ЦЕНТРАЛЬНЫЙ БАНК РОССИЙСКОЙ ФЕДЕРАЦИИ (БАНК РОССИИ)

СТАНДАРТ БАНКА РОССИИ

СТО БР НПС-6.1-2020

ФИНАНСОВЫЕ СООБЩЕНИЯ В НПС

ПРАВИЛА ОБМЕНА ДАННЫМИ В НАЦИОНАЛЬНОЙ ПЛАТЕЖНОЙ СИСТЕМЕ

Дата введения: 2019-09-30

Издание официальное

Москва
2020

Предисловие

ПРИНЯТ И ВВЕДЕН в действие приказом Банка России от 20 сентября 2019 года № ОД-2191 «О введении в действие стандарта СТО БР НПС-1.4-2019 Банка России «Финансовые сообщения в НПС. Правила обмена данными в национальной платежной системе».

Содержание

Введение

Настоящий Стандарт содержит рекомендации по реализации обмена данными в соответствии со стандартом ISO 20022[1] при переводе денежных средств в национальной платежной системе (далее – НПС).

1. Область применения

Настоящий Стандарт рекомендован к использованию разработчиками информационного и программного обеспечения, информационных систем при организации информационного взаимодействия между банками и их клиентами – юридическими лицами. В рамках межбанковского взаимодействия при оказании услуг по передаче финансовых сообщений возможно использование как собственных правил обмена данными, так и положений настоящего Стандарта.

Положения настоящего Стандарта применяются на добровольной основе, если только в отношении конкретных положений обязательность их применения не установлена нормативными актами Банка России или условиями договоров.

2. Термины и определения

В настоящем Стандарте применяются термины в соответствии со Стандартом Банка России СТО БР НПС-1.0-2017 «Финансовые сообщения в НПС. Общие положения» и Федеральным законом от 06.04.2011 N 63-ФЗ «Об электронной подписи», а также следующие термины с соответствующими определениями.

Сетевая модель BOC	–	базовая эталонная модель взаимодействия открытых систем в соответствии с ГОСТ Р ИСО/МЭК 7498-1-99 «Государственный стандарт Российской Федерации. Информационная технология. Взаимосвязь открытых систем. Базовая эталонная модель. Часть 1. Базовая модель»;
Сообщение	–	набор структурированной информации (в электронном виде), которой осуществляется обмен между двумя участниками информационного взаимодействия с использованием средств телекоммуникации;
Бизнес-сообщение	–	электронное сообщение стандарта ISO 20022, предназначенное для передачи прикладных данных между участниками информационного взаимодействия;
Транспортное сообщение	–	конверт, соответствующий спецификации SOAP 1.2, в заголовках которого находятся структуры, связанные с подписанием и шифрованием сообщения, а в теле передается бизнес-сообщение;
Способ доставки сообщения	–	группа параметров настройки для величин, определяющих характеристики доставки сообщений;
Система доставки сообщений	–	механизм, обеспечивающий принятие, транспортировку и доставку транспортных сообщений между участниками информационного взаимодействия;
Участник информационного взаимодействия	–	субъект, который создает, обрабатывает, принимает или отправляет электронные сообщения;
Хеш	–	строка ограниченной длины, полученная путем выполнения специальных преобразований над исходными данными в соответствии с ГОСТ Р 34.11-2012;
Deflate	–	алгоритм сжатия данных без потерь;
Base64	–	стандарт кодирования последовательности байт в строку и из строки в последовательность байт;
Валидация	–	проверка входных данных по заданным правилам.

[1] Международный стандарт ISO 20022 «Финансовые услуги – Универсальная схема сообщений для финансовой отрасли» (Financial services — Universal financial industry message scheme)

3. Общие положения

В соответствии с положениями части 6 «Характеристики транспортировки сообщений» (ISO 20022-6 «Message transport characteristics») стандарта ISO 20022 обмен данными в рамках НПС выполняется в следующих слоях:

- бизнес-слой – самый верхний слой, в котором определены правила и сценарии обмена бизнес-сообщениями, а также структуры бизнес-сообщений, используемых в процессах обмена в рамках конкретных моделей связей НПС. Бизнес-слой эквивалентен добавлению уровня 9 к Сетевой модели ВОС;
- слой транспортировки сообщений – слой, в котором определены правила обмена электронными сообщениями, независимо от бизнес-слоя. Слой транспортировки сообщений эквивалентен добавлению уровня 8 к Сетевой модели ВОС;
- прикладной слой – самый нижний слой обмена, соответствующий уровню 7 Сетевой модели ВОС.

Настоящий Стандарт определяет рекомендации по обмену данными в рамках НПС в каждом из указанных выше слоев. Соблюдение настоящих рекомендаций позволит обеспечить функциональную совместимость (интероперабельность) взаимодействующих в рамках НПС информационных систем, поддерживающих ISO 20022.

4. Бизнес-слой

В бизнес-слое обмен данными реализуется путем передачи бизнес-сообщений. Бизнес-сообщение представляет собой совокупность бизнес-заголовка и содержимого сообщения, описанного в Стандартах ISO 20022 НПС.

Бизнес-сообщения содержат только бизнес-данные и не должны содержать информацию о Системе доставки сообщений, о механизмах адресации, отправки, передачи или получения сообщений, а также прочую информацию, специфичную для использования в слое транспортировки сообщения или прикладном слое.

Бизнес-сообщения должны содержать всю информацию, необходимую для их корректной интерпретации Участниками информационного взаимодействия, должны быть понятны вне контекста транспортного конверта (раздел 5 настоящего Стандарта).

Бизнес-сообщения должны быть представлены в виде XML документов и должны соответствовать XML схемам и правилам заполнения документов, описанным в Стандартах ISO 20022 НПС.

4.1. Правила обмена данными в бизнес-слое

Правила обмена данными в бизнес-слое определяются Стандартами ISO 20022 НПС, содержащими модели связей между участниками перевода денежных средств, а также используемые в рамках этих моделей сообщения.

При описании обмена в бизнес-слое задаются Способы доставки сообщений (MessageTransportMode), которые определяют группы характеристик доставки сообщений, описанные в части 1 «Метамодель» (ISO 20022-1 «Metamodel») стандарта ISO 20022[2]. Характеристики доставки сообщений справочно приведены в Таблице 4.1 настоящего Стандарта.

Определенные в бизнес-слое Способы доставки сообщений должны быть реализованы в слое транспортировки сообщений и/или прикладном слое.

