127018, Москва, ул. Сущёвский вал, д. 18
Телефон: +7 (495) 995 4820
Факс: +7 (495) 995 4820
https://www.CryptoPro.ru
E-mail: info@CryptoPro.ru

УТВЕРЖДЕНЫ

ЖТЯИ.00118-01 95 01-ЛУ

Средство Криптографической Защиты Информации	Программный комплекс «КриптоПро Ключ» версия 1.0 Правила пользования

ЖТЯИ.00118-01 95 01

Листов 79

2025 г.

СОДЕРЖАНИЕ

Аннотация 4
Перечень используемых терминов и сокращений 5
1 Назначение ПК «КриптоПро Ключ» и его основные характеристики 7
1.1 Назначение и функции СКЗИ 7
1.2 Защищаемая информация 8
1.3 Варианты использования СКЗИ 8
1.4 Исполнения СКЗИ 9
1.5 Реализуемые криптографические алгоритмы и протоколы 9
1.6 Интерфейсы СКЗИ 11
1.6.1 Поддерживаемые форматы документов 12
2 Ключевая система и ключевые документы 14
2.1 Типы и сроки действия ключей 14
2.2 Ключевые носители 19
2.3 Размеры ключей 19
2.4 Управление ключевой информацией 20
2.4.1 Формирование, ввод и смена ключевой информации 20
2.4.2 Контроль использования ключей 21
2.4.3 Использование и хранение ключевых носителей 21
2.4.4 Резервирование ключевой информации 23
2.4.5 Компрометация ключевой информации 23
2.4.6 Уничтожение ключевой информации 23
3 Порядок распространения и учета СКЗИ 24
3.1 Способы передачи и хранения СКЗИ 24
3.2 Поэкземплярный учет СКЗИ 26
4 Требования по обеспечению безопасности при вводе СКЗИ в эксплуатацию 27
4.1 Требования к встраиванию СКЗИ в прикладные системы и к проведению исследований СФ СКЗИ 27
4.1.1 Применение СКЗИ без исследований СФ 27
4.1.2 Применение СКЗИ с проведением исследований по оценке влияния 28
4.1.3 Применение СКЗИ с проведением тематических исследований 29
4.1.4 Требования при встраивании СКЗИ в прикладные системы 29

4.2 Требования по размещению 33
4.3 Требования к персоналу, обслуживающему СКЗИ 35
4.4 Инициализация и ввод СКЗИ в эксплуатацию 36
4.4.1 Требования к установке СКЗИ, общесистемного и специального ПО 36
4.4.2 Установка СКЗИ 38
4.4.3 Настройки и параметры СКЗИ 39
4.4.4 Ввод СКЗИ в эксплуатацию 39
5 Требования по обеспечению безопасности при эксплуатации СКЗИ 40
5.1 Общие требования по защите от НСД 40
5.2 Требования к аутентификации и разграничению доступа 40
5.3 Требования по обеспечению целостности СКЗИ 42
5.4 Порядок обеспечения работоспособности СКЗИ 43
5.4.1 Ограничение срока непрерывного функционирования СКЗИ 43
5.4.2 Журналирование и аудит 44
5.4.3 Восстановление работоспособности СКЗИ 44
6 Требования по обеспечению безопасности при выводе СКЗИ из эксплуатации и передаче в ремонт 45
6.1 Ремонт СКЗИ 45
6.2 Вывод СКЗИ из эксплуатации 45
Приложение 1. Перечень методов интерфейса REST API серверных компонентов, использование которых при разработке систем на основе ПК «КриптоПро Ключ» возможно без дополнительных тематических исследований 46
Конечная точка health 46
Сервис Подписи 46
Сервис Ключ Lite 48
Сервис Аудита 50
Сервис Обработки Документов 50
Центр Идентификации 51
Приложение 2. Перечень методов интерфейсов фреймворков «КриптоПро Ключ SDK» и «КриптоКлюч SDK», использование которых при разработке систем на основе ПК «КриптоПро Ключ» возможно без дополнительных тематических исследований 61
КриптоПро Ключ SDK 61
КриптоКлюч SDK 68
Приложение 3. Перечень вызовов, использование которых для реализации TLS-соединения с одно- и двусторонней аутентификацией при разработке систем на основе ПК «КриптоПро Ключ» возможно без дополнительных тематических исследований 74

Аннотация

Данный документ содержит правила пользования СКЗИ «Программный комплекс «КриптоПро Ключ» версия 1.0» (далее — ПК «КриптоПро Ключ», СКЗИ), включая описание состава, назначения и основных характеристик СКЗИ, порядок распространения и эксплуатации СКЗИ, требования по обеспечению безопасности при вводе в эксплуатацию, использовании и выводе СКЗИ из эксплуатации, а также требования к встраиванию СКЗИ в прикладные системы.

Документ предназначен для администраторов (администраторов информационной безопасности), осуществляющих установку, обслуживание и контроль за соблюдением требований к эксплуатации СКЗИ, для администраторов серверов, сетевых ресурсов предприятия и других работников службы информационной безопасности, осуществляющих настройку рабочих мест для работы со средствами СКЗИ, а также для непосредственных пользователей СКЗИ.

Инструкции администраторам и пользователям различных ИС, использующих ПК «КриптоПро Ключ», должны разрабатываться с учетом требований настоящих Правил пользования.

Выполнение СКЗИ заявленных в эксплуатационной документации функций и соответствие СКЗИ заявленным характеристикам и требованиям по информационной безопасности, предъявляемым ФСБ России, гарантируется при выполнении требований, изложенных в настоящем документе и эксплуатационной документации, входящей в комплект поставки СКЗИ. Требования настоящих Правил пользования и эксплуатационной документации могут уточняться с учетом модели угроз и нарушителя ИС, в которой применяется СКЗИ, при проведении исследований ИС.

Перечень используемых терминов и сокращений

CRL	Список отзыва сертификатов (Certificate Revocation List)
CSP	Криптопровайдер (Cryptographic Service Provider)
HSM	Аппаратный модуль системы безопасности (Hardware security module)
OATH	Набор алгоритмов аутентификации с использованием одноразовых паролей
OAuth	Открытый протокол авторизации, который позволяет предоставить третьей стороне ограниченный доступ к защищенным ресурсам пользователя без необходимости передавать ей (третьей стороне) логин и пароль (Open Authorization)
OCSP	Протокол получения статуса сертификата в реальном времени (Online Certificate Status Protocol)
OTP	Пароль, действительный только для одного сеанса аутентификации (OneTime Password)
REST	Архитектурный стиль построения распределенного приложения (Representational State Transfer)
SDK	Набор программных компонентов для использования в мобильных приложениях (Software development kit)
TLS	Протокол защиты транспортного уровня (Transport Layer Security)
URL	Единый указатель ресурсов (Uniform Resource Locator)
АРМ	Автоматизированное рабочее место
БД	База данных
ДСЧ	Датчик случайных чисел
ЗПС	Замкнутая программная среда
ИС	Информационная система
НСД	Несанкционированный доступ
МП	Мобильное приложение
МУ	Мобильное устройство
МЭ	Межсетевой экран

ОС	Операционная система
ПАК	Программно-аппаратный комплекс
ПАКМ	Программно-аппаратный криптографический модуль
ПО	Программное обеспечение
СВТ	Средства вычислительной техники
СКЗИ	Средство криптографической защиты информации
СОД	Сервис Обработки Документов
СУБД	Система управления базой данных
СФ	Среда функционирования СКЗИ
УЗ	Учетная запись
УЦ	Удостоверяющий Центр
ФКН	Функциональный ключевой носитель
ЦИ	Центр Идентификации
ЭП	Электронная подпись

1 Назначение ПК «КриптоПро Ключ» и его основные характеристики

1.1 Назначение и функции СКЗИ

ПК «КриптоПро Ключ» представляет собой клиент-серверное решение, предназначенное для предоставления пользователям ИС возможности совершения криптографических операций при различных вариантах хранения и защиты ключей пользователей.

