МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ

DOI 10.17223/20710410/10/2

УДК 004.056

АУТЕНТИФИКАЦИЯ В МНОГОУРОВНЕВЫХ СИСТЕМАХ НА ОСНОВЕ ДОВЕРЕННОЙ ПОДПИСИ[1]

П. А. Паутов

Национальный исследовательский Томский государственный университет, г. Томск, Россия

E-mail: __Pavel__@mail.ru

Предлагаются два протокола аутентификации в многоуровневых системах на основе доверенной подписи и их реализация с помощью сертификатов. В одном протоколе доверенная подпись применяется для идентификации только клиента, а в другом — ещё и для проверки подлинности запросов внутри системы. Второй протокол обладает более сильными гарантиями безопасности, вытекающими из невозможности злоумышленником в отсутствие закрытых ключей сторон сформировать аутентичный набор запросов.

Ключевые слова: *многоуровневые системы, аутентификация, доверенная подпись.*

Введение

Рассмотрим систему, состоящую из трёх взаимодействующих подсистем: клиент, внешний сервер, внутренний сервер. Клиент взаимодействует только с внешним сервером, внешний сервер взаимодействует как с клиентом, так и с внутренним сервером (внешний сервер является клиентом внутреннего сервера). Внешний сервер взаимодействует с внутренним только для обработки запросов своих клиентов. Для взаимодействия с внутренним сервером внешний использует фиксированный набор учётных записей, т. е. одна учётная запись внутреннего сервера соответствует нескольким клиентам внешнего сервера.

Для взаимодействия с внутренним сервером используется учётная запись, соответствующая привилегиям клиента внешнего сервера. Например, на внешнем сервере клиенты делятся на группы по привилегиям: «гости», «операторы», «администраторы». Тогда для взаимодействия с внутренним сервером можно использовать три учётных записи, соответствующих группам клиентов. Если внешний сервер сам выбирает учётную запись для взаимодействия с внутренним сервером, то в случае компрометации первого злоумышленник сможет использовать учётную запись с максимальными привилегиями. Возникает задача разработки такой схемы аутентификации, при которой внешний сервер смог бы использовать для взаимодействия с внутренним сервером только ту учётную запись, которая соответствует клиенту, и только тогда, когда клиент взаимодействует с внешним сервером. Например, если клиент, обращающийся

[1]Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П1010).

к внешнему серверу, относится к группе «гости», то внешний сервер должен иметь возможность пройти аутентификацию перед внутренним сервером только от имени учётной записи «гость». Кроме того, внешний сервер не должен иметь возможности пройти аутентификацию перед внутренним сервером от имени учётной записи «гость» без помощи клиента, относящегося к группе «гости».

Искомая схема аутентификации должна удовлетворять следующим двум условиям:

- С1. При взаимодействии с клиентом внешний сервер может пройти аутентификацию перед внутренним сервером только от имени учётной записи, соответствующей данному клиенту.
- С2. Внешний сервер не может пройти аутентификацию перед внутренним сервером от имени какой-либо учётной записи без помощи клиента, соответствующего этой записи.

В работе автора [1] предложено несколько схем аутентификации на основе коммутативного шифрования, удовлетворяющих этим условиям. В данной работе предлагаются два протокола двухуровневой аутентификации на основе доверенной подписи и их реализации с помощью сертификатов. В одном протоколе (описанном в п. 1) используется доверенная подпись без полномочий и осуществляется идентификация только клиента (по его ключу), а в другом (п. 2) — применяется доверенная подпись с полномочиями и, кроме идентификации собственно клиента, производится ещё и проверка подлинности запросов, пересылаемых внешним сервером внутреннему. Второй протокол обладает более сильными гарантиями безопасности, вытекающими из невозможности злоумышленником в отсутствие закрытых ключей клиента и внутреннего сервера сформировать на внешнем сервере аутентичный набор запросов к внутреннему серверу.

1. Протокол на основе доверенной подписи

Механизм доверенных подписей (proxy signatures) позволяет одному пользователю (доверителю) делегировать возможность подписывать сообщения другому пользователю (доверенному подписчику). Доверенная подпись ставится доверенным подписчиком от имени доверителя. Концепция доверенных подписей была представлена в работе [2].

Далее будем использовать следующие обозначения: $PS_{a,b}(m)$ — доверенная подпись под сообщением m, поставленная пользователем b от имени a; C — клиент, F — внешний сервер, B — внутренний сервер; $f(C)$ — учётная запись внутреннего сервера, соответствующая клиенту C. Администратор системы делегирует каждому клиенту системы право подписи от имени соответствующей учётной записи внутреннего сервера. Для делегации права подписи необходимо выполнить некоторый протокол, который определяется выбранной схемой доверенной подписи. Он должен быть выполнен перед началом работы предлагаемого протокола аутентификации, который можно записать в виде следующей схемы, где r — случайное число:

$$\begin{array}{ll} C \to F: & C \\ F \to B: & f(C) \\ F \leftarrow B: & r \\ C \leftarrow F: & r \\ C \to F: & PS_{f(C),C}(r) \\ F \to B: & PS_{f(C),C}(r) \end{array}$$

Внутреннему серверу достаточно убедиться, что подпись поставлена от имени учётной записи $f(C)$. Идентификация доверенного подписчика (клиента) перед внутрен-

ним сервером в данном протоколе не требуется. Условия С1, С2 выполняются благодаря свойствам доверенной подписи. Корректную подпись $PS_{f(C),C}(r)$ может вычислить только клиент C, которому было делегировано право подписи от имени $f(C)$.