Таблица 4.1. Характеристики доставки сообщений

Характеристика	Описание
Обеспечение доставки (DeliveryAssurance)	Принимает одно из следующих значений: • не менее одного раза (AT_LEAST_ONCE), • в точности один раз (EXACTLY_ONCE), • не более одного раза (AT_MOST_ONCE)
Асинхронность отправителя (SenderAsynchronicity)	Принимает одно из следующих значений: • конечная точка синхронна (ENDPOINT_SYNCHRONOUS), • обмен синхронен (CONVERSATION_SYNCHRONOUS),

[2] www.iso20022.org

	• асинхронен (ASYNCHRONOUS)
Асинхронность получателя (ReceiverAsynchronicity)	Принимает одно из следующих значений: • конечная точка синхронна (ENDPOINT_SYNCHRONOUS), • обмен синхронен (CONVERSATION_SYNCHRONOUS), • асинхронен (ASYNCHRONOUS)
Порядок доставки сообщения (MessageDeliveryOrder)	Принимает одно из следующих значений: • порядок сохраняется для всех получателей и отправителей сообщений (EXPECTED_CAUSAL_ORDER), • порядок сохраняется только для каждой пары получателя и отправителя сообщений (FIFO_ORDERED), • порядок не сохраняется (UNORDERED)
Окно доставки сообщения (MessageDeliveryWindow)	Максимальная продолжительность времени, в пределах которого транспортное сообщение может быть доставлено без учета очередности. Значения должны быть большими или равными нулю.
Окно отправки сообщения (MessageSendingWindow)	Максимальная продолжительность времени, в течение которого транспортное сообщение может быть отправлено без учета очередности. Значения должны быть большими или равными нулю.
Метод рассылки сообщения (MessageCasting)	Принимает одно из следующих значений: • односторонняя рассылка (UNICAST), • групповая рассылка (MULTICAST), • рассылка по списку (BROADCAST), • любая рассылка (ANYCAST)
Ограниченная задержка связи (BoundedCommunicationDelay)	Максимальная продолжительность времени, в пределах которого сообщение должно быть доставлено. В качестве значения должна быть задана продолжительность времени в соответствии с ISO 8601
Валидация сообщений включена/ отключена (MessageValidationOnOff)	Принимает одно из следующих значений: • валидация включена (VALIDATION_ON) – сообщения валидируются Системой доставки сообщений, • валидация отключена (VALIDATION_OFF) – сообщения не валидируются Системой доставки сообщений
Результаты валидации сообщения (MessageValidationResults)	Принимает одно из следующих значений: • отклонение (REJECT), • отклонение и доставка (REJECT_AND_DELIVER), • доставка (DELIVER)
Уровень валидации сообщения (MessageValidationLevel)	Уровень валидации сообщения, требуемый Системой доставки сообщений. Принимает одно из следующих значений: • сообщение не валидировалось (NO_VALIDATION), • провалидирован синтаксис сообщения (SYNTAX_VALID), • сообщение соответствует XML схеме (SCHEMA_VALID), • сообщение соответствует … правилам (MESSAGE_VALID), • сообщение соответствует бизнес-правилам (RULE_VALID), • сообщение соответствует рыночной практике (MARKET_PRACTICE_VALID), • сообщение соответствует бизнес-процессу (BUSINESS_PROCESS_VALID), • полностью валидное сообщение (COMPLETELY_VALID)
Длительность хранения (Durability)	Принимает одно из следующих значений: • долговременно (DURABLE), • постоянно (PERSISTENT), • временно (TRANSIENT)
Максимальное отклонение времени (MaximumClockVariation)	Максимальное отклонение от Всемирного координированного времени (UTC) для обеспечиваемого Способа доставки сообщения
Максимальный размер сообщения (MaximumMessageSize)	Максимальный размер транспортного сообщения в килобайтах (любое положительное целое число, большее нуля).

4.2. Структура бизнес-сообщений

Бизнес-сообщение имеет следующую структуру:

```
<BusinessMessage xmlns="urn:cbrf:iso:20022:business-message">
  <AppHdr xmlns="urn:iso:std:iso:20022:tech:xsd:head.001.001.02">
    <!-- Содержимое заголовка -->
  </AppHdr>
  <Document xmlns="urn:iso:std:iso:20022:tech:xsd:xxxx.nnn.nnn.nn">
    <!-- Содержимое документа -->
  </Document>
</BusinessMessage>
```

В качестве заголовка бизнес-сообщения используется структура Business Application Header (BAH). Структура задана в документе «ISO 20022 Business Application Header. Message Definition Report» (версия от 30 августа 2019 года)[3]. Рамочные правила её использования заданы в документе «ISO 20022 Business Application Header Message Usage Guide Version 1.9» (версия от октября 2019 года). Описание структуры заголовка бизнес-сообщения и правила её заполнения для целей использования в НПС приведены в Таблице 4.2.

Таблица 4.2. Структура заголовка бизнес-сообщения

Прикладной атрибут	Правила формирования
Отправитель	Идентификатор отправителя записывается в реквизит AppHdr/Fr/*. При этом обязательно указывается идентификационная схема в реквизите */SchmeNm/Cd (при использовании идентификационных кодов, зарегистрированных в ISO 20022) или */SchmeNm/Prtry (При использовании собственных идентификационных кодов). В качестве основных типов идентификаторов могут выступать: LEI (Legal Entity Identifier), уникальный 20-значный буквенноцифровой идентификатор (код) юридического лица, который присваивается в соответствии с международным стандартом ISO 17442 ОГРН, идентификационная схема - OGRN ИНН, идентификационная схема - TXID Российский БИК, идентификационная схема – RUCBC УИС, идентификационная схема - RUCB Может быть указан иной идентификатор, принятый Участниками информационного взаимодействия, позволяющий однозначно идентифицировать участника в цепочке обмена. При формировании идентификатора отправителя и типа идентификатора недопустимо использование символов пробела, перевода строки, возврата каретки и табуляции.
Получатель	Идентификатор получателя записывается в реквизит AppHdr/To/*. Требования к идентификаторам атрибута Получатель аналогичны требованиям к идентификаторам атрибута Отправитель
Идентификатор сообщения	Идентификатор сообщения записывается в реквизит AppHdr/BizMsgIdr. Идентификатор сообщения должен быть уникальным по определенному Участнику информационного взаимодействия. При формировании идентификаторов сообщения рекомендуется использовать алгоритмы создания универсальных уникальных идентификаторов (UUID) согласно спецификации RFC 4122 (см. таблица 5.1). В целях соответствия требованиям BAH к длине идентификатора, при записи UUID в реквизит AppHdr/BizMsgIdr, в строковое представление UUID не должен включаться идентификатор пространства имен «urn:uuid:», а также должны быть исключены символы «-». Пример идентификатора: «f81d4fae7dec11d0a76500a0c91e6bf6»
Тип сообщения	Тип бизнес-сообщения записывается в реквизит AppHdr/MsgDefIdr. Могут использоваться только типы сообщений, применяемые в НПС.