ПК «КриптоПро Ключ» предназначен для:

1. генерации, управления, защищенного хранения и удаления ключевой информации;
2. генерации, распределенного хранения и управления ключевой информацией, распределенного создания ЭП, реализуемым в соответствии с протоколами схемы DKSSP;
3. шифрования и расшифрования данных, вычисления имитовставки;
4. создания ЭП в форматах CMS, CAdES, XMLDSig, XAdES, PAdES;
5. аутентификации пользователей;
6. аутентификации и обеспечения конфиденциальности передаваемых сообщений с использованием протокола TLS;
7. визуализации документов;
8. работы с УЗ пользователей (регистрация и удаление учетных записей, ведение реестра пользователей, оповещение пользователей);
9. управления сертификатами ключей проверки ЭП и открытых ключей обмена.

ПК «КриптоПро Ключ» обеспечивает выполнение следующих основных функций:

- функции по работе с УЗ Пользователей:
    - регистрация УЗ Пользователей (с учётом различных способов хранения и защиты ключей);
    - удаление УЗ Пользователей;
    - ведение реестра зарегистрированных Пользователей;
    - оповещение Пользователей о необходимости проведения операций и о результатах совершённых операций;
- функции, связанные с выполнением Пользователями криптографических операций:

- генерация ключей ЭП, ключей проверки ЭП, закрытых и открытых ключей шифрования, вспомогательных ключей для проведения криптографических операций, а также формирование запросов на сертификаты;
- аутентификация Пользователей;
- создание ЭП документов;
- шифрование/расшифрование документов;
- подтверждение проведения операции;
- визуализация документов перед выполнением операции с документом;

- функции по работе с ИС:
  - регистрация ИС;
  - авторизация ИС;
  - получение документов из ИС;
  - отправка документов в ИС;
- аудит событий, связанных с эксплуатацией программного комплекса.

1.2 Защищаемая информация

СКЗИ предназначено для защиты информации, не содержащей сведений, составляющих государственную тайну.

Допускается использование СКЗИ для криптографической защиты персональных данных[1].

Запрещено использование СКЗИ для защиты речевой информации.

1.3 Варианты использования СКЗИ

ПК «КриптоПро Ключ» может выступать как в качестве готового к применению комплекса, так и в качестве платформы для построения на его основе специализированных защищенных автоматизированных систем, программных, программно-аппаратных решений в области обеспечения информационной безопасности, основанных на применении российских криптографических алгоритмов.

Встраивание СКЗИ и проведение исследований среды функционирования СКЗИ на выполнение предъявленных к СКЗИ требований должны выполняться в соответствии с разд. 4.1.

[1] С учетом положений п. 4.1.4

1.4 Исполнения СКЗИ

ПК «КриптоПро Ключ» включает серверную и клиентскую части, разделенные на исполнения. Исполнение 1 содержит серверные компоненты СКЗИ, исполнения 2-5 — клиентские компоненты.

Исполнение	*Краткий состав*	*Класс защиты*
	Серверные компоненты	
Исполнение 1	ПАКМ «КриптоПро HSM», серверные программные компоненты ПК «КриптоПро Ключ»	**КС3**
	Клиентские компоненты	
Исполнение 2	МП «КриптоПро Ключ» и МП «КриптоКлюч», «КриптоПро Ключ SDK» и «КриптоКлюч SDK»	**КС1**
Исполнение 3	СКЗИ «КриптоПро CSP» (совместно с ПО «КриптоПро ЭЦП Browser plug-in») или СКЗИ «КриптоПро JCP»	**КС1**
Исполнение 4	СКЗИ «КриптоПро CSP» (совместно с ПО «КриптоПро ЭЦП Browser plug-in»)	**КС2**
Исполнение 5	СКЗИ «КриптоПро CSP» (совместно с ПО «КриптоПро ЭЦП Browser plug-in»)	**КС3**

С одним серверным компонентом одновременно могут функционировать различные клиентские компоненты. При этом класс защиты при подключении по протоколу TLS к серверной стороне определяется классом защиты клиентского компонента.

Подробное описание комплектации каждого исполнения приведено в разделе 3 документа «ЖТЯИ.00118-01 30 01. КриптоПро Ключ. Формуляр» (далее — Формуляр).

1.5 Реализуемые криптографические алгоритмы и протоколы

ПК «КриптоПро Ключ» реализует следующие криптографические алгоритмы и протоколы:

- формирование ЭП в соответствии с **ГОСТ Р 34.10-2012 (ГОСТ 34.10-2018)** «Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи»;
- зашифрование/расшифрование данных и вычисление имитовставки в соответствии с **ГОСТ Р 34.12-2015 (ГОСТ 34.12-2018)** «Информационная технология. Криптографическая защита информации. Блочные шифры», **ГОСТ Р 34.13-2015 (ГОСТ 34.13-2018)** «Информационная технология. Криптографическая защита

информации. Режимы работы блочных шифров», **ГОСТ 28147-89** «Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования», **Р 1323565.1.026–2019** «Информационная технология. Криптографическая защита информации. Режимы работы блочных шифров, реализующие аутентифицированное шифрование»[2];

- выработка значения хэш-функции в соответствии с **ГОСТ Р 34.11-2012 (ГОСТ 34.11-2018)** «Информационная технология. Криптографическая защита информации. Функция хэширования»;

- диверсификация ключевого материала, согласование ключей, вычисление кода аутентификации сообщения HMAC в соответствии с **Р 50.1.113–2016** «Информационная технология. Криптографическая защита информации. Криптографические алгоритмы, сопутствующие применению алгоритмов электронной цифровой подписи и функции хэширования»;

- экспорт и импорт ключей, режим шифрования в соответствии с **Р 1323565.1.017–2018** «Информационная технология. Криптографическая защита информации. Криптографические алгоритмы, сопутствующие применению алгоритмов блочного шифрования»;

- выработка ключа из пароля в соответствии с **Р 1323565.1.040-2022** «Информационная технология. Криптографическая защита информации. Парольная защита ключевой информации»;

- формирование защищенных сообщений в формате CMS в соответствии с **МР 26.2.002-2013** «Информационная технология. Криптографическая защита информации. Использование алгоритмов ГОСТ 28147-89, ГОСТ Р 34.10 и ГОСТ Р 34.11 в криптографических сообщениях формата CMS», **Р 1323565.1.025–2019** «Информационная технология. Криптографическая защита информации. Форматы сообщений, защищенных криптографическими методами»;

- протокол TLS с использованием российских криптографических алгоритмов в соответствии с **Р 1323565.1.020-2020** «Информационная технология. Криптографическая защита информации. Использование российских криптографических алгоритмов в протоколе безопасности транспортного уровня (TLS 1.2)» и **Р 1323565.1.030-2020** «Информационная технология. Криптографическая защита информации. Использование криптографических алгоритмов в протоколе безопасности транспортного уровня (TLS 1.3)»;

[2] По умолчанию в СКЗИ ПК «КриптоПро Ключ» используются ГОСТ Р 34.12-2015 и ГОСТ Р 34.13-2015. При необходимости использования ГОСТ 28147-89 требуется обоснование при проведении работ по оценке влияния (см. 4.1.2).

- протокол TSP (в качестве клиента) в соответствии с **Р 1323565.1.044–2022** «Информационная технология. Криптографическая защита информации. Использование российских криптографических алгоритмов в протоколе штампов времени (TSP)»;

- протокол OCSP (в качестве клиента) в соответствии с **МР 26.2.004–2023** «Информационная технология. Криптографическая защита информации. Использование российских криптографических алгоритмов в протоколе получения актуальных статусов сертификатов (OCSP)»;

- протоколы схемы DKSSP: протокол *Setup* инициализации защиты ключевой информации на устройстве, протокол *GetCertRequest* формирования запроса на сертификат, протокол *KGen* генерации ключевой пары подписи, протокол *Sign* формирования подписи под сообщением;

- протоколы OpenID Connect в соответствии с **МР 26.2.002-2024** «Информационная технология. Криптографическая защита информации. Использование российских криптографических алгоритмов в протоколах OpenID Connect», ЕСИА в соответствии с Методическими рекомендациями по использованию ЕСИА (версия 3.43, 2023), OAuth 2.0 в соответствии с **RFC 6749** «The OAuth 2.0 Authorization Framework», OIDF «OAuth 2.0 Multiple Response Types», OIDF «OAuth 2.0 Form Post Response Mode», TokenExchange в соответствии с **RFC 8693** «OAuth 2.0 Token Exchange».

