банка или организации). У хакера должна быть возможность прослушивать пакеты, идущие от произвольного DNS (пусть это будет , ns.bla.com), для этого он должен контролировать DNS (ns.attacker.com), авторитетный для домена attacker.com. Атака включает в себя следующие шаги:

• Запрашиваем у целевого сервера (ns.target.com) IP адрес (random).bla.com

• Целевой сервер (ns.target.com) отсылает ответ контролируемому DNS (ns.attacker.com), в котором кроме всего прочего есть ID, сгенерированный целевым сервером (ns.target.com)

• Хакер прослушивает ns.attacker.com и узнает ID

• Он подменяет IP адрес www.spoofed.com на подконтрольный. Для этого он посылает DNS запрос на разрешение имени www.spoofed.com ns.target.com. Затем как можно быстрее шлет ложные DNS ответы с измененным на какой-то свой IP адресом www.spoofed.com и IP адресом источника, подмененным на адрес одиного из DNS серверов домена spoofed.com.

Поскольку хакер знает предыдущий ID, он шлет ответы с ID, увеличивающиеся на 1, т. к. возможно сервер ответил на запросы других клиентов. Этот прием значительно увеличивает вероятность угадывания ID.

Таким образом, этот вариант сценария отличается от DNS Cache Poisoning тем, что хакер быстрее угадывает ID, так как знает, от чего отталкиваться.

Есть несколько доступных инструментов, которые можно использовать для осуществления подделки DNS. Одним из таких инструментов является Ettercap [7], который имеет версии под Windows и Linux. Этот инструмент весьма широко используется в различных видах MITM атак.

Ettercap в своей сути представляет анализатор пакетов, использующий ряд подключаемых модулей для выполнения различных атак.

Защита от подделки DNS

От подделки DNS очень сложно защититься в силу того, что атака по своей природе является пассивной. Обычно, мы даже не узнаем, что кто-то пытается подделать наш DNS, пока этого не произойдет. Мы получим веб страницу, отличающуюся от той, которую собирались посетить. В целенаправленных атаках высока вероятность того, что мы никогда не узнаем, что нас обманом заставили ввести свои учетные данные на ложном сайте, пока не получим звонок из банка с вопросом о том, зачем мы только что купили яхту на берегу Греции. Учитывая это, все же есть ряд вещей, которые можно сделать, чтобы защитить себя от таких типов атак:

1. **Можно защитить свои внутренние машины:** такие атаки, как правило, осуществляются во внутренней сети. Если ваши сетевые устройства защищены, то меньше шансов пасть жертвой такой атаки.

2. **Нужно использовать IDS:** системы обнаружения вторжений, если их правильно настроить, могут быстро обнаруживать подделку DNS.

3. **Нужно использовать DNSSEC:** DNSSEC – это более новая альтернатива протоколу DNS, разработанная для защиты от таких атак с помощью цифрового «подписывания», которое позволяет быть уверенным в их достоверности [8].

DNSSEC была разработана для обеспечения безопасности клиентов от фальшивых DNS данных, например, создаваемых DNS cache poisoning. Все ответы от DNSSEC имеют цифровую подпись. При проверке цифровой подписи DNS-клиент проверяет верность и целостность информации.

Рассмотрим принцип действия DNSSEC.

DNSSEC защищает интернет-сообщество от поддельных данных DNS с помощью шифрования с применением открытых ключей, которое используется для цифровых подписей данных уполномоченных серверов зон. Проверка посредством DNSSEC обеспечивает использование данных из указанного источника и предотвращает изменение этих данных при передаче. DNSSEC может определить несуществующие доменные имена.

DNSSEC расширяет возможности защиты DNS, но при этом не является всеобъемлющим решением. Эта служба не защищает от распределенных атак типа «отказ в обслуживании» (DDoS), не обеспечивает конфиденциальность при передаче данных.

В системе DNSSEC каждая зона содержит два ключа: открытый и закрытый. Открытый ключ

общедоступен и отражается в DNS, в то время как доступ к закрытому ключу зоны ограничен. То есть другими словами принцип работы DNSSEC тот же, что и у цифровой подписи. То есть закрытым ключом подписываем, открытым сверяем.