[3] www.iso20022.org

	Пример: «pain.001.001.07»
Бизнес-сервис	Бизнес-сервис записывается в реквизит AppHdr/BizSvc. Реквизит опциональный, правила заполнения определяются Участниками информационного взаимодействия.
Дата и время формирования сообщения	Дата и время формирования сообщения записывается в реквизит AppHdr/CreDt Время указывается по Гринвичу

5. Слой транспортировки сообщений

5.1. Правила обмена данными в слое транспортировки сообщений

Обмен данными в слое транспортировки сообщений выполняется с использованием транспортных сообщений, в которые упакованы бизнес-сообщения. Структура и правила формирования транспортных сообщений определены в разделе 5.2 Стандарта.

Обмен выполняется по следующему сценарию:

1. Отправитель формирует бизнес-сообщение;
2. Отправитель упаковывает сформированное бизнес-сообщение в транспортное сообщение;
3. Отправитель отправляет транспортное сообщение получателю;
4. Получатель принимает транспортное сообщение;
5. Получатель распаковывает транспортное сообщение и извлекает из него бизнес-сообщение;
6. Если процедура распаковки закончилась неудачей, Получатель формирует сообщение об ошибке в соответствие с правилами раздела 5.2.5 Стандарта и отправляет его Отправителю;
7. Получатель выполняет валидацию бизнес-сообщения, если необходимость валидации определена на бизнес-уровне;
8. Получатель выполняет обработку бизнес-сообщения в случае успешной валидации (либо если валидация не проводилась);
9. Если валидация закончилась неудачей, Получатель формирует сообщение об ошибке в соответствие с правилами раздела 5.2.5 Стандарта и отправляет его отправителю.

Процедуры упаковки (распаковки) транспортного сообщения включают в себя операции шифрования (расшифрования), а также подписания (проверки) электронной подписи. Правила выполнения указанных процедур описаны в разделах 5.2.3 и 5.2.4 настоящего Стандарта.

Отправитель может отправлять транспортные сообщения более чем одному получателю в зависимости от характеристики «Метод рассылки сообщений», установленной в бизнес-слое.

Процедуры доставки сообщений от отправителя к получателю могут выполняться с использованием одной или более Систем доставки сообщений, которые обеспечивают доставку сообщений в пределах своей зоны доставки, и могут, в том числе, выполнять процедуры рассылки транспортных сообщений.

Системы доставки сообщений могут использовать для идентификации получателя (получателей) как реквизиты транспортного сообщения, так и реквизиты бизнес-сообщения (например, значения элементов заголовка бизнес-сообщения). Недопустимо изменение содержимого бизнес-сообщения при его обработке Системой доставки сообщений.

На уровне транспортировки сообщений информационное взаимодействие осуществляется в соответствии со спецификацией WS-I Basic Profile Version 2.0.

5.2. Структура и правила формирования транспортного конверта сообщения

5.2.1. Общие правила формирования транспортного конверта сообщения

При описании структуры и правил формирования транспортного сообщения используются спецификации, указанные в Таблице 5.1.

Таблица 5.1. Используемые спецификации

Краткое наименование	Полное наименование
SOAP 1.2	SOAP Version 1.2 Part 1: Messaging Framework (Second Edition). W3C Recommendation 27 April 2007. http://www.w3.org/TR/soap12-part1
WS Security 1.1.1	Web Services Security: SOAP Message Security Version 1.1.1. OASIS Standard. 18 May 2012. http://docs.oasis-open.org/wss-m/wss/v1.1.1/os/wss-SOAPMessageSecurity-v1.1.1-os.html

Краткое наименование	Полное наименование
XML-binary Optimized Packaging	XML-binary Optimized Packaging. W3C Recommendation 25 January 2005. http://www.w3.org/TR/2005/REC-xop10-20050125
XML Encryption 1.1	XML Encryption Syntax and Processing Version 1.1. W3C Recommendation 11 April 2013 https://www.w3.org/TR/xmlenc-core/
XML 1.0	Extensible Markup Language (XML) 1.0 (Fifth Edition). W3C Recommendation 26 November 2008. http://www.w3.org/TR/2008/REC-xml-20081126
RFC 2045	RFC 2045: Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies. http://tools.ietf.org/rfc/rfc2045.txt
RFC 3986	RFC 3986. Uniform Resource Identifier (URI): Generic Syntax. http://tools.ietf.org/html/rfc3986
RFC 4122	RFC 4122. A Universally Unique IDentifier (UUID) URN Namespace. http://www.ietf.org/rfc/rfc4122.txt
RFC 4648	RFC 4648: The Base16, Base32, and Base64 Data Encodings. http://tools.ietf.org/rfc/rfc4648.txt

При описании структуры электронных сообщений используются пространства имен, приведенные в Таблице 5.2.

Таблица 5.2. Перечень пространств имен

Префикс	Пространство имен
ds	http://www.w3.org/2000/09/xmldsig#
soap	http://www.w3.org/2003/05/soap-envelope
xenc	http://www.w3.org/2001/04/xmlenc#
wsa	http://schemas.xmlsoap.org/ws/2004/08/addressing
wsse	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
wsu	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd
head	urn:iso:std:iso:20022:tech:xsd:head.001.001.02
msg	urn:cbrf:iso:20022:business-message
vr	urn:cbrf:iso:20022:validation-results

Транспортный конверт сообщения имеет следующую структуру:

```
<soap:Envelope
    xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
  <soap:Header>
    <!-- Заголовок транспортного конверта сообщения -->
  </soap:Header>
  <soap:Body wsu:Id="BusinessMessage">
    <!-- Бизнес-сообщение -->
  </soap:Body>
</soap:Envelope>
```

При формировании электронных сообщений используется кодировка UTF-8.

5.2.2. Двоичные вложения

Бизнес-сообщение может содержать в своей структуре двоичные вложения. При передаче бизнес-сообщений в составе транспортного сообщения, двоичные вложения могут передаваться в теле сообщения в стандарте кодирования Base64 (согласно RFC 4648) или оформлены в виде отдельных MIME-частей.

Оформление двоичных вложений в виде отдельных MIME-частей выполняется для оптимизации процессов передачи и уменьшения объемов сообщений с двоичными вложениями.

При передаче двоичных вложений в виде отдельных MIME-частей в теле сообщения создается ссылка на MIME-часть согласно спецификации XML-binary Optimized Packaging.

MIME-части сообщения используются исключительно для оптимизации процессов передачи сообщения. Обработка сообщения должна осуществляться согласно модели обработки XOP спецификации XML-binary Optimized Packaging.

Рекомендуется оформлять двоичные вложения в виде MIME-частей в том случае, если размер двоичного вложения превышает 100 Кб.

5.2.3. Подписание сообщений и проверка электронной подписи сообщений

Подписание и проверка электронной подписи сообщений осуществляется в соответствии со спецификацией WS Security 1.1.1. Рекомендации по созданию и проверке электронной подписи приведены в Приложении 1.