1.6 Интерфейсы СКЗИ

ПК «КриптоПро Ключ» предоставляет следующие интерфейсы для взаимодействия с пользователями:

1. Графические интерфейсы:
    - веб-интерфейс Пользователя;
    - мобильные приложения «КриптоПро Ключ» и «КриптоКлюч»;
2. Интерфейсы для встраивания:
    - REST API;
    - cpcurl, JSSE (установка TLS-соединения);
    - фреймворки «КриптоПро Ключ SDK» и «КриптоКлюч SDK» для встраивания в мобильные приложения.

Порядок использования веб-интерфейса Пользователя описан в документе «ЖТЯИ.00118-01 93 01. Руководство Пользователя».

Порядок использования интерфейсов для встраивания описан в разделе 4.1.

1.6.1 Поддерживаемые форматы документов

Для визуализации через **СОД** реализована поддержка следующих форматов документов:

- Документы Word: DOC, DOCX, DOCM (документ Word с поддержкой макросов);
- Шаблоны Word: DOT, DOTX, DOTM (шаблон Word с поддержкой макросов);
- Форматы Word на основе XML: FlatOpc (XML-структура для документов Word), FlatOpcMacroEnabled (документ Word с поддержкой макросов в формате FlatOpc), FlatOpcTemplate (шаблон Word в формате FlatOpc), FlatOpcTemplateMacroEnabled (шаблон Word с поддержкой макросов в формате FlatOpc), WordML;
- TXT — текстовый документ;
- PDF — межплатформенный открытый формат электронных документов;
- BMP, GIF, JPEG, JPG, PNG, TIFF — форматы графических изображений;
- OTT, ODT — шаблон текстового документа OpenDocument (OpenOffice/LibreOffice);
- OOXML стандарт Office Open XML.2;
- RTF — формат для обмена документами;
- XML — расширяемый язык разметки;
- HTML/XHTML — форматы текстовой разметки веб-страниц;
- MHTML — веб-архив документа, включающий HTML-код и ресурсы (например, изображения).

Для визуализации с использованием **МП** реализована поддержка следующих форматов документов:

1. МП под управлением ОС iOS:

- Документы Word: DOC, DOCX;
- TXT — текстовый документ;
- PDF — межплатформенный открытый формат электронных документов;
- BMP, GIF, JPEG, JPG, PNG, TIFF — форматы графических изображений;
- RTF — формат для обмена документами;
- XML — расширяемый язык разметки;
- JSON — это текстовый формат обмена данными;
- HTM, HTML — форматы текстовой разметки веб-страниц;
- CSV — форматы табличных данных.

2. МП под управлением ОС Android:

- Документы Word: DOC, DOCX;
- TXT — текстовый документ;
- PDF — межплатформенный открытый формат электронных документов;
- BMP, JPEG, JPG, PNG — форматы графических изображений;
- XML — расширяемый язык разметки;
- JSON — это текстовый формат обмена данными;
- CSV — форматы табличных данных.

2 Ключевая система и ключевые документы

2.1 Типы и сроки действия ключей

Ключевая система ПК «КриптоПро Ключ» содержит следующие типы ключей в зависимости от их характеристик:

Характеристика	Типы ключей	Описание
1. Срок действия	Долговременные	Хранятся на компонентах ПК и имеют следующие сроки действия: • 1 год и 3 месяца — при хранении в извлекаемом/экспортируемом виде, • 3 года — при хранении в ФКН или в ПАКМ «КриптоПро HSM» в неизвлекаемом виде; После истечения строка действия вырабатываются новые значения ключей тем же способом, каким была выполнена выработка предыдущих значений.
	Эфемерные	Вырабатываются и удаляются в процессе выполнения протокола или алгоритма, входящего в состав СКЗИ.
2. Вид криптосистемы	Асимметричные	Ключевые пары (ключ ЭП/ключ проверки ЭП или закрытый/открытый ключ)
	Симметричные	Ключи аутентификации, ключи защиты данных/ шифрования ключей
3. Способ выработки	Мастер-ключи	Генерируются с помощью ДСЧ, реализованного в ПАКМ «КриптоПро HSM» или СКЗИ «КриптоПро CSP»/«КриптоПро JCP», входящего в состав компонентов ПК.
	Производные	Вырабатываются из мастер-ключей с помощью функции выработки производных ключей.

Полный перечень долговременных ключей и сертификатов, используемых в ПК «КриптоПро Ключ», с описанием назначения, сроков действия и смены приведен в таблице ниже:

Ключи серверного компонента СКЗИ			
Мастер-ключ аутентификации устройства пользователя (K^1_{master})	Используется для выработки ключей аутентификации устройства пользователя (K_{HMAC}) и проверки значения HMAC в рамках аутентификации устройства пользователя. Применяется во всех взаимодействиях между клиентским и серверными компонентами. Генерируется в момент настройки СКЗИ администратором с помощью ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Симм	3 года
Мастер-ключ аутентификации пользователя (K^2_{master})	Используется для выработки ключей аутентификации пользователя (K_{auth}) и проверки значения HMAC в рамках аутентификации пользователя и проверки аутентичности направляемых пользователем запросов. Применяется во взаимодействиях между клиентским и серверными компонентами, требующих подтверждения пользователя. Генерируется в момент настройки СКЗИ администратором с помощью ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Симм	3 года
Мастер-ключ защиты данных сессий при выполнении интерактивных протоколов (K^3_{master})	Используется для выработки ключей шифрования параметров state и nonce в протоколе OpenID Connect. Требуется при взаимодействии с клиентами протокола OpenID Connect. Генерируется в момент настройки СКЗИ администратором с помощью ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Симм	3 года
Мастер-ключ выработки секретов сервера (K^{SK}_{master})	Используется для выработки ключей защиты данных на устройстве пользователя. Генерируется в момент настройки СКЗИ администратором с помощью ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Симм	3 года
Ключевая пара подтверждения (формирования квитанции) (d^{conf}, Q^{conf})	Используется в схеме DKSSP для формирования квитанции, являющейся подтверждением получения ЭП документа или открытого текста от клиентского компонента. Генерируется ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Асимм	3 года

Ключевая пара дополнительной защиты ключа ЭП пользователя *(mask, Q)*	Используется для обеспечения дополнительной защиты ключа ЭП пользователя на сервере (исполнение 2 в режиме распределенного хранения ключей ЭП). Представляет собой вектор защиты ключа ЭП и соответствующий ключ проверки ЭП. Генерируется ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде. Используется в схеме DKSSP при формировании ЭП документа.	Асимм	1 год 3 месяца
Ключевые пары безопасности (TLS) (d^{TLS}, Q^{TLS})	Ключи и сертификаты TLS сервера и TLS клиента для серверных компонентов используются для защиты каналов по протоколу TLS между ИС и серверными компонентами, между различными серверными компонентами, а также между серверными и клиентским компонентами. Генерируются в момент настройки СКЗИ администратором с помощью СКЗИ «КриптоПро CSP» и хранятся на серверах серверных компонентов ПК.	Асимм	1 год 3 месяца
Ключевая пара обеспечения целостности журнала аудита (d^{audit}, Q^{audit})	Используется для обеспечения целостности журнала аудита. Генерируется с помощью ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Асимм	3 года
Ключевая пара защиты базы данных (d^{bd}, Q^{bd})	Используется для защиты базы данных. Генерируется с помощью ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Асимм	3 года
Ключевая пара для подписи маркеров аутентификации пользователей (OAuth) (d^{auth}, Q^{auth})	Используется для подписи маркеров аутентификации пользователей по протоколу OAuth. Генерируется с помощью ПАКМ «КриптоПро HSM» и хранится в неизвлекаемом виде.	Асимм	3 года
Ключи клиентского компонента СКЗИ			
Ключ аутентификации устройства	Используется для аутентификации мобильного устройства пользователя. Применяется во всех взаимодействиях между	Симм	Подлежит плановой смене 1 раз в 1 год 3 месяца