Схему доверенной подписи для этого протокола можно реализовать с помощью сертификата. В такой схеме доверитель подписывает сертификат, содержащий информацию, позволяющую идентифицировать доверенного подписчика (например, его открытый ключ). Сертификат отправляется доверенному подписчику. Для того чтобы сформировать доверенную подпись, доверенный подписчик подписывает сообщение своим закрытым ключом и прилагает полученный сертификат. Проверяющий проверяет подпись под сертификатом и подпись под сообщением. Следовательно, для проверки такой доверенной подписи требуется в два раза больше времени, чем при проверке одной обычной подписи. Большинство работ по доверенным подписям посвящены тому, чтобы создать схему, требующую при проверке меньше времени, чем проверка двух обычных подписей. Схемы доверенной подписи, построенные с помощью сертификатов, рассматриваются в работах [3, 4]. Такие схемы представляют интерес, так как строятся на основе проверенных технологий, их проще реализовать, используя готовые решения для обычной схемы цифровой подписи.

Для реализации предложенного протокола аутентификации на основе доверенной подписи с сертификатом будем использовать сначала вариант без полномочий, когда в сертификат входит только открытый ключ доверенного подписчика. Каждому клиенту системы и каждой учётной записи внутреннего сервера ставится в соответствие пара ключей (x, y) — соответственно закрытый (для подписания) и открытый (для проверки подписи). На внешнем сервере для каждого клиента хранится значение $(y_C, S_{x_{f(C)}}(y_C))$, где y_C — открытый ключ клиента C; S_k — некоторый алгоритм цифровой подписи на ключе k; $x_{f(C)}$ — закрытый ключ учётной записи внутреннего сервера, соответствующей клиенту C. Протокол аутентификации в этом случае записывается как

$$\begin{aligned} C \to F:&\quad C \\ F \to B:&\quad f(C) \\ F \leftarrow B:&\quad r \\ C \leftarrow F:&\quad r \\ C \to F:&\quad S_{x_C}(r) \\ F \to B:&\quad S_{x_C}(r), y_C, S_{x_{f(C)}}(y_C) \end{aligned}$$

В конце протокола внутренний сервер проверяет подпись $S_{x_C}(r)$ с помощью ключа y_C и подпись $S_{x_{f(C)}}(y_C)$ с помощью ключа $y_{f(C)}$, который хранится на внутреннем сервере. Это обеспечивает выполнение условий С1, С2.

2. Аутентификация с проверкой подлинности запросов

Описанные выше протоколы обеспечивают аутентификацию клиента, и они затрагивают только начало сеанса работы клиента с системой. Если злоумышленник скомпрометирует внешний сервер (получит над ним полный контроль), то после аутентификации клиента злоумышленник сможет подменять запросы, отправляемые на внутренний сервер в рамках данного сеанса работы. Для предотвращения этой угрозы нужно сопоставить запросы, отправляемые клиентом, с запросами, отправляемыми внешним сервером на внутренний, таким образом, чтобы злоумышленник не смог отправить на внутренний сервер запрос, не соответствующий клиентскому.

Будем считать, что взаимодействие между клиентом и внешним сервером, внешним сервером и внутренним происходит с помощью запросов вида $(req, (a_1, ..., a_n))$,

состоящих из имени req и набора аргументов $(a_1, ..., a_n)$. В ответ на любой такой запрос клиента внешний сервер посылает один или несколько запросов внутреннему серверу. Для каждого запроса клиента набор отправляемых на внутренний сервер запросов фиксирован, и в каждом запросе для внутреннего сервера используется подмножество параметров запроса клиента. Иначе говоря, для каждого запроса $Freq$ с n параметрами, посылаемого на внешний сервер, можно записать набор $((Breq_1, (i_1, ..., i_{k_1})), ..., (Breq_m, (i_1, ..., i_{k_m})))$, где $Breq_j$ — запросы, посылаемые на внутренний сервер, а i_j — номера параметров запроса $Freq$. Далее данный набор будем обозначать $g(Freq)$. В рамках такой модели можно сформулировать ещё одно требование безопасности:

С3. При обработке запроса клиента внешний сервер может использовать только определённые для него запросы внутреннего сервера с параметрами, переданными клиентом.