Структура элемента wsse:Security, содержащего электронную подпись, представлена в Таблицах 5.3-5.4.

Таблица 5.3. Структура элемента wsse:Security, содержащего электронную подпись

№	Элемент	Описание	Область значений	Мн.
а)	Признак обязательности интерпретации (@soap:mustUnderstand)	признак того, что получатель сообщения должен обработать элемент согласно спецификации. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	логическое значение	1
1	Сертификат ключа проверки электронной подписи (wsse:BinarySecurityToken)	сертификат ключа проверки электронной подписи, с помощью которой подписано сообщение	строка символов, закодированная с помощью Base64	1
а)	Идентификатор (@wsu:Id)	уникальный идентификатор элемента в сообщении	идентификатор элемента документа	1
б)	Вид (@ValueType)	стандарт сертификата ключа проверки электронной подписи. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1
в)	Кодировка (@EncodingType)	схема кодирования сертификата ключа проверки электронной подписи. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1
2	Электронная подпись (ds:Signature)	электронная подпись сообщения согласно таблице 5.4	структура сведений описана в Таблице 5.4	1

Таблица 5.4. Структура электронной подписи (ds:Signature)

№	Элемент	Описание	Область значений	Мн.
1	Подписываемые сведения (ds:SignedInfo)	набор подписываемых элементов, содержащих хэш бизнес-сообщения и набор инструкций по обработке хэша и подписи	определяется областями значений вложенных элементов	1
1.1	Метод каноникализации (ds:CanonicalizationMethod)	инструкции по каноникализации	отсутствует	1
а)	Алгоритм (@Algorithm)	алгоритм каноникализации. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1
1.2	Метод подписывания (ds:SignatureMethod)	элемент, содержащий алгоритм формирования и проверки подписи	отсутствует	1
а)	Алгоритм (@Algorithm)	идентификатор алгоритма формирования и проверки подписи. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1
1.3	Ссылка (ds:Reference)	ссылка на данные, для которых формируется хэш	определяется областями значений	1

№	Элемент	Описание	Область значений	Мн.
			вложенных элементов	
а)	Идентификатор (@URI)	идентификатор хэшируемой области. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	идентификатор элемента документа	1
1.3.1	Преобразования (ds:Transforms)	инструкции по преобразованиям хэшируемой области	определяется областями значений вложенных элементов	1
1.3.1.1	Преобразование (ds:Transform)	элемент, содержащий единичную инструкцию по преобразованию хэшируемой области	определяется областями значений вложенных элементов	1
а)	Алгоритм (@Algorithm)	идентификатор алгоритма преобразования. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1
1.3.1.2	Метод хэширования (ds:DigestMethod)	сведения о методе хэширования	отсутствует	1
а)	Алгоритм (@Algorithm)	идентификатор алгоритма хэширования. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1
1.3.1.3	Хеш (ds:DigestValue)	значение хэша	строка символов, закодированная с помощью Base64	1
2	Значение подписи (ds:SignatureValue)	значение подписи	определяется областями значений вложенных элементов	1
3	Сведения о ключе (ds:KeyInfo)	сведения об открытом ключе, используемом для проверки подписи	определяется областями значений вложенных элементов	1
3.1	Ссылка на токен (wsse:SecurityTokenReference)	сведения о местонахождении ключа	определяется областями значений вложенных элементов	1
3.1.1	Ссылка (wsse:Reference)	элемент, содержащий идентификатор местонахождения ключа	отсутствует	1
а)	Идентификатор (@URI)	идентификатор местонахождения ключа. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	идентификатор элемента документа	1
б)	Вид (@ValueType)	стандарт сертификата ключа. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1

5.2.4. Шифрование и расшифрование сообщений

Шифрование и расшифрование сообщений осуществляется в соответствии со спецификацией WS Encryption 1.1. Шифрование сообщений выполняется после их подписания.

Шифрование сообщения осуществляется следующим образом.

1. В заголовок транспортного конверта сообщения добавляется элемент wsse:Security. Данный элемент должен предшествовать элементу wsse:Security, содержащему электронную подпись, если таковой имеется.
2. Сертификат открытого ключа, с помощью которого шифруется сообщение, включается в элемент wsse:Security/wsse:BinarySecurityToken. Сертификат соответствует стандарту X.509

(указывается в атрибуте ValueType) и кодируется с использованием стандарте кодирования Base64 (указывается в атрибуте EncodingType).

3. В элементе wsse:Security/xenc:EncryptedKey/ds:KeyInfo указывается ссылка на сертификат открытого ключа.
4. Бизнес-сообщение, содержащееся в элементе soap:Body шифруется в соответствии со спецификацией XML Encryption 1.1. Зашифрованное сообщение помещается в элемент xenc:EncryptedData, который заменяет исходное бизнес-сообщение.
5. В элементе wsse:Security/xenc:EncryptedKey/xenc:ReferenceList указывается ссылка на зашифрованное сообщение.

Расшифрование сообщения осуществляется следующим образом.

1. Если в заголовке транспортного конверта сообщения содержится элемент wsse:Security/ xenc:EncryptedKey, то сообщение является зашифрованным и требуется его расшифрование.
2. Для сертификата открытого ключа, ссылка на который указана в элементе wsse:Security/ xenc:EncryptedKey/ds:KeyInfo, определяется закрытый ключ для расшифрования сообщения.
3. Данные, ссылка на которые указана в элементе wsse:Security/xenc:EncryptedKey/xenc:ReferenceList, дешифруются с помощью закрытого ключа. Результат помещается в сообщение вместо зашифрованных данных.
4. Из сообщения удаляется элемент wsse:Security, содержащий сведения о шифровании сообщения.
5. В случае невозможности расшифровать сообщение формируется сообщение об ошибке в соответствии с требованиями раздела 5.2.5 Стандарта.

Структура элемента wsse:Security, содержащего сведения о шифровании сообщения, описана в Таблицах 5.5-5.6.