пользователя (K_{HMAC})	клиентским (исполнение 2) и серверными компонентами. Вырабатывается из мастер-ключа K^1_{master} с помощью ПАКМ «КриптоПро HSM» в процессе привязки МП к УЗ пользователя.		
Ключ аутентификации пользователя (K_{auth})	Используется для подтверждения запросов пользователя на выполнение криптографических функций СКЗИ. Применяется во взаимодействиях между клиентским (исполнение 2) и серверными компонентами, требующих подтверждения пользователя. Вырабатывается из мастер-ключа K^2_{master} с помощью ПАКМ «КриптоПро HSM» в процессе привязки МП к УЗ пользователя.	Симм	Подлежит плановой смене 1 раз в 1 год 3 месяца
Мастер-ключ клиента (SK_C)	Используется для выработки ключа защиты данных на мобильном устройстве пользователя. Вырабатывается из мастер-ключа K^{SK}_{master} в взаимодействиях между клиентским (исполнение 2) и серверными компонентами, требующих подтверждения пользователя, и хранится на мобильном устройстве пользователя.	Симм	Подлежит плановой смене 1 раз в 1 год 3 месяца
Пин-код *pw*	Используется для выработки ключа защиты данных на мобильном устройстве пользователя. Генерируется с использованием «КриптоПро Ключ SDK»/«КриптоКлюч SDK» на МУ, выбирается и запоминается пользователем, не хранится. Должен соответствовать политике паролей (см. раздел 5.2).	—	6 месяцев
Пароль для аутентификации в веб-интерфейсе *password*	Используется для аутентификации пользователя с помощью веб-интерфейса. Генерируется ЦИ или задается пользователем, не хранится. Должен соответствовать политике паролей (см. раздел 5.2).	—	6 месяцев
Ключевая пара ЭП пользователя *(d^{mask}, Q)*	Используется при создании и проверке ЭП в случае хранения ключей с дополнительной защитой на сервере (исполнение 2 в режиме распределенного хранения ключей ЭП).	Асимм	1 год 3 месяца

	Генерируется СКЗИ «КриптоПро CSP» и хранится на МУ пользователя. Доступ к ключу обеспечивается за счет предъявления серверу пин-кода *pw*.		
Ключевая пара ЭП/шифрования пользователя *(d, Q)*	Используется при проведении операций с ЭП/шифрования в случае хранения ключей на АРМ пользователя (исполнения 3-5) или на МУ с автономной защитой (исполнение 2 в режиме автономного хранения ключей ЭП). Генерируется СКЗИ «КриптоПро CSP»/«КриптоПро JCP» и хранится на устройстве пользователя.	Асимм	Зависит от используемого ключевого носителя
Ключевая пара аутентификации пользователя	Ключ и сертификат аутентификации пользователя предназначены для обеспечения защиты по протоколу TLS (с аутентификацией клиента по сертификату) канала связи между ИС/веб-браузером Пользователя и серверными компонентами СКЗИ при HTTPS-доступе ИС/пользователей ПК к целевым функциям. Сертификат ключа аутентификации пользователя в поле EKU содержит стандартное расширение «Проверка подлинности клиента». Формируются клиентским компонентом (исполнения 3-5) или сторонними СКЗИ с последующим вводом ключей в ПК. Ключ и сертификат хранятся в контейнере ключа аутентификации на ключевом носителе, сертификат также хранится в профиле УЗ пользователя в БД пользователей ПК.	Асимм	1 год 3 месяца
Ключевая пара аутентификации Оператора	Ключ и сертификат аутентификации Оператора предназначены для обеспечения защиты по протоколу TLS (с аутентификацией клиента по сертификату) канала связи между АРМ Оператора и остальными серверными компонентами СКЗИ при HTTPS-доступе Оператора к функциям управления. Сертификат ключа аутентификации Оператора в поле EKU содержит стандартное расширение «Проверка подлинности клиента». Ключ и сертификат хранятся в контейнере ключа аутентификации на отчуждаемом ключевом носителе, сертификат также хранится в профиле УЗ оператора в БД пользователей ПК.		1 год 3 месяца

2.2 Ключевые носители

Варианты поддерживаемых ключевых носителей зависят от исполнения СКЗИ и компонента, входящего в это исполнение:

Исполнение 1		
Компонент	*Ключевой носитель*	*Режим*
ПАКМ «КриптоПро HSM»	В соответствии с эксплуатационной документацией на ПАКМ «КриптоПро HSM»	
Серверные программные компоненты ПК «КриптоПро Ключ» (ключи протокола TLS)	○ Реестр Windows ○ Раздел HDD/SSD	Пассивный
АРМ Оператора (ключевая пара аутентификации Оператора)	В соответствии с эксплуатационной документацией на используемую версию СКЗИ «КриптоПро CSP»	
Исполнение 2		
МП «КриптоПро Ключ»/ МП «КриптоКлюч»/ МП, разработанные на базе фреймворков «КриптоПро Ключ SDK» или «КриптоКлюч SDK»	○ Рутокен ЭЦП 3.0 ○ Рутокен ЭЦП 2.0 3000	ФКН с поддержкой SESPAKE/ ФКН без поддержки SESPAKE
	○ Рутокен ЭЦП 2.0, Рутокен ЭЦП 2.0 2100, Рутокен ЭЦП 2.0 2151	ФКН без поддержки SESPAKE
	○ Устройство iOS/Android	Пассивный
Исполнения 3-5		
СКЗИ «КриптоПро CSP»/ СКЗИ «КриптоПро JCP»	В соответствии с эксплуатационной документацией на используемую версию СКЗИ «КриптоПро CSP» / СКЗИ «КриптоПро JCP»	

2.3 Размеры ключей

Размеры ключей электронной подписи:	
ключ электронной подписи	256 бит или 512 бит
ключ проверки электронной подписи	512 бит или 1024 бита
Размеры ключей, используемых при шифровании:	

закрытый ключ	256 бит или 512 бит
открытый ключ	512 бит или 1024 бита
симметричный ключ	256 бит

2.4 Управление ключевой информацией

2.4.1 Формирование, ввод и смена ключевой информации

Для выработки ключевой и криптографически опасной информации в ПК «КриптоПро Ключ» используются ПДСЧ, реализованные в ПАКМ «КриптоПро HSM», СКЗИ «КриптоПро CSP» и СКЗИ «КриптоПро JCP», входящих в комплектацию ПК «КриптоПро Ключ». В качестве источника инициализирующей последовательности ПДСЧ применяются источники, поддерживаемые в работе ПАКМ «КриптоПро HSM», СКЗИ «КриптоПро CSP» и СКЗИ «КриптоПро JCP» в соответствии с эксплуатационной документацией на указанные СКЗИ.

Помимо ключевой информации, вырабатываемой с использованием ПАКМ «КриптоПро HSM», СКЗИ «КриптоПро CSP» и СКЗИ «КриптоПро JCP», ПК «КриптоПро Ключ» также поддерживает использование сторонних (полученных без использования ПК «КриптоПро Ключ») и/или ранее сгенерированных с использованием ПК «КриптоПро Ключ» ключевых пар для шифрования и формирования ЭП (для ключей с автономной защитой и ключевых пар безопасности).

Для исполнения 2 СКЗИ при использовании Пользователем МП ввод ключевой информации реализован с помощью средств встроенного фреймворка «КриптоПро Ключ SDK» или «КриптоКлюч SDK». Для исполнений 3-5 СКЗИ ввод ключевой информации на стационарное устройство Пользователя (АРМ) реализован средствами установленного СКЗИ «КриптоПро CSP» и СКЗИ «КриптоПро JCP».

Для ввода ключевой информации она должна быть представлена в формате данных, который поддерживается СКЗИ. Для исполнения 2 поддерживается ввод с отчуждаемого ключевого носителя (внутренний формат ФКН, только для режима автономного хранения ключей ЭП) и из файловой системы (сертификат в формате X.509), для исполнений 3-5 — форматы pfx-контейнера, внутренний формат ФКН при хранении на них ключей в неизвлекаемом виде или формат ключевого контейнера СКЗИ «КриптоПро CSP»/«КриптоПро JCP».

При хранении Пользовательских ключей на стационарном устройстве (исполнения 3-5 ПК «КриптоПро Ключ») смена ключевой информации реализована средствами СКЗИ «КриптоПро CSP»/«КриптоПро JCP». При хранении ключей Пользователей в МП (исполнение 2 ПК «КриптоПро Ключ») смена ключей защиты ключей ЭП/шифрования реализована с помощью средств встроенного фреймворка «КриптоПро Ключ SDK» или «КриптоКлюч SDK».

