УДК 004.056

МЕХАНИЗМЫ ЗАЩИТЫ ИНФОРМАЦИИ ПРИ ВЫРАБОТКЕ ОБЛАЧНОЙ ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСИ

В.А. ГЕРАСИМОВ[1], М.А. КАЗЛОВСКИЙ[1], О.В. БОЙПРАВ[2]

[1]Научно-производственное республиканское унитарное предприятие «Научно-исследовательский институт технической защиты информации», [2]Белорусский государственный университет информатики и радиоэлектроники, г. Минск, Республика Беларусь

Введение

Электронная цифровая подпись предназначена для контроля целостности и подлинности электронных документов. При выработке подписи используется личный ключ, который находится в распоряжении владельца. Только владелец личного ключа может выработать корректную подпись. Корректность подписи означает, что документ не был изменен и не позволяет владельцу отказаться от авторства подписанного документа [1].

В настоящее время процесс выработки электронной цифровой подписи может быть реализован в формате облачного сервиса. Такой сервис позволит обеспечить удобные и безопасные условия использования электронной цифровой подписи как для физических, так и для юридических лиц. При использовании этого сервиса подписант передает личный ключ в защищенное облачное хранилище. Тем не менее, владелец не теряет контроль над своим ключом, так как выработка подписи происходит только после успешной аутентификации и явной передачи права подписать определенный документ системе документооборота.

Для безопасной выработки облачной электронной цифровой подписи можно предложить систему облачной подписи (далее – СОП), в которую входят [2] следующие компоненты:

– сервер подписи (далее – СП), предназначенный для генерации и хранения личных ключей подписантов, выработки ЭЦП под контролем подписантов от их лица, при этом составным компонентом СП является аппаратное устройство создания подписи (далее - УСП);

– сервер документооборота (далее – СД), предназначенный для создания и проверки электронного документа;

– клиентская программа (далее – КП), предназначенная для взаимодействия подписанта с компонентами системы облачной подписи;

– сервер регистрации (далее – СР), отвечающий за регистрацию и деактивацию аккаунтов подписантов в СОП;

– прикладная система (далее – ПС), внешний компонент по отношению к системе облачной подписи, отвечающий за загрузку, разработку, хранение и отображение подписываемых документов.

Однако, распределения обязанностей между компонентами не может являться гарантией безопасности СОП. Для того чтобы СОП гарантировала корректное использование личного ключа подписанта авторы предлагают использовать определенные механизмы, которые позволят обеспечить прозрачное и безопасное функционирование СОП.

1. Использование JWT при аутентификации пользователей

Для обеспечения надежной передачи данных между компонентами СОП используется JSON Web Token (далее – JWT). Объект JWT состоит из нескольких кодовых слов, разделенных символом «.» (точка), которые описывают компоненты,

представляющие собой объекты в формате JSON. Компоненты сначала кодируются по кодируются правилам UTF8, а затем по правилам base64url [3].

Для обеспечения принципов целостности и подлинности информации используется модифицированный объект JSON Web Signature (далее – JWS), который представляет собой один из типов JWT.

Объект JWS состоит из трех компонентов: заголовок, подписываемые данные и подпись. Предложенная модификация состоит в том, что третья часть JWS представляет собой не подпись, а CMS-структуру с отсоединенными данными, которая помимо подписи содержит дополнительные атрибуты. Это сделано для того, чтобы упростить проверку JWS с помощью входящих в состав CMS-структуры атрибутов.

2. Использование сетевого токена для аутентификации пользователей

Аутентификация подписанта в КП проходит после аутентификации в ПС. Для связи сеанса ПС с КП используется сетевой токен, который генерируется СП в соответствии с требованиями СТБ 34.101.47-2017 «Информационные технологии и безопасность. Алгоритмы генерации псевдослучайных чисел» с применением режима TOTP (A.8), длиной 6 символов.

Сетевой токен относится к фактору «чем я владею». Вместе с сетевым токеном, который подписант получает по каналу GSM, аутентификация проводится и по коду сессии. Для получения кода сессии вычисляется хэш-значение от описанного в пункте 1 модифицированного JWS. Восемь последних шестнадцатеричных символов полученного хэш-значения являются кодом сессии.