Таблица 5.5. Структура элемента wsse:Security, содержащего сведения о шифровании сообщения

№	Элемент	Описание	Область значений	Мн.
а)	Признак обязательности интерпретации (@soap:mustUnderstand)	признак того, что получатель сообщения должен обработать элемент согласно спецификации. Атрибут имеет фиксированное значение «true»	логическое значение	1
1	Сертификат открытого ключа (wsse:BinarySecurityToken)	сертификат открытого ключа, используемого для шифрования сообщения	строка символов, закодированная с помощью Base64	1
а)	Идентификатор (@wsu:Id)	уникальный идентификатор элемента в сообщении	идентификатор элемента документа	1
б)	Вид (@ValueType)	стандарт сертификата открытого ключа. Атрибут имеет фиксированное значение, аналогичное описанному в Приложении 1	унифицированный идентификатор ресурса	1
в)	Кодировка (@EncodingType)	схема кодирования сертификата открытого ключа. Атрибут имеет фиксированное значение, аналогичное описанному в Приложении 1	унифицированный идентификатор ресурса	1
2	Сведения о зашифрованных данных (xenc:EncryptedKey)	сведения о зашифрованных данных	структура сведений описана в Таблице 5.6	1

Таблица 5.6. Структура сведений о зашифрованных данных (xenc:EncryptedKey)

№	Элемент	Описание	Область значений	Мн.
1	Метод шифрования (xenc:EncryptionMethod)	сведения о методе шифрования	определяется областями значений вложенных элементов	1
а)	Алгоритм (@Algorithm)	идентификатор алгоритма шифрования	унифицированный идентификатор ресурса	1

№	Элемент	Описание	Область значений	Мн.
2	Сведения о ключе (ds:KeyInfo)	сведения об открытом ключе, используемом для шифрования	определяется областями значений вложенных элементов	1
2.1	Ссылка на токен (wsse:SecurityTokenReference)	сведения о местонахождении ключа	определяется областями значений вложенных элементов	1
2.1.1	Ссылка (wsse:Reference)	элемент, содержащий идентификатор местонахождения ключа	отсутствует	1
а)	Идентификатор (@URI)	идентификатор местонахождения ключа. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	идентификатор элемента документа	1
б)	Вид (@ValueType)	стандарт сертификата ключа. Атрибут имеет фиксированное значение в соответствии с описанием Приложения 1	унифицированный идентификатор ресурса	1
3	Список ссылок (xenc:ReferenceList)	список ссылок на зашифрованные данные	определяется областями значений вложенных элементов	1
3.1	Ссылка на данные (xenc:DataReference)	ссылка на зашифрованные данные	отсутствует	1
а)	Идентификатор (@URI)	идентификатор зашифрованных данных	идентификатор элемента документа	1

Структура зашифрованных данных представлена в Таблице 5.7.

Таблица 5.7. Структура зашифрованных данных (xenc:EncryptedData)

№	Элемент	Описание	Область значений	Мн.
а)	Идентификатор (@Id)	идентификатор зашифрованных данных	идентификатор элемента документа	1
1	Метод шифрования (xenc:EncryptionMethod)	сведения о методе шифрования	отсутствует	1
а)	Алгоритм (@Algorithm)	идентификатор алгоритма шифрования	унифицированный идентификатор ресурса	1
2	Зашифрованные данные (xenc:CipherData)	элемент, содержащий зашифрованные данные	определяется областями значений вложенных элементов	1
2.1	Зашифрованное значение (xenc:CipherValue)	зашифрованные данные	строка символов, закодированная с помощью Base64	1

5.2.5. Формирование сообщений об ошибках

Сообщения об ошибках формируются в соответствии со спецификацией SOAP 1.2.

Сведения об ошибке передаются в теле транспортного конверта и должны быть единственным элементом.

Таблица 5.8. Структура сведений об ошибке (soap:Fault)

№	Элемент	Описание	Тип данных	Мн.
1	Код ошибки (soap:Code)	формализованные сведения о коде ошибки	определяется областями значений вложенных элементов	1
1.1	Значение (soap:Value)	значение класса ошибки	перечисление soap: faultCodeEnum в	1

			соответствие со спецификацией SOAP 1.2	
1.2	Код ошибки (soap:Subcode)	сведения о коде ошибки в соответствие с ее классом	определяется областями значений вложенных элементов	1
1.2.1	Значение (soap:Value)	значение кода ошибки. Перечень типовых кодов ошибок представлен в таблице 5.9	строка символов	1
2	Причина (soap:Reason)	сведения об ошибке	определяется областями значений вложенных элементов	1..*
2.1	Текст (soap:Text)	текст, описывающий ошибку	строка символов	1
a)	Язык (@xml:lang)	идентификатор языка, на котором приведен текст, описывающий ошибку		1
3	Сведения об ошибке (soap:Detail)	дополнительные детализированные сведения об ошибке	определяется областями значений вложенных элементов	0..1

Элемент soap:Code/soap:Value заполняется согласно требованиям спецификации SOAP 1.2.

Перечень типовых кодов ошибок представлен в Таблице 5.9.

Таблица 5.9. Перечень типовых ошибок

Класс ошибки	Код ошибки	Текст ошибки
soap:Sender	wsse:UnsupportedSecurityToken	рекомендуется использовать типовые коды ошибок, определенные стандартом WS-Security
soap:Sender	wsse:UnsupportedAlgorithm	
soap:Sender	wsse:InvalidSecurity	
soap:Sender	wsse:InvalidSecurityToken	
soap:Sender	wsse:FailedAuthentication	
soap:Sender	wsse:FailedCheck	
soap:Sender	wsse:SecurityTokenUnavailable	
soap:Sender	wsse:MessageExpired	
soap:Sender	vr:InvalidSyntax	Синтаксис сообщения некорректен
soap:Sender	vr:InvalidStructure	Структура сообщения не соответствует XML схеме
soap:Sender	vr:MessageRulesViolated	Сообщение не соответствует структурным правилам
soap:Sender	vr:BusinessRulesViolated	Сообщение не соответствует бизнес-правилам
soap:Sender	vr:MarketPracticeViolated	Сообщение не соответствует рыночной практике
soap:Sender	vr:BusinessProcessViolated	Сообщение не соответствует бизнес-процессу

В случае несоответствия сообщения правилам формирования (vr:MessageRulesViolated, vr:BusinessRulesViolated) в элемент soap:Fault/soap:Detail включаются элементы vr:Result, содержащие сведения о результатах валидации.

Для каждого провалидированного элемента и каждого правила формируется отдельный элемент vr:Result. Сведения об ошибках валидации (error) включаются в сообщение в обязательном порядке. В отладочных целях в сообщение об ошибке могут быть включены сведения об иных результатах валидации (valid, irrelevant, absent, unsupported).