Формирование и управление сертификатами открытых ключей производится УЦ. Взаимодействие с компонентами УЦ при управлении ключами должно осуществляться в соответствии с Регламентом УЦ.

При формировании запроса на сертификат Исполнение 2 (МП/SDK в режиме распределенного хранения ключей ЭП) не включает в состав данного запроса сведений о режиме генерации и хранения ключей, т.е. не обеспечивает возможность верификации со стороны УЦ участия Серверных компонентов в процессе создания ключевой пары, для открытого ключа которой запрашивается сертификат.

2.4.2 Контроль использования ключей

В МП, разрабатываемых на базе фреймворков «КриптоПро Ключ SDK» или «КриптоКлюч SDK», входящих в исполнение 2 ПК «КриптоПро Ключ», должен быть включён режим усиленного контроля использования ключей в соответствии с эксплуатационной документацией на используемый SDK.

В МП «КриптоПро Ключ» и МП «КриптоКлюч», входящих в исполнение 2 ПК «КриптоПро Ключ», данный режим используется без возможности выключения.

В клиентских компонентах исполнений 3-5 ПК «КриптоПро Ключ», включающих в состав СКЗИ «КриптоПро CSP» или СКЗИ «КриптоПро JCP», должен быть включён режим усиленного контроля использования ключей в соответствии с эксплуатационной документацией на используемую версию СКЗИ «КриптоПро CSP» или СКЗИ «КриптоПро JCP».

Использование ПК «КриптоПро Ключ» без включения режима усиленного контроля использования ключей разрешается исключительно в тестовых целях.

2.4.3 Использование и хранение ключевых носителей

При использовании любых типов ключевых носителей помимо требований эксплуатационной документации на СКЗИ необходимо руководствоваться эксплуатационной документацией на сами носители.

При хранении ключей и ключевых носителей, используемых в исполнениях 1 и 3-5 ПК «КриптоПро Ключ», следует руководствоваться требованиями, приведенными в эксплуатационной документации на СКЗИ, входящее в комплектацию исполнения ПК «КриптоПро Ключ» (в частности, для исполнения 1 — в разделе «Хранение и использование ключей» Правил пользования на ПАКМ «КриптоПро HSM», для исполнений 3-5 — в разделе «Хранение ключей и ключевых носителей» Правил пользования на СКЗИ «КриптоПро CSP» и разделе «Хранение ключевых носителей» Правил пользования на СКЗИ «КриптоПро JCP»).

При хранении ключей и ключевых носителей, используемых в исполнении 2 ПК «КриптоПро Ключ», должны выполняться следующие требования:

- Необходимо обеспечить невозможность доступа к отчуждаемым ключевым носителям не допущенных к ним лиц.

- Запрещается оставлять без контроля МУ с установленным СКЗИ после ввода ключевой информации.

- Пользователь несёт персональную ответственность за хранение личных ключевых носителей. В случае централизованного хранения ключевых носителей в организации, эксплуатирующей СКЗИ, Администратор безопасности несёт персональную ответственность за хранение личных ключевых носителей пользователей.

- Хранение ключей на несъемных носителях (МУ с ОС Apple iOS/Android) допускается только при условии распространения на носитель требований по обращению с ключевыми носителями (в том числе и после удаления ключей).

- Необходимо использовать парольную защиту, если не оговорено иное.

- Запрещено разглашать содержимое носителей ключевой информации, выводить ключевую информацию на дисплей, принтер и иные средства отображения информации (за исключением случаев, предусмотренных эксплуатационной документацией на СКЗИ).

- При необходимости передачи ключевого носителя постороннему лицу информацию с носителя необходимо гарантированно удалить.

При использовании ключевых носителей, поддерживаемых СКЗИ, запрещается:

- использовать ключевые носители в режимах, не предусмотренных функционированием СКЗИ;

- вставлять ключевой носитель в устройство считывания в режимах, не предусмотренных штатным режимом использования ключевого носителя;

- записывать на ключевые носители постороннюю информацию;

- использовать бывшие в работе ключевые носители для записи новой информации без предварительного уничтожения на них ключевой информации, подлежащей уничтожению, средствами СКЗИ.

При использовании *пассивных ключевых носителей и ФКН без поддержки SESPAKE* необходимо обеспечить выполнение следующих условий:

- подключение съемных носителей должно осуществляться непосредственно к считывателю (считывателю смарт-карт, USB-портам и т.п.), с обеспечением отсутствия канала связи между носителем и СКЗИ, в котором может действовать нарушитель;

- при конструктивном исполнении считывателя ключевого носителя с кабелем необходимо обеспечить нахождение считывателя и кабеля на той же контролируемой территории, что и ПЭВМ, и отсутствие доступа нарушителя к ним.

- использование ключевых носителей в качестве пассивного хранилища ключевой информации с передачей данных по бесконтактному интерфейсу не допускается, кроме

использования носителей, обеспечивающих защиту передаваемых по данному интерфейсу ключей (и при наличии соответствующего заключения ФСБ России).

При невозможности выполнения указанных условий необходимо с учётом модели возможных угроз и нарушителя разработать организационно-технические мероприятия по защите взаимодействия носителя с СКЗИ с последующей оценкой таких мероприятий в рамках проведения соответствующих исследований.

2.4.4 Резервирование ключевой информации

В случае централизованного хранения ключевых носителей в организации, эксплуатирующей СКЗИ при определении регламента резервного копирования и хранения резервных копий ключевых носителей следует руководствоваться требованиями по организации процедуры резервного копирования и хранения резервных копий ключевых носителей, приведенными в эксплуатационной документации на СКЗИ, используемое в составе серверного или клиентского компонента ПК «КриптоПро Ключ» (в частности, разделом «Требования по организации процедуры резервного копирования и хранения резервных копий ключевых носителей» Правил пользования на СКЗИ «КриптоПро CSP»).

При эксплуатации СКЗИ запрещено осуществлять несанкционированное Администратором безопасности копирование ключевых носителей. Резервное копирование ключевых носителей в МП, входящих в исполнение 2 ПК «КриптоПро Ключ», не поддерживается.

2.4.5 Компрометация ключевой информации

При компрометации своего ключа Пользователь должен немедленно прекратить связь по сети с другими пользователями. При этом Пользователь (или Администратор безопасности) должен немедленно известить ЦР (УЦ) о компрометации ключа пользователя.

По факту компрометации должно быть проведено служебное расследование. Скомпрометированные ключи выводятся из действия.

Скомпрометированные ключи подлежат смене. После компрометации ключей Пользователь формирует новый закрытый ключ и запрос на сертификат.

2.4.6 Уничтожение ключевой информации

Выведенные из действия ключи ЭП/закрытые ключи (в том числе, по истечении срока действия или в случае компрометации ключа) должны быть уничтожены.

Ключи Пользователей могут быть удалены (уничтожены) следующими способами:

- Удаление ключа аутентификации/ключа ЭП в МП (исполнение 2 ПК «КриптоПро Ключ»);
- Удаление ключа ЭП средствами ПАКМ «КриптоПро HSM» или СКЗИ «КриптоПро CSP»/СКЗИ «КриптоПро JCP» (исполнения 1, 3-5 ПК «КриптоПро Ключ»).

3 Порядок распространения и учета СКЗИ

3.1 Способы передачи и хранения СКЗИ

ПК «КриптоПро Ключ» и комплект эксплуатационной документации к нему могут поставляться пользователю Уполномоченной организацией[3] следующими способами:

1) На физическом носителе — оптическом диске (CD, DVD).

Транспортировку носителя рекомендуется осуществлять в упаковке изготовителя СКЗИ с избеганием механических воздействий и влияния внешней среды на носитель. Хранение носителя должно осуществляться в упаковке (изготовителя или специальных кейсах/конвертах) в темном, прохладном, сухом месте (температура воздуха от +10°C до +20°C, относительная влажность воздуха от 20% до 65%) без резких колебаний температуры и влажности, без воздействий прямых солнечных лучей, грязи и пыли, химических паров/жидкостей, вдали от источников тепла и влаги.

При установке носителя СКЗИ на хранение соответствующая информация должна быть внесена в раздел «СВЕДЕНИЯ О ХРАНЕНИИ» Формуляра на СКЗИ.

