УДК 681.3.067

А. И. Колыбельников

Московский физико-технический институт (государственный университет)
ООО «Код безопасности»

Новый алгоритм формирования списков отозванных сертификатов

В данной статье обсуждается оптимизация размеров файлов списков отозванных сертификатов(СОС), которые используются в инфраструктуре открытых ключей (ИОК). Так же представлен новый алгоритм формирования архивных форматов подписей на основе предлагаемого алгоритма формирования списков отозванных сертификатов.

Ключевые слова: информационная безопасность, ИОК, СОС, электронная подпись, криптография.

A. I. Kolybelnikov

Moscow Institute of Physics and Technology (State University)
Security Code LTD

New algorithm for formation of certificate revocation lists

This paper discusses the optimization of the size of lists of revoked certificates that are used in the public key infrastructure. A new algorithm for formation of lists of revoked certificates and an algorithm based on it for formation of archival electronic signature formats are presented.

Key words: information security, PKI, CRL, electronic signature, cryptography.

1. Введение

За время своего существования инфраструктура открытых ключей (ИОК) стала широко применяться для аутентификации сайтов, пользователей порталов и узлов сети, с ее помощью построены большинство систем документооборота для обеспечения юридической значимости документов. Поэтому ИОК стала одним из объектов атак со стороны хакеров, а также привлекла внимание криптографов, которые стремятся обеспечить ее стабильную работу и защитить от возможных атак. Одним из важных алгоритмов, который используется в ИОК, является алгоритм формирования списков отозванных сертификатов, сам алгоритм и структура списков описаны в документе RFC5280 [1]. Алгоритм формирования списков отозванных сертификатов представляет интерес для исследования его безопасности, так как на его основе построены алгоритмы проверки электронной подписи и аутентификации пользователя по сертификатам. Кроме того, СОС используются при формировании архивных форматов электронной подписи для подтверждения действительности подписи на момент проверки или формирования.

Когда СОС используется в архивных форматах электронной подписи, например XadES-A [2], он является главной причиной роста размера электронной подписи. В случае перехода от подписи формата XMLDsig [2] к архивному формату XadES-A рост объема документа может составить от 2-х до 3-х порядков, в зависимости от того, какое количество сертификатов выпускается одновременно конкретным удостоверяющим центром (УЦ). Ограничение на включение в СОС информации об отзыве просроченных сертификатов приводит к тому что УЦ не хранит архив СОС. В результате, если пользователь хочет проверить подпись сертификат, которой истек, он сможет это сделать только при одном

условии, если сам пользователь сохранил СОС и сертификат УЦ во время их действия. Далее в статье приводятся основные методы работы с СОС при проверке электронной подписи, описываются преимущества и недостатки этих методов, предлагается новый алгоритм формирования СОС, который позволяет уменьшить объем хранимой информации в архивных форматах подписи, дает возможность проверять электронную подпись вне зависимости от срока действия сертификата открытого ключа.

2. Формирование списков отозванных сертификатов согласно рекомендаций RFC5280

Списки отозванных сертификатов (СОС) в соответствии с [1] применяются для того, чтобы установить, был ли сертификат пользователя или удостоверяющего центра отозван в связи с компрометацией ключей. Иногда СОС используют для того, чтобы сформировать список всех выпущенных сертификатов конкретным издателем. Важное свойство СОС — он содержит информацию только о не просроченных сертификатах. Каждый СОС имеет конкретную область применения. Область применения СОС представляет собой совокупность сертификатов, которые могли быть размещены в конкретном СОС. Например, областью применения могли бы быть:

- «Все сертификаты, изданные УЦ X», «все сертификаты УЦ, изданные УЦ X».
- «Все сертификаты, изданные УЦ X, которые были удалены по причинам компрометации ключа и компрометации УЦ».
- Совокупность сертификатов, основанных на любой локальной информации, такой как «все сертификаты, изданные для служащих Национального института стандартов и технологий, работающих в Москве».