Таблица 5.10. Структура сведений о результатах валидации

№	Элемент	Описание	Тип данных	Мн.
1	Результат валидации (vr:Result)	результат валидации	определяется областями значений вложенных элементов	1..*
1.1	Вид результата валидации (vr:Kind)	кодовое обозначение вида результата валидации	одно из следующих значений: • valid – проверка пройдена успешно, • error – ошибка,	1

№	Элемент	Описание	Тип данных	Мн.
			• irrelevant – элементы не проверялись, т.к. для них не выполнено предусловие, • absent – элементы для валидации отсутствуют в сообщении, • unsupported – правило не поддерживается	
1.2	Описание результата валидации (vr:Description)	описание результата валидации	строка символов	1
1.3	Код правила (vr:RuleCode)	код правила	строка символов	1
1.4	Описание правила (vr:RuleDescription)	описание правила	строка символов	1
1.5	Спецификация правила (vr:RuleOcl)	спецификация правила на формальном объектном языке ограничений (OCL)	строка символов	0..1
1.6	Элемент (vr:Element)	провалидированный элемент сообщения	определяется областями значений вложенных элементов	1..*
1.6.1	Путь (vr:Path)	путь к элементу сообщения, представленный в виде XPath выражения	строка символов	1
1.6.2	Описание (vr:Description)	описание пути к элементу на русском языке	строка символов	0..1

6. Прикладной слой

В качестве протокола передачи данных прикладного слоя Участниками информационного взаимодействия выбирается один из протоколов, обеспечивающий требуемые характеристики доставки сообщений, определяемые в бизнес-слое.

Рекомендуется использовать следующие протоколы:

- HTTPS: Hypertext Transfer Protocol - HTTP/1.1 (RFC 2616) совместно со спецификацией HTTP Over TLS (RFC 2818);
- AMQP: ISO/IEC 19464:2014 Information technology - Advanced Message Queuing Protocol (AMQP) v1.0 specification

Приложение 1

Подписание сведений, передающихся в транспортном конверте сообщения с помощью электронной подписи осуществляется следующим образом (см. рисунок А.1). Для краткости на схеме опущены объявления пространств имен, а значения некоторых атрибутов приведены с сокращениями.

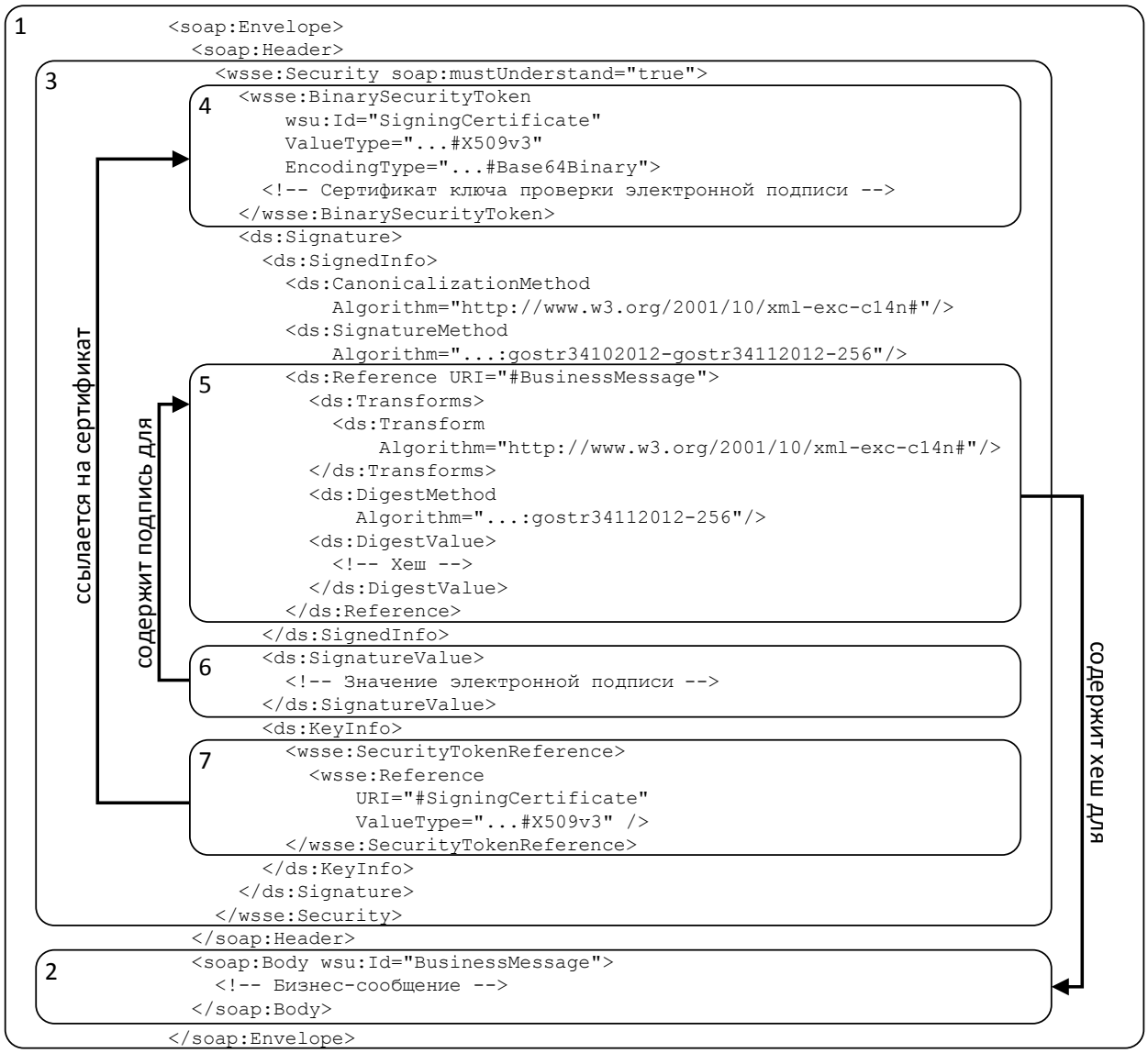

Рисунок А.1. Схема подписания сообщения

На 1-ом шаге формируется транспортный конверт сообщения (1-й шаг не имеет непосредственного отношения к формированию подписи, однако приводится в настоящем алгоритме как подготовительное мероприятие для создания XML-структур, в которые будет погружена подпись и подписываемые сведения).

На 2-ом шаге в элемент soap:Body транспортного конверта помещается бизнес-сообщение. В атрибуте @wsu:ID элемента soap:Body задается идентификатор, который должен быть уникальным в рамках конверта и его содержимого; для идентификации тела сообщения рекомендуется использовать значение «BusinessMessage». Формируемая электронная подпись будет ссылаться по данному идентификатору на подписываемые данные.

На 3-ем шаге в заголовке транспортного конверта сообщения формируется блок сведений wsse:Security. У элемента задается атрибут @soap:mustUnderstand со значением «true». Это обязывает получателя сообщения обрабатывать данный блок сведений. В случае если он не сможет обеспечить обработку электронной подписи, то он должен уведомить отправителя соответствующим сообщением об ошибке.

На 4-ом шаге в блок сведений wsse:Security включается элемент wsse:BinarySecurityToken, содержащий сертификат ключа проверки электронной подписи. В атрибуте @wsu:Id указывается идентификатор сертификата, который должен быть уникальным в рамках конверта и его содержимого; для идентификации сертификата рекомендуется использовать значение «SigningCertificate». Формируемая электронная подпись будет ссылаться на сертификат по этому идентификатору. Сертификат должен соответствовать стандарту X.509 и кодироваться с использованием стандарта кодирования Base64.