2) Посредством загрузки через информационно-телекоммуникационные сети (в том числе Интернет).

В этих случаях для получения возможности загрузки установочных модулей СКЗИ и комплекта эксплуатационной документации пользователь направляет свои учётные данные Уполномоченной организации. Учётные данные могут быть направлены посредством заполнения специализированной регистрационной формы на сайте Уполномоченной организации.

После получения Уполномоченной организацией учётных данных пользователю предоставляется доступ на страницу загрузки установочных модулей СКЗИ и комплекта эксплуатационной документации. При загрузке пользователем установочных модулей СКЗИ и комплекта эксплуатационной документации Уполномоченной организацией присваивается учётный номер, идентифицирующий экземпляр СКЗИ, предоставленный пользователю.

На странице загрузки вместе с дистрибутивом и документацией размещается отделенная ЭП, для проверки которой необходимо использовать утилиту *cpverify*, полученную доверенным образом и содержащую ключ проверки данной ЭП.

3) В составе прикладной программы (для клиентских компонентов под управлением мобильных ОС iOS и Android).

[3] ООО «КРИПТО-ПРО» или его официальный дилер/дистрибьютор

В этом случае прикладная программа, содержащая СКЗИ, и комплект документации поставляются пользователю Уполномоченной организацией способом, определенным в документации на прикладную программу, например:

а) посредством загрузки прикладной программы в корпоративной сети;

б) посредством загрузки в сети Интернет через магазин приложений (App Store, Google Play и другие).

При необходимости активации СКЗИ в составе прикладной программы пользователь направляет свои учётные данные Уполномоченной организации. Учётные данные могут быть направлены посредством заполнения специализированной регистрационной формы на сайте Уполномоченной организации.

После получения Уполномоченной организацией учётных данных пользователю предоставляется лицензионный код. Лицензионный код может вводиться как в окне панели управления СКЗИ, так и устанавливаться в составе сертификата открытого ключа пользователя, а также его ввод может быть реализован средствами прикладной программы.

Разработчики программного обеспечения (в случае приложений для ОС iOS — одновременно с формированием ЭП дистрибутивов на зарубежных криптоалгоритмах по установленным компанией Apple процедурам) должны вычислять значения контрольных сумм дистрибутивов разрабатываемого продукта и ЭП этой контрольной суммы при помощи средства контроля целостности (*cpverify* или иного сертифицированного средства) в соответствии с «Руководством разработчика» (для используемых SDK и ОС). Данная контрольная сумма и её отделенная ЭП должны поставляться Уполномоченной организацией.

Документацией на прикладную программу также должна быть учтена необходимость проверки ЭП контрольной суммы, полученной от Уполномоченной организации, и проверки целостности дистрибутива прикладной программы с помощью полученной контрольной суммы до установки этого дистрибутива.

Установка СКЗИ может быть осуществлена только в случае подтверждения целостности полученных установочных модулей приложения (при наличии), модулей СКЗИ и эксплуатационной документации (подробнее см. п. 5.3).

Использование утилиты cpverify

1. Средство контроля целостности (*cpverify*) первоначально должно быть получено пользователем на физическом носителе в офисе Уполномоченной организации. Такая утилита считается полученной доверенным образом. Далее полученной доверенным образом признается очередная версия утилиты, полученная любым образом (например, скачанная с сайта https://cryptopro.ru/), при условии, что она была проверена другим экземпляром утилиты, полученным ранее доверенным образом, и проверка была успешной.
2. Ключ проверки ЭП, а также информация о нем (дата создания, алгоритм хэш-функции, идентификатор алгоритма подписи) записываются в исходный код утилиты на этапе сборки.
3. Контроль целостности дистрибутива СКЗИ и компонентов среды функционирования (СФ) СКЗИ обеспечивается при помощи утилиты *cpverify* в соответствии с Приложением 1 Правил пользования СКЗИ «КриптоПро CSP».

3.2 Поэкземплярный учет СКЗИ

Для всех приведённых выше способов передачи СКЗИ поэкземплярный учёт СКЗИ осуществляется в процессе приобретения пользователем экземпляра СКЗИ.

Поэкземплярный учёт СКЗИ проводится как изготовителем СКЗИ, так и официальными дилерами/дистрибьютерами.

Кроме того, в случае, когда для конечной пользовательской ИС поставляется набор неперсонифицированных экземпляров СКЗИ, разработчик СКЗИ фиксирует у себя количество предоставленных экземпляров СКЗИ, а администраторы ИС должны самостоятельно выполнять учёт каждого экземпляра СКЗИ, передаваемого пользователю.

Порядок поэкземплярного учета СКЗИ может уточняться с учетом условий эксплуатации ИС.

4 Требования по обеспечению безопасности при вводе СКЗИ в эксплуатацию

4.1 Требования к встраиванию СКЗИ в прикладные системы и к проведению исследований СФ СКЗИ

ПК «КриптоПро Ключ» может выступать как в качестве готового к применению комплекса, так и в качестве платформы для построения на его основе специализированных защищенных автоматизированных систем, программных, программно-аппаратных решений в области обеспечения информационной безопасности, основанных на применении российских криптографических алгоритмов.

Для обеспечения информационной безопасности рабочих мест и информационных систем, использующих СКЗИ, должны быть определены модель возможных угроз и нарушителя и выработана политика безопасности. В зависимости от модели возможных угроз и нарушителя определяется необходимый уровень защиты и, соответственно, необходимый класс СКЗИ.

Встраивание СКЗИ в СФ проводится в соответствии с порядком, определённым Положением ПКЗ-2005, организациями, имеющими соответствующие лицензии на указанные виды деятельности.

В зависимости от условий эксплуатации и от конкретной СФ, в составе которой применяется СКЗИ, могут существовать следующие варианты наличия или отсутствия необходимости проведения исследований СФ:

1. исследования СФ не требуются;
2. требуются исследования по оценке влияния СФ на СКЗИ;
3. требуется проведение тематических исследований СФ со встроенным СКЗИ как самостоятельного шифровального (криптографического) средства.

Перечень компонентов СФ, для которых необходимо проводить исследования, определяется исходя из архитектуры информационной системы и выполняемых функций.

Перечень работ, которые необходимо провести в рамках исследований, определяется в зависимости от конечной ИС и актуальной для неё модели угроз и нарушителя.

4.1.1 Применение СКЗИ без исследований СФ

Исследования СФ не требуются при выполнении одного из следующих условий:

1. применение СКЗИ и СФ не подпадает под случаи, приведённые в п. 3 Положения ПКЗ-2005, и иными нормативными документами не определены дополнительные условия применения шифровальных (криптографических) средств;

2. при использовании веб-интерфейсов, входящих в состав серверных компонентов СКЗИ при соблюдении правил настройки, изложенных в п. 4.1.4 настоящего документа и в «ЖТЯИ.00118-01 91 01. Руководство Администратора», а также при условии, что порядок использования клиентского компонента (СКЗИ «КриптоПро CSP» или СКЗИ «КриптоПро JCP») соответствует условиям применения СКЗИ без исследований СФ в соответствии с Правилами пользования на используемый клиентский компонент.

4.1.2 Применение СКЗИ с проведением исследований по оценке влияния

Проведение исследований по оценке влияния СФ на СКЗИ требуется при одновременном выполнении следующих условий:

1. применение СКЗИ и СФ подпадает под случаи, приведённые в п. 3 Положения ПКЗ-2005, и/или иными нормативными документами определены дополнительные условия применения шифровальных (криптографических) средств;

2. в отношении СКЗИ и/или СФ выполняется хотя бы одно из следующих условий:

- СКЗИ используется совместно со сторонними центрами идентификации для аутентификации пользователей СКЗИ (отличных от ЕСИА);
- в качестве СФ используются программные компоненты, осуществляющие вызовы (напрямую или опосредовано через различные программные интерфейсы) функций СКЗИ, приведённых в Приложениях 1-3;
- вариант использования клиентского компонента (СКЗИ «КриптоПро CSP» или «КриптоПро JCP») соответствует условиям применения СКЗИ с проведением исследований по оценке влияния в соответствии с Правилами пользования на используемый клиентский компонент.