Данное требование можно выполнить, применив схему доверенной подписи с полномочиями. В таких схемах доверенный подписчик может поставить подпись от имени доверителя только под сообщением, которое соответствует полномочиям, делегированным ему доверителем. Доверенную подпись под сообщением m, поставленную пользователем b от имени a в соответствии с набором полномочий w, будем обозначать как $PS_{a,b,w}(m)$.

При добавлении нового клиента системы C администратор должен делегировать ему право подписи от имени $f(C)$ для каждого запроса $Freq$, посылаемого клиентом, использовав пару $(Freq, g(Freq))$ в качестве набора полномочий.

Для краткости изложения протокола будем предполагать, что по любой из подписей $PS_{a,b,w}(m)$ и $S_x(m)$ можно восстановить сообщение m. Тогда работа системы происходит по следующему протоколу, поясняемому ниже:

$$\begin{aligned}
C \to F: &\quad C, PS_{f(C),C,(Freq,g(Freq))}(Freq, A_{1,n}, t) \\
F \to B: &\quad f(C), (Breq_1, A_{1,k_1}), PS_{f(C),C,(Freq,g(Freq))}(Freq, A_{1,n}, t) \\
&\quad \dots \\
F \to B: &\quad f(C), (Breq_m, A_{1,k_m}), PS_{f(C),C,(Freq,g(Freq))}(Freq, A_{1,n}, t)
\end{aligned}$$

В нём клиент посылает внешнему серверу подписанный запрос с набором аргументов $A_{1,n}$ и временно́й меткой t. Внешний сервер посылает серию соответствующих запросов внутреннему. Значения, возвращаемые в качестве результата исполнения запросов, опущены для краткости. Внутренний сервер при обработке каждого запроса проверяет полученную подпись, а также соответствие запроса $(Breq_i, A_{1,k_i})$ описанию $g(Freq)$ и исходному запросу $(Freq, A_{1,n})$. Вместо временной метки можно использовать случайные числа, генерируемые внутренним сервером.

Доверенная подпись с полномочиями может быть реализована посредством сертификатов следующим образом: $PS_{a,b,w}(m) = (S_{x_b}(m), y_b, w, S_{x_a}(y_b, w))$. Рассмотрим протокол аутентификации, использующий данную реализацию. Каждому клиенту системы и каждой учётной записи внутреннего сервера, как обычно, ставится в соответствие пара ключей (x, y) (закрытый и открытый). На внешнем сервере для каждого запроса $Freq$, посылаемого клиентом C на внешний сервер, хранится сертификат $S_{x_{f(C)}}(y_C, Freq, g(Freq))$. Аутентификация в системе происходит по следующему протоколу:

$$
\begin{aligned}
C \to F: &\quad C, S_{x_C}(Freq, A_{1,n}, t)\\
F \to B: &\quad f(C), (Breq_1, A_{1,k_1}), S_{x_C}(Freq, A_{1,n}, t), S_{x_{f(C)}}(y_C, Freq, g(Freq))\\
&\quad \dots\\
F \to B: &\quad f(C), (Breq_m, A_{1,k_m}), S_{x_C}(Freq, A_{1,n}, t), S_{x_{f(C)}}(y_C, Freq, g(Freq))
\end{aligned}
$$

В данной реализации клиент подписывает свои запросы, используя лишь свой закрытый ключ, в то время как в предыдущем варианте (без сертификата) каждый запрос должен быть подписан с использованием соответствующего набора полномочий.

Заключение

В работе предложены протоколы аутентификации в многоуровневой системе, ограничивающие возможности злоумышленника, получившего контроль над внешним сервером системы. Аутентификация в них основывается на доверенной подписи. Использование последней с полномочиями позволяет не только идентифицировать клиента, но и проверить подлинность запросов от его имени с внешнего сервера к внутреннему, что усиливает защиту системы от злоумышленника, скомпрометировавшего внешний сервер. Реализация же доверенной подписи с помощью сертификатов позволяет реализовать протоколы аутентификации исключительно на базе алгоритмов обычной цифровой подписи, что ввиду развитости инфраструктуры цифровых подписей в мире делает предложенные протоколы аутентификации совершенно практическими.

ЛИТЕРАТУРА

1. *Паутов П. А.* Аутентификация в модели доверенной подсистемы на основе коммутативного шифрования // Прикладная дискретная математика. 2010. № 3. С. 90–95.
2. *Mambo M. M., Usuda K., Okamoto E.* Proxy Signatures: Delegation of the Power to Sign Message // IEICE Transaction Fundamentals. 1996. V. E79-A. No. 9. P. 1338–1353.
3. *Boldyreva A., Palacio A., Warinschi B.* Secure Proxy Signature Schemes for Delegation of Signing Rights. 2003. `http://eprint.iacr.org/2003/096.pdf`
4. *Tan Z., Liu Z.* Provably Secure Delegation-by-Certification Proxy Signature Schemes // InfoSecu'04. Proceedings of the 3rd International Conference on Information security. 2004. P. 38–43.