Стандарт сертификата указывается в атрибуте @ValueType со значением «http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3». Схема кодирования сертификата указывается в атрибуте @EncodingType со значением «http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary».

На 5-ом шаге формируется хеш бизнес-сообщения. Для этого в блоке сведений wsse:Security формируется элемент ds:Signature/ds:SignedInfo/ds:Reference. В атрибуте @URI этого элемента указывается ссылка на подписываемое бизнес-сообщение (если на 2-м шаге задан рекомендуемый идентификатор, то в атрибуте @URI должна быть указана ссылка «#BusinessMessage»). Если двоичные вложения передаются в виде MIME, то далее они должны быть перенесены в бизнес-сообщение согласно алгоритму, описанному в спецификации XML-binary Optimized Packaging. После этого бизнес-сообщение приводится к каноническому виду с помощью алгоритма Exclusive XML Canonicalization Version 1.0. Для полученного фрагмента XML документа в соответствии с ГОСТ Р 34.11-2012 вычисляется 256-битный хеш, который записывается в элемент ds:DigestValue.

Алгоритм каноникализации указывается в атрибуте ds:Transforms/ds:Transform/@Algorithm со значением «http://www.w3.org/2001/10/xml-exc-c14n#». Алгоритм вычисления хеша указывается в атрибуте ds:DigestMethod/@Algorithm со значением «urn:ietf:params:xml:ns:cpxmlsec:algorithms:gostr34112012-256».

На 6-ом шаге для хеша бизнес-сообщения формируется электронная подпись. Для этого в элемент ds:Signature/ds:SignedInfo добавляются сведения об алгоритме его каноникализации (Exclusive XML Canonicalization Version 1.0) и алгоритме вычисления электронной подписи (ГОСТ 34.10-2012, 256 бит). Затем с помощью указанных алгоритмов элемент ds:Signature/ds:SignedInfo приводится к каноническому виду и для полученного фрагмента XML документа вычисляется электронная подпись. Полученное значение записывается в элемент ds:SignatureValue.

Алгоритм каноникализации указывается в атрибуте ds:CanonicalizationMethod/@Algorithm со значением «http://www.w3.org/2001/10/xml-exc-c14n#». Алгоритм вычисления электронной подписи указывается в атрибуте ds:SignatureMethod/@Algorithm со значением «urn:ietf:params:xml:ns:cpxmlsec:algorithms:gostr34102012-gostr34112012-256».

На 7-ом шаге в сведениях об электронной подписи в элементе ds:KeyInfo формируется ссылка на сертификат ключа проверки электронной подписи. Для этого используется элемент wsse:SecurityTokenReference. Ссылка на сертификат указывается в атрибуте wsse:Reference/@URI (если на 4-м шаге задан рекомендуемый идентификатор, то в атрибуте @URI должна быть указана ссылка «#SigningCertificate»). Стандарт сертификата указывается в атрибуте wsse:Reference/@ValueType со значением «http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3».

Проверка электронной подписи сообщения осуществляется следующим образом (см. рисунок А.2). Для краткости на схеме опущены объявления пространств имен, а значения некоторых атрибутов приведены с сокращениями.

На 1-ом шаге в заголовке транспортного конверта сообщения проверяется наличие элемента wsse:Security/ds:Signature. Если элемент присутствует, то сообщение является подписанным и требуется проверка наложенной электронной подписи.

На 2-ом шаге проверяется сертификат открытого ключа электронной подписи в элементе wsse:BinarySecurityToken. Сертификат должен быть корректным, а в элементе ds:KeyInfo/ wsse:SecurityTokenReference должна присутствовать ссылка на него.

На 3-ем шаге проверяется действительность электронной подписи. Для этого элемент ds:SignedInfo, приводится к каноническому виду с помощью алгоритма Exclusive XML Canonicalization Version 1.0. Для полученного фрагмента XML документа вычисляется 256-битное значение электронной подписи в соответствии с ГОСТ 34.10-2012. Если результат совпадает со значением элемента ds:SignatureValue, то подпись является действительной.

Проверяется значение атрибута ds:SignedInfo/ds:CanonicalizationMethod/@Algorithm, оно должно соответствовать используемому методу каноникализации и иметь значение «http://www.w3.org/2001/10/xml-exc-c14n#». Проверяется значение атрибута ds:SignedInfo/ds:SignatureMethod/@Algorithm, оно должно соответствовать используемому алгоритму вычисления электронной подписи и иметь значение «urn:ietf:params:xml:ns:cpxmlsec:algorithms:gostr34102012-gostr34112012-256».

На 4-ом шаге проверяется целостность бизнес-сообщения. Если двоичные вложения передаются в виде MIME, то перед проверкой целостности они должны быть перенесены в бизнес-сообщение согласно алгоритму, описанному в спецификации XML-binary Optimized Packaging. Затем бизнес-сообщение приводится к каноническому виду с помощью алгоритма Exclusive XML Canonicalization Version 1.0. Для полученного фрагмента XML документа в соответствии с ГОСТ Р 34.11-2012 вычисляется 256-битный хеш. Если полученное значение совпадает со значением элемента ds:SignedInfo/ds:Reference/ds:DigestValue, то бизнес-сообщение целостно.

Проверяется значение атрибута ds:SignedInfo/ds:Reference/ds:Transforms/ds:Transform/@Algorithm, оно должно соответствовать используемому методу каноникализации и иметь значение «http://www.w3.org/2001/10/xml-exc-c14n#». Проверяется значение атрибута ds:SignedInfo/ ds:Reference/ds:DigestMethod/@Algorithm, оно должно соответствовать используемому алгоритму вычисления хеша и иметь значение «urn:ietf:params:xml:ns:cpxmlsec:algorithms:gostr34112012-256».

В случае неудачной проверки электронной подписи формируется сообщение об ошибке в соответствии с требованиями раздела 5.2.5 Стандарта.