3. Использование PIN для аутентификации пользователей

Для подписи документа пользователь должен ввести PIN-код, который позволит получить доступ к личному ключу. PIN-код относиться к фактору «что я знаю». PIN-код должен состоять как минимум из 6 символов. Дополнительно PIN-код используется для парольной аутентификации сторон и формирования общего ключа защищенного соединения между КП и УСП с использованием криптографического протокола BPACE, определенного в СТБ 34.101.66-2014 «Информационные технологии и безопасность. Протоколы формирования общего ключа на основе эллиптических кривых».

4. Использование данных активации подписи для повышения уровня гарантий контроля

При выработке облачной электронной цифровой подписи определяются уровни гарантий контроля (далее – УГК). Для системы облачной подписи определяются два УГК: УГК1 (базовые гарантии) и УГК2 (высокие гарантии).

УГК1 может применяться только тогда, когда компрометация личного ключа приводит к минимальным потерям. Например, этот уровень соответствует ситуации, когда подписываются документы, предназначенные для использования только внутри организации для ее собственных нужд.

УГК2 может применяться для подписи юридически значимых документов. При применении УГК2 является обязательным применение данных активации подписи (далее – ДАП). ДАП являются результатом криптографических преобразований, которые связывают, аутентификацию подписанта, идентификатор личного ключа подписанта и данные для подписи. Формирование ДАП независимо друг от друга выполняют КП и УСП. При совпадении ДАП, переданных от КП в УСП, с ДАП, сформированными в УСП, УСП выполняет выработку значения электронной цифровой подписи.

5. Использование протокола активации подписи для повышения уровня гарантий контроля

Для повышения гарантий контроля при выработке электронной цифровой подписи вводится понятие протокола активации подписи (далее – ПАП). Он представляет собой последовательность действий, которую должны выполнить компоненты СОП для получения электронной цифровой подписи документа.

Основные обозначения, используемые в алгоритме ПАП, представлены в таблице 1.

Таблица 1 – Список основных обозначений алгоритма ПАП

Обозначение	**Пояснение**
$A \rightarrow B$	Обмен информацией между сторонами от А к В
JW$S-M$	Передача данных в формате JSON Web Signature Mode
JSON$(data1, data2, \dots, dataN)$	Передача данных в формате JavaScript Object Notation
h$(data)$	Хэш-значения данных, выработанные по алгоритму из СТБ 34.101.31-2020 (пункт 7.8)
$val1 \| \| val2 \| \| \dots \| \| valn$	Конкатенация значений
$display(Source$: $data1, data2, \dots, dataN)$	Отображение в клиенте определенных данных Пример: $display$(КП: $CodeSession, OTP$), $display$(ПС: $hDoc, IdSession$)
input(data1,data2,…, dataN)	Ввод значений через устройство ввода (выбор документа через диалоговые окна, ввод PIN или OTP, CodeSession). Пример: $input$(КП: $CodeSession, OTP$), $input$(ПС: $CodeSession, OTP$) В примере пользователь через КП и ПС заполняет данные о CodeSession и OTP
formation(data1,data2,…dataN) → $JWS-M(JWSname)$	Формирование JWS-M из данных и запись под ключом JWSname
auth(source)	Прохождение аутентификации
save(data1,data2,…, dataN)	Сохранение данных
and, or	Варианты и/или
[data]	Массив, состоящий из элементов *data*
ASNdata	Подписанные данные, возвращенные СД
Attr	Атрибуты, которые извлекаются из ASNdata
AuthData	Подтверждение аутентификации пользователя
CMS	Структура, которая передается в качестве результата выполнения протокола активации подписи
CodeSession	Код сессии
Count	Количество данных (числовой тип);
Doc	Номер текущего документа
hData	Хэш-значение подписываемых данных
hDoc	Хэш-значение документа
IdSession	Идентификатор сессии

IdSlot	Идентификатор слота
Message	Информационное сообщение
NumberPhone	Номер мобильного телефона, указанный в данных подтверждения аутентификации
OTP	Одноразовый пароль
SignVal	Значение подписи
SlotList	Список слотов, содержащий информацию о слотах пользователя
Status	Статус выполнения операции (true/false)
ЕС ИФЮЛ	Единая система идентификации физических и юридических лиц
СОК	Сертификат открытого ключа

Сам алгоритм протокола активации подписи имеет вид:

1. auth(ПС: ЕС ИФЮЛ)(строгая аут.) or auth(ПС: ПС) (базовая аут.)
2. formation($hDoc, NumberPhone, IdNum, count, \dots, etc$) → $JWS - M(AuthData)$ and ПС $\xrightarrow{TLS}$ СД: JSON($AuthData$)
3. СД $\xrightarrow{TLS}$ ПС: JSON(IdSession, CodeSession)
4. display(ПС: CodeSession, hDoc[1], hDoc[2], … hDoc[n])
5. СД $\xrightarrow{TLS}$ СП: JSON(IdSession, AuthData)
6. СП $\xrightarrow{TLS}$ OTP: JSON(NumberPhone)
7. input(КП: CodeSession, OTP)
8. КП $\xrightarrow{TLS}$ СП: JSON(CodeSession, OTP)
9. СП $\xrightarrow{TLS}$ КП: JSON(IdSession, SlotList)
10. display(КП: SlotList)
11. input(КП: IdSlot, IdSession)
12. КП $\xrightarrow{TLS}$ СП: JSON(IdSlot, IdSession)
13. СП $\xrightarrow{TLS}$ СД: JSON(IdSession, СОК)
14. СД $\xrightarrow{TLS}$ СП: JSON(IdSession, [{hDoc[1], hData[1], ASNdata[1]}, {hDoc[2], hData[2], ASNdata[2]}, … , {hDoc[n], hData[n], ASNdata[n]}])
15. save(СП: [{hDoc[1], hData[1], ASNdata[1]}, {hDoc[2], hData[2], ASNdata[2]}, … ,{hDoc[n], hData[n], ASNdata[n]}])
16. КП $\xrightarrow{TLS}$ СП: JSON(IdSession)
17. СП $\xrightarrow{TLS}$ КП: JSON(IdSession, count, [{hDoc[1], ASNdata[1]}, {hDoc[2], ASNdata[2]}, … , {hDoc[n], ASNdata[n]})
18. display(КП: hDoc[i], attr[i])) and КП $\xrightarrow{TLS}$ СП: JSON(IdSession, $hData, Doc[i]$)
19. СП $\xrightarrow{TLS}$ КП: JSON(IdSession, hData, Status)
20. КП $\xrightarrow{TLS+BPACE(user)}$ УСП: DAP)
21. save(УСП: ДАП)
22. КП $\xrightarrow{TLS}$ СП: IdSession, h$Data$, IdSlot
23. СП $\xrightarrow{BPACE(own)}$ УСП: IdSession, h$Data$, idslot

24. УСП $\xrightarrow{BPACE(own)}$ СП: SignVal
25. СП $\xrightarrow{TLS}$ СД: JSON(IdSession, SignVal, hData)
26. save(СД: SignVal) and СД $\xrightarrow{TLS}$ СП: JSON($Status$)
27. СП $\xrightarrow{TLS}$ КП: JSON($Status$)
28. display(КП: message))
29. ПС $\xrightarrow{TLS}$ СД: JSON(IdSession)
30. СД $\xrightarrow{TLS}$ ПС: JSON([CMS])
31. display(ПС: CMS))

Заключение

Таким образом, предложенные механизмы обеспечивают достаточный уровень гарантий контроля пользователем личного ключа подписанта, используемого в СОП. Реализация данных механизмов позволяет снизить уровень риска неавторизованной выработки электронной цифровой подписи от лица пользователя и достигнуть соответствия требованиям проекта стандарта «Информационные технологии и безопасность. Требования безопасности к системам облачной подписи».

Список литературы

1. СТБ 34.101.45-2013 Информационные технологии и безопасность. Алгоритмы электронной цифровой подписи и транспорта ключа на основе эллиптических кривых. [Электронный ресурс] – Режим доступа: https://apmi.bsu.by/resources/std.html. – Дата доступа: 07.04.2023.

2. СТБ 34.101.bclo Информационные технологии и безопасность. Требования безопасности к системам облачной подписи. [Электронный ресурс] – Режим доступа: https://apmi.bsu.by/resources/std.html. – Дата доступа: 07.04.2023.

3. СТБ 34.101.87-2022 Информационные технологии и безопасность. Инфраструктуры аутентификации. [Электронный ресурс] – Режим доступа: https://apmi.bsu.by/resources/std.html. – Дата доступа: 07.04.2023.