Структура СОС, создаваемая в соответствии с Рекомендацией ITU-T X.509 2-й версии, оптимизирована по содержанию и размеру. Кроме заголовка файла все остальные данные являются номерами отозванных сертификатов и информацией о времени, когда был произведен отзыв. Как только срок действия отозванного сертификата истекает, информация о нем удаляется из СОС. В среднем на запись об одном отозванном сертификате приходится около сотни байт. Это значение увеличивается в случае, если СОС содержит мало записей. В итоге в ИОК отдельной страны, если хотя бы один миллион граждан пользуется ЭП, суммарный годовой объем СОС будет около 300–500 Гб. Эти данные подтверждаются размерами СОС в Эстонии и РФ. В первом случае все граждане страны обеспечены ЭП, удостоверяющий центр один; во втором случае ЭП используется в основном юридическими лицами, удостоверяющих центров более 900. В свете изложенных фактов использование СОС, как источника информации об отозванных сертификатах, предъявляет значимые требования к объему долговременного хранилища данной информации, на один год эксплуатации архив СОС потребует не менее 300 Гб постоянной памяти.

2.1. Использование СОС в архивных форматах ЭП

Списки отозванных сертификатов применяются в архивных форматах подписи в качестве документа, выданного УЦ, который удостоверяет, что на момент формирования архивного формата сертификат открытого ключа электронной подписи не был отозван. Ниже, на рис. 1, приведен полный состав архивной подписи. Как можно видеть, в составе архивной подписи находится минимум два списка отозванных сертификатов, первый — для сертификата открытого ключа ЭП, второй — для сертификата открытого ключа ЭП удостоверяющего центра.

Для иллюстрации необходимости оптимизировать СОС следует рассмотреть статистику использования подписи формата CAdES-A в РФ. При среднем объеме подписываемого

сообщения в 50 Кб, подписи для него в 1 Кб и сертификате 1Кб список отозванных сертификатов в среднем будет занимать от 3 Мб до 30 Мб, объем подписи архивного формата составит 7 Мб до 31 Мб. То есть накладные расходы на память для архивного хранения документов могут составлять от 100 до 1000 порядков.