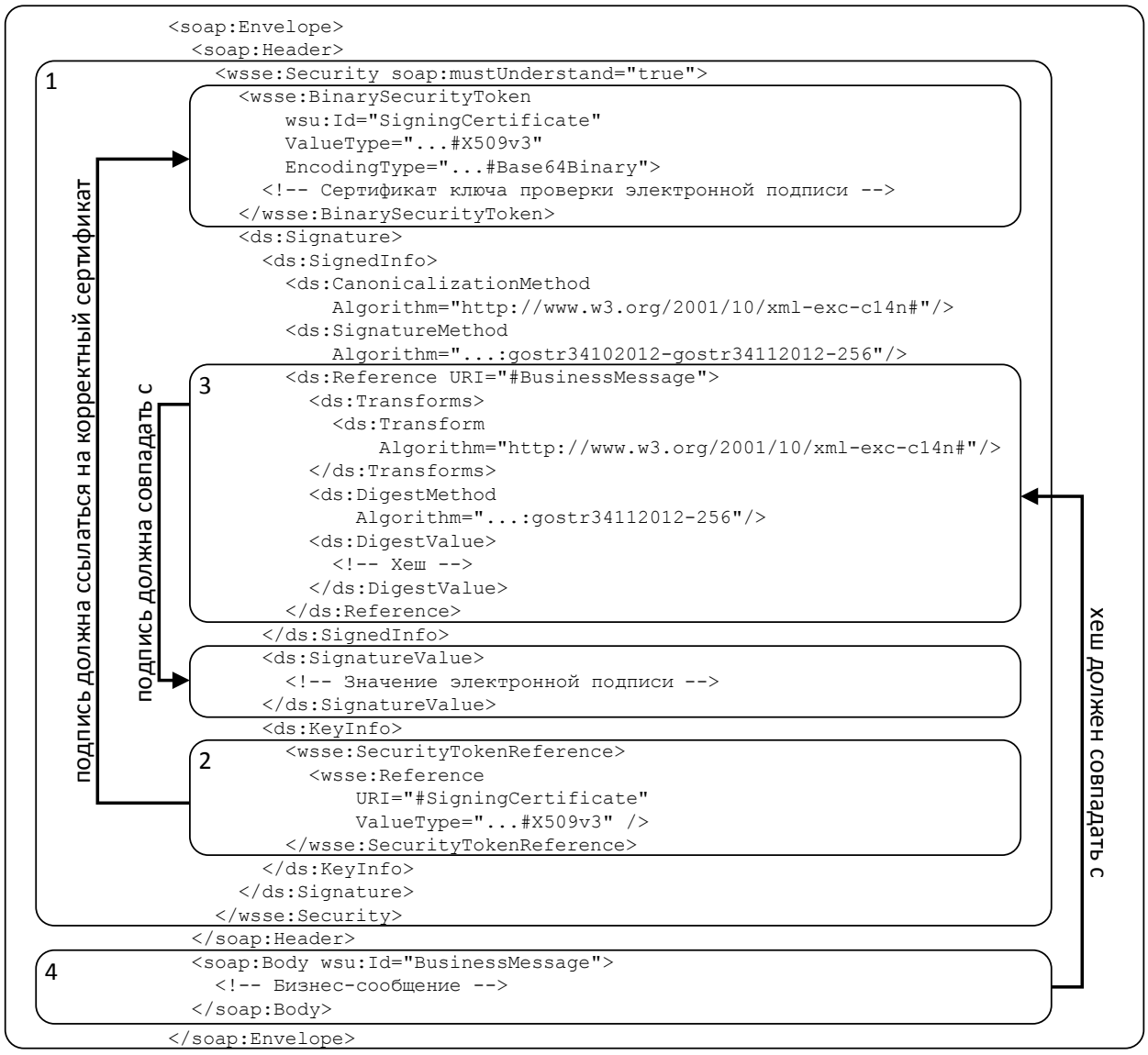

Рисунок А.2. Схема проверки электронной подписи

Приложение 2

XML схема бизнес-сообщения.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="urn:cbrf:iso:20022:msg"
    targetNamespace="urn:cbrf:iso:20022:msg"
    xmlns:bah="urn:iso:std:iso:20022:tech:xsd:head.001.001.02"
    elementFormDefault="qualified">
  <xs:import namespace="urn:iso:std:iso:20022:tech:xsd:head.001.001.02"/>
  <xs:element name="BusinessMessage" type="BusinessMessageType" />
  <xs:complexType name="BusinessMessageType">
    <xs:sequence>
      <xs:element ref="bah:AppHdr"/>
      <xs:any processContents="strict" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Приложение 3

Пример сообщения с электронной подписью.

```
<soap:Envelope
    xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header>
    <wsse:Security
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
        soap:mustUnderstand="true">
      <wsse:BinarySecurityToken
          wsu:Id="SigningCertificate"
          ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
x509-token-profile-1.0#X509v3"
          EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
soap-message-security-1.0#Base64Binary">
        <!-- Сертификат ключа проверки электронной подписи -->
      </wsse:BinarySecurityToken>
      <ds:Signature>
        <ds:SignedInfo>
          <ds:CanonicalizationMethod
              Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
          <ds:SignatureMethod
              Algorithm=
    "urn:ietf:params:xml:ns:cpxmlsec:algorithms:gostr34102012-gostr34112012-256"/>
          <ds:Reference URI="#BusinessMessage">
            <ds:Transforms>
              <ds:Transform
                  Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
            </ds:Transforms>
            <ds:DigestMethod
                Algorithm=
    "urn:ietf:params:xml:ns:cpxmlsec:algorithms:gostr34112012-256"/>
            <ds:DigestValue>
              <!-- Хеш -->
            </ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>
          <!-- Значение электронной подписи -->
        </ds:SignatureValue>
        <ds:KeyInfo>
          <wsse:SecurityTokenReference>
            <wsse:Reference
                URI="#SigningCertificate"
                ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-x509-token-profile-1.0#X509v3" />
          </wsse:SecurityTokenReference>
        </ds:KeyInfo>
      </ds:Signature>
    </wsse:Security>
  </soap:Header>
  <soap:Body wsu:Id="BusinessMessage">
    <!-- Бизнес-сообщение -->
  </soap:Body>
</soap:Envelope>
```

Приложение 4

Пример зашифрованного сообщения.

```
<soap:Envelope
    xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header>
    <wsse:Security
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
        soap:mustUnderstand="true">
      <wsse:BinarySecurityToken
          wsu:Id="EncryptionCertificate"
          ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
x509-token-profile-1.0#X509v3"
          EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
soap-message-security-1.0#Base64Binary">
          <!-- Сертификат открытого ключа -->
      </wsse:BinarySecurityToken>
      <xenc:EncryptedKey>
        <xenc:EncryptionMethod
            Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
        <ds:KeyInfo>
          <wsse:SecurityTokenReference>
            <wsse:Reference
                URI="#EncryptionCertificate"
                ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-x509-token-profile-1.0#X509v3" />
          </wsse:SecurityTokenReference>
        </ds:KeyInfo>
        <xenc:ReferenceList>
          <xenc:DataReference URI="#EncryptedMessage"/>
        </xenc:ReferenceList>
      </xenc:EncryptedKey>
    </wsse:Security>
  </soap:Header>
  <soap:Body>
    <xenc:EncryptedData Id="EncryptedMessage">
      <xenc:EncryptionMethod
          Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
      <xenc:CipherData>
        <xenc:CipherValue>
          <!-- Зашифрованное сообщение -->
        </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </soap:Body>
</soap:Envelope>
```