В DNSSEC используется надежная модель доверия, в которой цепочка сертификатов распространяется от вышестоящей зоны в нижестоящую зону. Вышестоящие зоны, т. е. зоны более высокого уровня, подписывают открытые ключи для нижестоящих зон, т. е. зон более низкого уровня.

Когда конечный пользователь пытается получить доступ к веб-сайту, преобразователь в операционной системе пользователя запрашивает запись доменного имени с сервера имен провайдера. Когда сервер запрашивает эти данные, также запрашивается ключ DNSSEC, связанный с этой зоной. С помощью этого ключа сервер может проверить, соответствует ли информация записи на уполномоченном сервере имен.

Для того чтобы использовать преимущества технологии DNSSEC предварительно необходимо подписать зону закрытым ключом шифрования (Рис. 12).

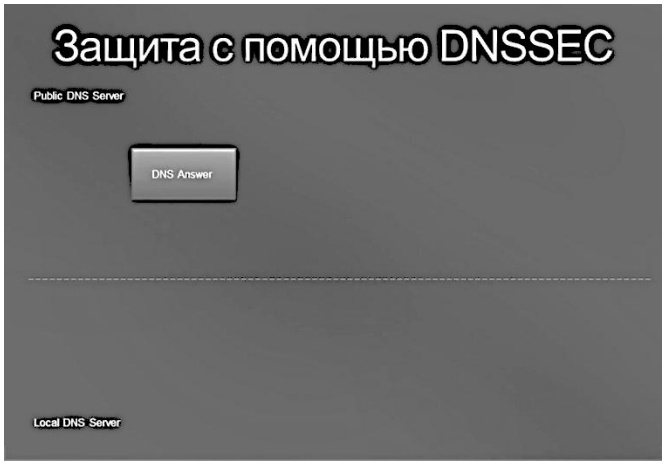

Рис. 12. Подпись зоны

При этом формируется цифровая подпись для каждой ресурсной записи (Рис. 13).

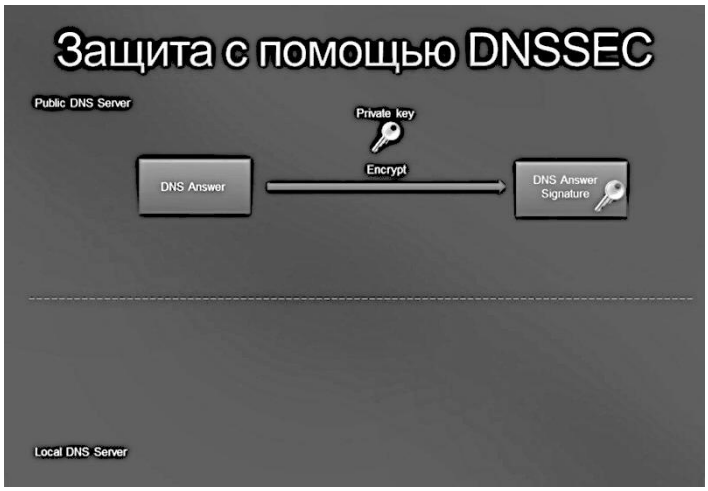

Рис. 13. Цифровая подпись каждой записи

При получении DNS запроса, сервер в открытом виде отправляет ответ, содержащий необходимую ресурсную запись и подпись этой ресурсной записи (Рис. 14).

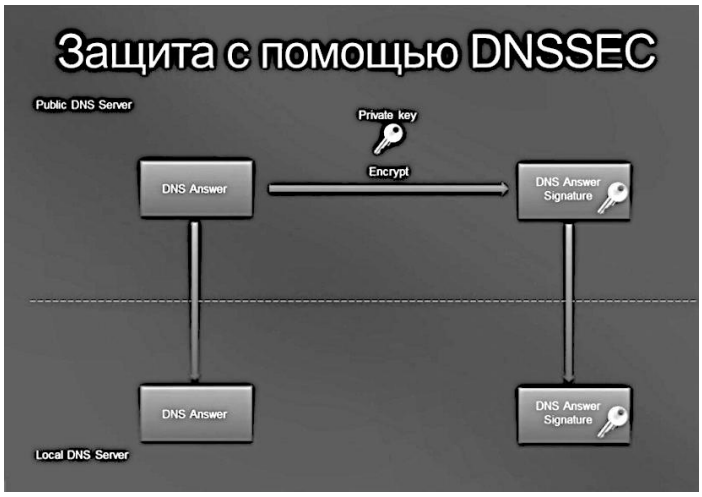

Рис. 14. Отправка ответа и подписи

После получения информации, на приемной стороне выполняется дешифрация подписи с использованием открытого ключа шифрования данной зоны (Рис. 15).