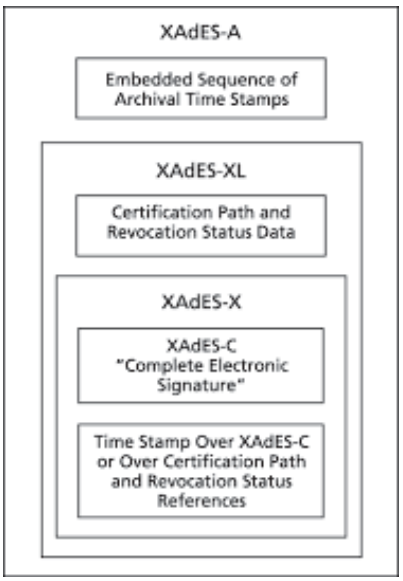

Рис. 1. Формат архивной подписи XAdES-A

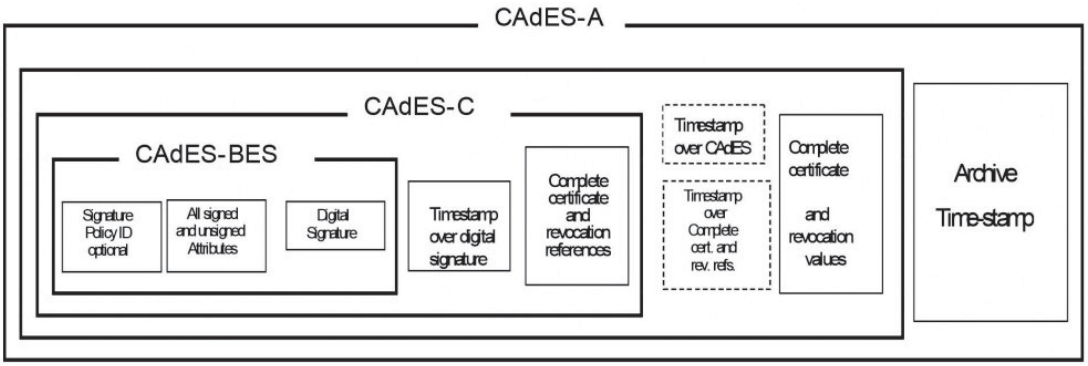

Рис. 2. Формат архивной подписи CAdES-A

2.2. Предлагаемый алгоритм формирования списков отозванных сертификатов

Для того что бы оптимизировать работу со списками отозванных сертификатов, уменьшить накладные расходы на хранение информации, дать возможность всем пользователям ИОК проверять ЭП вне зависимости от срока действия сертификата предлагается изменить алгоритм формирования, порядок хранения и алгоритм проверки СОС. Предлагается возложить на удостоверяющий центр обязанность хранить информацию об отозванных сертификатах с момента открытия УЦ и до его закрытия. При этом необходимо сохранить возможность получать информацию из СОС в объеме всех отозванных сертификатов, не завершивших свой срок действия. Объем СОС не должен содержать более 5% дополнительной информации. Предлагается для каждой записи добавить поле значения хеша. Для первого блока хеш рассчитать от сообщения и метки времени — поля revocationDate Time. Далее вычисленный хеш следует включить отдельным полем в состав СОС и рассчитать значение подписи для этого хеша. Модифицированный список отозванных сертификатов

ведется удостоверяющим центром непрерывно. Данный СОС пользователи могут получить от УЦ в любой момент времени. Причем, для оптимизации объема передаваемых данных, может предоставляться часть СОС. Так, для проверки ЭП с действующим сертификатом достаточно предоставить весь объем записей из СОС, в которых метка времени на сутки больше, чем дата создания сертификата. Общая логическая схема формирования СОС приведена ниже, она описывает общий принцип размещения данных, формирования хеша и электронной подписи.

$$M_0\|T_0\|(M_0\|T_0)_{Hash_0}\|(Hash_0)_{Sign_0},$$
$$M_1\|T_1\|Hash_0\|(M_1\|T_0\|Hash_0)_{Hash_1}\|(Hash_0)_{Sign_1},$$
$$M_2\|T_2\|Hash_1\|(M_2\|T_1\|Hash_1)_{Hash_2}\|(Hash_2)_{Sign_2},$$
$$\dots$$
$$M_n\|T_n\|Hash_{n-1}\|(M_n\|T_n\|Hash_{n-1})_{Hash_n}\|(Hash_n)_{Sign_n}.$$

Общий вид СОС с доработанной схемой формирования в виде соответствующем Рекомендации ITU-T X.509 версии 2 представлен ниже. Жирным шрифом выделены дополнительные поля, которые необходимо добавить, чтобы модифицировать процесс формирования СОС.

CertificateList ::= SEQUENCE {
tbsCertList TBSCertList,
signatureAlgorithm AlgorithmIdentifier,
hashAlgorithm AlgorithmIdentifier,
signatureValue BIT STRING }

TBSCertList ::= SEQUENCE {
version Version OPTIONAL,

signature AlgorithmIdentifier,
issuer Name,
thisUpdate Time,
nextUpdate Time OPTIONAL,
HashUpdateValue BIT STRING
revokedCertificates SEQUENCE OF SEQUENCE {
userCertificate CertificateSerialNumber,
revocationDate Time,
HashPreviosValue BIT STRING
crlEntryExtensions Extensions OPTIONAL

} OPTIONAL,
crlExtensions [0] EXPLICIT Extensions OPTIONAL

}

Поле hashAlgorithm AlgorithmIdentifier определяет тип алгоритма, который используется для вычисления хеша. Поле HashUpdateValue содержит значение хеша типа BIT STRING, вычисленного от самого последнего блока данных об отозванных сертификатах. Значение хеша из данного поля должно использоваться для вычисления электронной подписи, которое помещается в поле signatureValue и имеет тип BIT STRING. Поле HashPreviosValue содержит строку типа BIT STRING, которая является значением ранее

вычисленного хеша от предыдущего информационного блока данных об отозванных сертификатах revokedCertificates. Для первого блока информации revokedCertificates значение поля HashPreviosValue будет равно NULL. Если каждое последующее значения хеша, кроме первого, вычислять от новых данных, новой информации о дате отзыва и от значения хеша предыдущего блока для обеспечения целостности СОС, можно будет использовать электронную подпись, накладываемую на самый последний блок информации, от которого посчитан хеш. Такой механизм вычисления хешей и ЭП от них позволяет подтверждать авторство, целостность и достоверность информации в СОС с момента его создания, вне зависимости от того, истек срок действия отозванного сертификата или нет.