Оценка влияния СФ СКЗИ на выполнение предъявленных к СКЗИ требований должна выполняться по Техническому заданию, согласованному с 8 Центром ФСБ России.

При проведении исследований по ОВ должны быть предусмотрены следующие направления исследований (включая, но не ограничиваясь):

- по исходным кодам и эксплуатационной документации на СФ осуществляется анализ выполнения требований и рекомендаций по встраиванию СКЗИ, изложенных в документации на СКЗИ, а также анализ полноты и корректности эксплуатационной документации на компоненты СФ, включая отражение в указанной документации организационно-технических мер по защите информации с использованием СКЗИ и правил обращения с ключевыми документами;
- проверка корректности реализации в СФ работы с криптографическими ключами;
- проверка корректности реализации в СФ обработки возвращаемых СКЗИ данных, ошибок, исключений и т.п.;

- проверка корректности работы криптографических функций в штатном и не штатном режимах и в условиях искаженных и/или отсутствующих объектов PKI;
- проверка корректности реализации механизмов контроля целостности СФ и СКЗИ;
- проверка корректности реализации механизма аутентификации;
- проверка отсутствия в СФ кода, позволяющего модифицировать содержимое памяти СКЗИ, осуществляющего доступ к произвольным областям памяти (в том числе к памяти других процессов), позволяющего повышать свои привилегии при выполнении, позволяющего формировать исполняемый код во время работы;
- проверка отсутствия в СФ известных уязвимостей, опубликованных в общедоступных источниках;
- проверка механизмов визуализации информации для пользователя (п.п. 8 и/или 9 Приложения 1 к Приказу ФСБ России от 27 декабря 2011 г. № 796 «Об утверждении Требований к средствам электронной подписи и Требований к средствам удостоверяющего центра»);
- в случае опосредованного использования функций программных интерфейсов серверных и клиентских компонентов СКЗИ из других программных интерфейсов, необходимо проведение исследований прохождения вызовов и передаваемых данных между вызываемым интерфейсом и поддерживаемыми СКЗИ интерфейсами (построение трасс вызовов).

4.1.3 Применение СКЗИ с проведением тематических исследований

Проведение тематических исследований СФ со встроенным СКЗИ как самостоятельного шифровального (криптографического) средства требуется при одновременном выполнении следующих условий:

1. применение СКЗИ и СФ подпадает под случаи, приведённые в п. 3 Положения ПКЗ-2005, или иными нормативными документами определены дополнительные условия применения шифровальных (криптографических) средств;

2. в качестве СФ используются программные компоненты, осуществляющие вызовы (напрямую или опосредовано через различные программные интерфейсы) функций СКЗИ, не приведённых в Приложениях 1-3 настоящего документа, или вариант использования клиентского компонента (СКЗИ «КриптоПро CSP» или «КриптоПро JCP») соответствует условиям применения СКЗИ с проведением тематических исследований в соответствии с Правилами пользования на используемый клиентский компонент.

4.1.4 Требования при встраивании СКЗИ в прикладные системы

При встраивании СКЗИ в прикладные системы (прикладное ПО, СФ) должны выполняться следующие требования:

1. Сертификаты открытых ключей (ключей проверки ЭП), используемые СКЗИ, следует выпускать УЦ, сертифицированным ФСБ России по классу защиты не ниже класса защиты используемого СКЗИ, с учетом модели угроз и нарушителя ИС, в которой применяется СКЗИ, а также с учетом нормативных документов, определяющих создание такой ИС.

При этом нормативные документы, определяющие создание такой ИС (включая Техническое задание на разработку и проведение исследований ИС или ее компонентов) могут устанавливать дополнительные требования к построению ключевой системы с проведением установленным порядком ее исследований.

2. При использовании функций СКЗИ должна быть обеспечена актуальность сертификатов открытых ключей/сертификатов ключей проверки ЭП (с использованием УЦ, построенного на базе Средств УЦ, сертифицированных ФСБ России и правил обработки объектов PKI в СФ).

Актуальность сертификатов открытых ключей/сертификатов ключей проверки ЭП (далее — сертификатов) должна обеспечиваться:

- проверкой корректности ЭП сертификатов из цепочки сертификатов;
- проверкой сроков действия сертификатов из цепочки сертификатов;
- проверкой области использования сертификата;
- проверкой сертификатов из цепочки сертификатов на отозванность при помощи списков отозванных (аннулированных) сертификатов (CRL) или с помощью OCSP-службы;
- проверкой сроков действия CRL (ответа OCSP-службы);
- проверкой корректности ЭП CRL (ЭП ответа OCSP-службы);
- проверкой наличия CRL;
- проверкой соответствия сертификата, указанного в полученном ответе OCSP-службы, сертификату, который был указан в соответствующем запросе к OCSP-службе;
- проверкой актуальности сертификата, используемого для проверки ЭП под CRL (сертификата OCSP-службы); в рамках данной проверки выполняются все выше перечисленные проверки;
- при использовании метки времени должны быть проведены аналогичные проверки для сертификата TSP-сервера, а также проверка ЭП ответа TSP-сервера и проверка соответствия хэш-значения ЭП, содержащегося в ответе TSP-сервера, хэш-значению ЭП, содержащемуся в соответствующем запросе TSP-серверу.

3. При вызове функций клиентского компонента исполнений 3-5 ПК «КриптоПро Ключ» требуется указывать имя и тип провайдера из состава СКЗИ «КриптоПро CSP»/«КриптоПро JCP»; в случае использования провайдера «по умолчанию» (NULL/без указания имени провайдера) необходимо предусмотреть невозможность использования

провайдеров, не входящих в состав СКЗИ. Для функций CryptoAPI допускается использование только провайдеров из состава СКЗИ «КриптоПро CSP» (типы 75, 80 и 81). Для методов JCA/JCE, в которых могут быть заданы алгоритмы и используемые криптопровайдеры, необходимо использовать отечественные алгоритмы ГОСТ и провайдеры типов JCP, JCSP, Crypto, RevCheck, JTLS.

4. При вызове функций СКЗИ в прикладном ПО должна быть предусмотрена проверка кода завершения вызываемой функции с обработкой ошибочных ситуаций и регистрацией событий. При возникновении критических исключений необходимо блокировать криптографические вызовы, а при возникновении других исключений корректно их обрабатывать.

5. При необходимости обеспечения аутентификации отправителя зашифрованного сообщения (подписанного и зашифрованного сообщения), а также имитозащиты, должны применяться дополнительные механизмы аутентификации/имитозащиты. Указанная необходимость определяется в Техническом задании.

6. SDK (в составе исполнения 2) и исполнения 3-5 СКЗИ помимо программных интерфейсов, описанных в «Руководствах разработчика» (для используемых SDK и ОС), входящих в состав эксплуатационной документации на СКЗИ, содержат программный интерфейс встроенного СКЗИ «КриптоПро CSP»[4]. В случае его использования должны выполняться соответствующие требования раздела «Требования при встраивании СКЗИ в прикладные системы» Правил пользования на используемое в составе ПК «КриптоПро Ключ» СКЗИ «КриптоПро CSP».

7. Корневые сертификаты должны быть получены по доверенному каналу. Размещение в хранилище корневых сертификатов должно проводиться Администратором безопасности с контролем необходимости наличия конкретных корневых сертификатов, используемых для построения цепочек сертификатов.

8. В эксплуатационной документации на СФ должна быть предусмотрена необходимость проведения проверки ПО BIOS ПЭВМ, на которых предполагается функционирование СКЗИ и СФ, на соответствие методическим документам ФСБ России в области исследований программного обеспечения BIOS, либо проведение указанной проверки для конкретной конфигурации аппаратной платформы.

9. При встраивании СКЗИ в компоненты СФ, в которых функции создания электронной подписи не являются автоматическими, необходимо проводить оценку соответствия компонентов СФ п.п.8 и/или 9 «Требований к средствам электронной подписи», утверждённых Приказом ФСБ России от 27 декабря 2011 г. №796 «Об утверждении Требований к средствам электронной подписи и Требований к средствам удостоверяющего

[4] «КриптоПро Ключ SDK» и «КриптоКлюч SDK» содержат программный интерфейс встроенного СКЗИ «КриптоПро CSP» версии 5.0 R3.