Рис. 15. Дешифрация подписи

Если информация в ответе и в расшифрованной подписи совпадает, то ответ отправляется клиенту (Рис. 16).

Рис. 16. Идентификация

Этот процесс называется проверкой. Если запись адреса была изменена или отправлена из источника, который отличается от указанного, сервер отказывает пользователю в доступе к адресу с мошенническим именем.

Рассмотрим более подробно, с точки зрения техники, принцип работы DNSSEC.

Еще раз напомню, что принцип работы DNSSec тот же, что и у цифровой подписи. То есть закрытым ключом подписываем, открытым сверяем. Особенность состоит в том, что DNSSec использует два типа ключей — одним подписывается зона (ZSK, zone signing key), другим подписывается набор ключей (KSK, key signing key). Сделано это из таких соображений: зона может быть достаточно большой, чтобы удалось подобрать закрытый ключ, поэтому его надо менять почаще, да и сделать его можно покороче, чтобы зоны подписывались быстрее; открытый ключ KSK же используется для небольших объемов данных, поэтому его можно и подлиннее сделать и менять пореже.

Вся информация о защищенном домене в системе DNSSEC определенным образом зашифрована. Закрытый ключ (ZSK) используется для подписи зоны после каждого изменения. Ключ KSK подписывает ключ ZSK, который подписывает DNS-записи, следовательно, для проверки DNS-записи в зоне необходимо знать только открытый ключ KSK. Открытый хэш ключа KSK в форме записи DelegationSigner (DS) передается выше по дереву записей — родительской зоне. Родительская зона подписывает дочернюю запись DS своим закрытым ключом ZSK, который подписывается своим ключом KSK. Это значит, что если DNSSEC полностью приняла ключ KSK для корневой зоны, то она становится частью цепочки проверки для каждого проверяемого доменного имени DNSSEC.

Таким образом, организуется цепочка доверия. Зная открытый ключ администратора родительской зоны, можно проверить «валидность» открытого ключа любой из дочерних зон. Каждый узел в дереве DNS связан с некоторым открытым ключом. Криптографические подписи обеспечивают целостность за счет вычисления криптографического хэша (т. е. уникальной контрольной суммы) данных и, затем, защиты вычисленной величины от несанкционированных изменений посредством ее шифрования. Хэш шифруется с помощью закрытого ключа из пары ключей, чтобы любой желающий мог воспользоваться открытым ключом для его дешифровки. Если дешифрованное получателем значение хэша совпадает с вычисленным, то данные достоверны (не подвергались несанкционированному изменению). Для проверки подписи используется открытый криптографический ключ, который хранится в DNSKEY-записи ресурса. В ходе проверки DNS-сервер извлекает DNSKEY-запись.