2.3. Алгоритм проверки списков отозванных сертификатов

Предлагается использовать следующий алгоритм проверки СОС. Пользователь запрашивает информацию о СОС у удостоверяющего центра, указав в запросе дату выпуска сертификата открытого ключа подписи, которую он хочет проверить. Удостоверяющий центр предоставляет пользователю информацию в виде набора записей из общего СОС. В результате УЦ отправляет пользователю все записи, относящиеся к периоду от начала действия сертификата открытого ключа ЭП и до текущего состояния СОС, которое обладает действительной и валидной подписью. Проверив подпись под последней записью, пользователь может доверять всем остальным записям, так как они связаны между собой при помощи связанных значений хеша. Пример передаваемого СОС от УЦ приведен ниже:

$$M_0\|T_0\|(M_0\|T_0)Hash_0\|Hash_0,$$

$$M_1\|T_1\|Hash_0\|(M_1\|T_0\|Hash_0)Hash_1\|Hash_0,$$

$$M_2\|T_2\|Hash_1\|(M_2\|T_1\|Hash_1)Hash_2\|Hash_2,$$

$$\dots$$

$$M_n\|T_n\|Hash_{n-1}\|(M_n\|T_n\|Hash_{n-1})Hash_n\|(Hash_n)Sign_n.$$

3. Оптимизация архивного формата электронной подписи

Если использовать предлагаемый выше формат СОС в архивных форматах электронной подписи, можно с дополнительной оптимизацией в качестве СОС достаточно включать не весь СОС, а его текущее состояние, например, в виде

$$(M_n\|T_n\|Hash_{n-1}\|(M_n\|T_n\|Hash_{n-1})Hash_n\|(Hash_n)Sign_n.$$

В этом случае размер СОС для таких форматов ЭП будет оптимизирован на несколько порядков, будет сохраняться связность зафиксированного состояния сертификата на момент проверки с СОС. Кроме того, любая из таких позиций СОС связана при помощи хешей со всеми предыдущими состояниями СОС и с будущими, а также в каждый момент времени подтверждается действительной и валидной подписью УЦ под СОС.

То есть при использовании предлагаемого архивного формата ЭП можно не только обеспечить экономию дискового пространства в электронном архиве от 100 раз от размера сообщения, но и обеспечить действительность результатов проверки или формирования ЭП на протяжении всего срока хранения без каких-либо дополнительных процедур по повторному подписыванию архива.

4. Выводы

Существующий алгоритм формирования СОС является не оптимальным. Использование рекомендаций [1] для формирования СОС приводит к тому, что электронная подпись не является полным аналогом собственноручной подписи, так как электронная подпись утрачивает свою действительность по истечению срока действия сертификата открытого ключа электронной подписи. Применение архивных форматов электронной подписи в качестве средства продления срока действия ЭП также не является эффективным, так как требует затрат на хранение СОС для каждой ЭП, которые на несколько порядков больше самого электронного сообщения и периодического переподписания архива, в случае истечения срока действия сертификата открытого ключа, который применялся для заверения результатов проверки ЭП пользователя. Подход к формированию списков отозванных сертификатов, применяющийся в настоящее время, не оптимален и может быть доработан. Доработки алгоритма формирования СОС, предложенные в этой статье, позволят расширить возможности ИОК в оптимизации затрат на хранение СОС и обеспечит действие ЭП документа на неограниченный срок в полном соответствии со свойствами собственноручной подписи.

Литература

1. RFC 5280. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile https://www.ietf.org/rfc/rfc5280.txt

2. XML Advanced Electronic Signatures (XAdES) https://www.w3.org/TR/XAdES/

3. Cryptographic Message Syntax (CMS) https://tools.ietf.org/html/rfc3852

References

1. RFC 5280. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile https://www.ietf.org/rfc/rfc5280.txt

2. XML Advanced Electronic Signatures (XAdES) https://www.w3.org/TR/XAdES/

3. Cryptographic Message Syntax (CMS) https://tools.ietf.org/html/rfc3852

Поступила в редакцию 29.04.2017