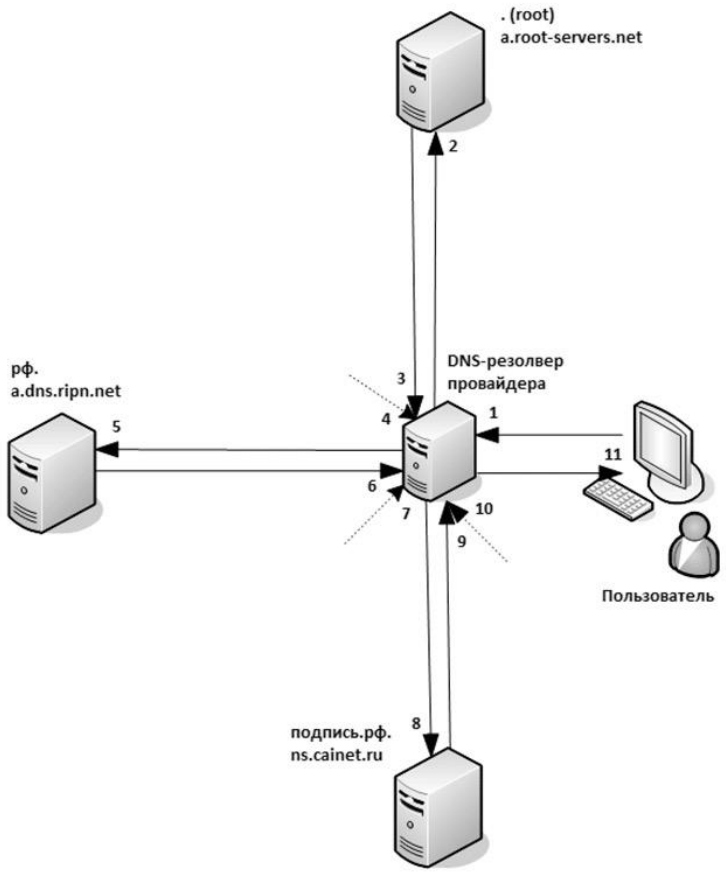

Рис. 17. Пример выполнения

Рассмотрим пример (Рис. 17).

1. 1. Пользователь вводит в строке браузера адрес **подпись.рф**. В соответствии с сетевыми настройками компьютер пользователя посылает запрос на разрешение доменного имени на DNS-сервер (обычно это сервер провайдера доступа к Интернет, который является валидирующим резолвером).

2. 2. DNS-сервер провайдера при отсутствии в кэше информации о запрашиваемом домене посылает запрос к корневому DNS-серверу. В запросе дополнительно устанавливается признак того, что резолвер желает получить ответ с использованием DNSSEC (выставленный бит DO).

3. Корневой DNS-сервер отвечает, что за зону РФ. ответственным является DNS-сервер a.dns.ripn.net. Одновременно он возвращает подписанный с помощью закрытого KSK корневой зоны набор открытых ключей подписи корневой зоны (набор ключей состоит из открытого KSK и открытого ZSK) и подписанный с помощью закрытого ZSK корневой зоны хэш открытого KSK для зоны РФ.

4. Резолвер сравнивает полученный от корневого DNS-сервера открытый KSK корневой зоны с содержащимся в своей памяти открытым KSK корневой зоны. Затем резолвер проверяет содержащуюся в RRSIG электронную подпись набора ключей корневой зоны. Если подпись верная, резолвер начинает доверять ZSK корневой зоны и проверяет с его помощью подпись DS-записи для нижестоящей зоны – РФ. (DS-запись представляет собой хэш KSK зоны РФ).

5. Получив от корневого сервера адрес ответственного за зону РФ. DNS-сервера a.dns.ripn.net, резолвер выполняет к последнему аналогичный запрос – каков IP-адрес сайта **подпись.рф** – и указывает, что использует DNSSEC.

6. Ответственный за зону РФ. DNS-сервер отвечает, что информация о зоне ПОДПИСЬ.РФ. содержится на DNS-сервере cainet.ru; возвращает подписанный с помощью закрытого KSK зоны РФ. набор открытых ключей подписи зоны РФ. и подписанный с помощью закрытого ZSK зоны РФ. хэш открытого KSK зоны ПОДПИСЬ.РФ.

7. Резолвер сравнивает полученную от корневого DNS-сервера DS-запись (хэш) KSK зоны РФ. с рассчитанным самостоятельно хэшем полученного от a.dns.ripn.net открытого KSK зоны РФ. При совпадении хэшей резолвер начинает доверять открытому KSK зоны РФ. и может проверить подпись ключевого набора зоны РФ, а следовательно – доверять открытому ZSK зоны РФ. из ключевого набора и проверить подпись открытого KSK зоны ПОДПИСЬ.РФ. После всех проверок резолвер имеет DS-запись KSK зоны ПОДПИСЬ.РФ и адрес ответственного за зону ПОДПИСЬ.РФ. DNS-сервера cainet.ru.

8. Резолвер обращается к DNS-серверу cainet.ru с запросом адреса сайта **подпись.рф** и указывает, что использует DNSSEC.

9. Сервер cainet.ru знает, что зона ПОДПИСЬ.РФ содержится на нём самом и возвращает резолверу ответ: подписанный с помощью закрытого KSK зоны ПОДПИСЬ.РФ набор открытых ключей подписи зоны ПОДПИСЬ.РФ и подписанный с помощью закрытого ZSK зоны ПОДПИСЬ.РФ адрес сайта **подпись.рф**.

10. Аналогично пункту 7 резолвер проверяет открытый KSK зоны ПОДПИСЬ.РФ, открытый ZSK зоны ПОДПИСЬ.РФ и адрес сайта **подпись.рф**.

11. При удачной проверке в пункте 10 резолвер возвращает пользователю ответ, содержащий в себе адрес сайта **подпись.рф** и подтверждение, что ответ верифицирован (выставленный бит AD).

Заключение

Подделка DNS представляет собой очень опасную форму MITM атаки, если ее инициировать с должным уровнем умений и злоумышленными намерениями. Используя эту методику, мы можем воспользоваться методами фишинга для хищения учетных данных, установки вредоносного ПО или даже для того, чтобы спровоцировать атаки отказа нормальной работы.

Литература

1. Understanding Man-In-The-Middle Attacks. Part2: DNS Spoofing Updated on 7 April 2010. [Electronic resource]. URL: http://www.windowsecurity.com/articles-tutorials/authentication_and_encryption/Understanding-Man-in-the-Middle-Attacks-ARP-Part2.html/ (date of access: 21.09.16).
2. Understanding Man-In-The-Middle Attacks - Part 3: Session Hijacking Updated on 5 May 2010. [Electronic resource]. URL: http://www.windowsecurity.com/articles-tutorials/authentication_and_encryption/Understanding-Man-in-the-Middle-Attacks-ARP-Part3.html/ (date of access: 21.09.16).
3. Detached Domain Name System (DNS) Information. Опубликовано: Март 1999 г. [Электронный ресурс]. Режим доступа: http://rfc.com.ru/rfc2540.htm/ (дата обращения: 21.09.16).
4. Understanding Man-in-the-Middle Attacks – ARP Cache Poisoning (Part 1) Updated on 17 March 2010. [Electronic resource]. URL: http://www.windowsecurity.com/articles-

tutorials/authentication_and_encryption/Understanding-Man-in-the-Middle-Attacks-ARP-Part1.html/ (date of access: 21.09.16).
5. Vulnerabilities of TCP/IP protocols and attacks based on them. Опубликовано: 2004. [Electronic resource]. URL:http://re.mipt.ru/infsec/2004/essay/2004_Vulnerabilities_of_TCP_IP_and_attacks__Andreev.htm/ (date of access: 21.09.16).
6. Внешние атаки. Опубликовано: 22.03.2004. [Электронный ресурс]. URL:http://www.linuxfocus.org/Russian/March2003/article282.shtml/ (дата обращения: 21.09.16).
7. Ettercap. Updated on 27.12.2015. [Electronic resource]. URL: https://kali.tools/?p=830/ (date of access: 21.09.16).
8. DNSSEC – What Is It and Why Is It Important? Updated on 2007. [Electronic resource]. URL: https://www.icann.org/resources/pages/dnssec-qaa-2014-01-29-ru/ (date of access: 21.09.16).
9. Обзор технологии DNSSEC. Опубликовано: 21.12.2011. [Электронный ресурс]. Режим доступа: https://www.techdays.ru/videos/3809.html/ (date of access: 13.10.16).
10. DNSSEC. [Electronic resource]. URL: https://www.verisign.com/ru_RU/domain-names/dnssec/what-is-dnssec/index.xhtml?/ (date of access: 13.10.16).
11. DNSSEC как инструмент обеспечения протокола безопасной работы интернета. [Электронный ресурс]. Режим доступа: https://www.verisign.com/ru_RU/domain-names/dnssec/how-dnssec-works/index.xhtml/ (дата обращения 13.10.16).
12. Принцип работы DNSSEC. [Электронный ресурс]. Режим доступа: https://cctld.ru/ru/domains/dnssec/how/zoom.php/ (дата обращения: 27.10.16).
13. Компьютерные сети / Таненбаум Э. СПб.: Питер, 2003. 4-е издание (7 глава).
14. Компьютерные сети. Принципы, технологии, протоколы / В. Г. Олифер, Н. А. Олифер. СПб: Питер, 2010. 4-е издание (15 глава).