центра». При этом должно быть обеспечено однозначное отображение всех подписываемых данных.

10. При использовании СКЗИ в компонентах СФ для автоматического и/или неавтоматического создания электронной подписи данная СФ должна обеспечивать соответствие данных, подаваемых в функции работы с ЭП, заданному формату, а также контроль содержимого подписываемых данных.

11. В случае опосредованного использования в ПО вызовов функций СКЗИ Техническое задание на проведение работ по ОВ такого ПО на СКЗИ необходимо согласовывать, в том числе, с ООО «КРИПТО-ПРО».

12. При встраивании СКЗИ в ИС необходимо на основании модели угроз и нарушителя на эту систему определить необходимость применения антивирусных средств (АВС). Если такая необходимость определена, должны применяться АВС, сертифицированные органом, ответственным за обеспечение информационной безопасности в создаваемой ИС.

13. При реализации ПО ИС, обращающегося к серверным компонентам, необходимо учитывать уровень доверия, назначаемый в зависимости от сценариев авторизации ИС и идентификации/аутентификации Пользователя (описание уровней доверия приведено в «ЖТЯИ.00118-01 91 01. Руководство разработчика»).

При разработке ИС с использованием функций СКЗИ назначаемый уровень доверия может быть повышен (соблюдая правило, что сценарий авторизации только с идентификацией может быть повышен только до среднего уровня доверия, сценарий авторизации с аутентификацией может быть повышен только до высокого уровня доверия) по результатам проведения исследований по ОВ.

14. Получение Операторами полного доступа к функциям СКЗИ без проведения исследований по ОВ допускается только при использовании следующих сценариев авторизации ИС и аутентификации Оператора (описание сценариев приведено в «ЖТЯИ.00118-01 91 01. Руководство разработчика»):

- авторизация ИС по сценарию 4 (описание сценария приведено в разделе «Авторизация с аутентификацией по сертификату с использованием кода авторизации» документа «ЖТЯИ.00118-01 91 01. Руководство разработчика») с использованием параметра promt со значением none;
- аутентификация Оператора через Веб-интерфейс Пользователя без сторонних ЦИ по сценарию 4.

Получение Оператором полного доступа к функциям СКЗИ при использовании других сценариев допускается только по результатам проведения исследований по ОВ.

15. При получении сертификатов пользователя (например, с помощью методов certificates/* и v2/certificates/* интерфейса REST API) в случае использования сценариев

авторизации с низким уровнем доверия ИС должна убедиться, что полученные сертификаты (данные о сертификатах) соответствуют целевому пользователю.

16. При необходимости включения ИС в число доверенных для снятия в СКЗИ ограничений доступа к функциям от имени пользователя данная ИС должна обеспечивать получение аутентификационных данных пользователя доверенным образом и поддержку необходимых мер защиты процесса своей авторизации. Указанная необходимость и перечень мер защиты определяется в Техническом задании.

17. При разработке ИС с использованием функций СКЗИ в реализации сценариев её авторизации необходимо обеспечивать конфиденциальность соответствующих подключению к СКЗИ кодов авторизации, маркеров доступа и случайных величин в state.

18. При разработке ИС с использованием функций СКЗИ рекомендуется реализовывать клиентскую часть ПО протоколов TLS и OAuth 2.0/OIDC в рамках одного ПО и располагать их на ЭВМ с адресом redirect_uri, указываемым в процессе авторизации ИС. Реализация ИС другим образом допускается только в случае дополнительной реализации и использования ИС параметра state протокола OAuth 2.0.

19. Операторам и Пользователям запрещено размещать конфиденциальные сведения о Пользователях в УЗ Пользователей.

20. Должны быть проведены исследования реализации в информационной системе (ИС) механизмов формирования и отправки документов на серверные компоненты. Перечень работ, которые необходимо провести в рамках исследований, определяется в зависимости от конечной ИС и актуальной для неё модели угроз и нарушителя.

4.2 Требования по размещению

Серверные компоненты

При размещении ПАКМ «КриптоПро HSM», входящего в исполнение 1 ПК «КриптоПро Ключ»), следует руководствоваться требованиями, приведенными в эксплуатационной документации на используемый ПАКМ (в частности, в разделе «Требования по размещению технических средств» Правил пользования на ПАКМ «КриптоПро HSM»).

При размещении технических средств с установленными серверными компонентами исполнения 1 ПК «КриптоПро Ключ», включающие в свой состав СКЗИ «КриптоПро CSP», следует руководствоваться требованиями и рекомендациями, приведенными в разделе «Требования по размещению технических средств с установленным СКЗИ» Правил пользования на используемую на серверном компоненте версию СКЗИ «КриптоПро CSP».

Клиентские компоненты

При размещении технических средств с установленными клиентскими компонентами исполнений 3-5 ПК «КриптоПро Ключ», включающими в состав СКЗИ «КриптоПро CSP» или «КриптоПро JCP», следует руководствоваться требованиями, приведенными в разделе

«Требования по размещению технических средств с установленным СКЗИ» Правил пользования на используемую версию СКЗИ «КриптоПро CSP» или «КриптоПро JCP».

Для клиентских компонентов исполнения 2 ПК «КриптоПро Ключ» — МП «КриптоКлюч», МП «КриптоПро Ключ» и МП, разработанных на основе фреймворков «КриптоКлюч SDK» и «КриптоПро Ключ SDK» — к техническим средствам МУ должны быть приняты меры по исключению НСД посторонних лиц к МУ с установленным СКЗИ:

1. Пользователь МП должен обеспечить недоступность своего МУ посторонним лицам на протяжении всего времени использования СКЗИ с момента появления ключа ЭП в МУ и до того момента, пока сертификат ключа проверки ЭП не перестанет быть действительным.

2. В случае необходимости доступа посторонних лиц к МУ пользователь МП должен безопасным образом удалить с МУ ключи ЭП, хранящиеся с автономной защитой.

Указанные выше требования должны выполняться в том числе и при условии прекращения использования ИС, в рамках которой применяются ключи пользователя.

Размещение технических средств с установленным СКЗИ, а также внос и использование МУ с установленным СКЗИ в помещения, в которых осуществляется обработка информации, содержащей сведения, составляющие государственную тайну, осуществляется установленным порядком.

При обработке с помощью СКЗИ конфиденциальной информации, передаваемой по каналам связи, выходящим за пределы контролируемой территории, необходимо:

1. Для проводных каналов (электрических и оптических) использовать любое из следующих устройств:

- Волоконно-оптические линии связи;
- Оптические развязывающие устройства, устанавливаемые в тракт передачи информации для создания оптоволоконного фрагмента сети;
- Сертифицированные СКЗИ для передачи информации соответствующего уровня конфиденциальности.

2. Для радиоканалов:

Использовать радиоканал GSM, либо GPRS, либо 3G/4G, либо Wi-Fi, либо другой канал мобильной и беспроводной связи, работающий с цифровой модуляцией штатного информационного сигнала.

При размещении технических средств с установленными серверными и клиентскими компонентами СКЗИ должны выполняться следующие требования:

1. Для взаимодействия МП/ИС и серверных компонентов СКЗИ необходимо использовать канал связи, защищенный с использованием протокола TLS с поддержкой российских криптографических алгоритмов (ГОСТ TLS).

2. Защита канала связи между серверными компонентами СКЗИ с использованием протокола ГОСТ TLS не требуется в случае их размещения на одном общем техническом средстве или размещения в одной серверной стойке (при этом канал связи между этими компонентами СКЗИ должен быть выделенным).

3. Серверные компоненты СКЗИ Сервис Подписи (СП) и Сервис Обработки Данных (СОД) должны размещаться либо на одном общем техническом средстве, либо на технических средствах, расположенных в одной серверной стойке.

4. В случае размещения серверных компонентов СКЗИ на разных технических средствах и в разных серверных стойках для взаимодействия этих компонентов необходимо использовать канал связи, защищенный с использованием протокола ГОСТ TLS. При этом для защиты каналов связи между Центром Идентификации (ЦИ) и Сервисом Обработки Данных (СОД) необходимо использовать протокол ГОСТ TLS с двусторонней аутентификацией.

5. Размещение СУБД из состава серверных компонентов СКЗИ и остального ПО серверных компонентов СКЗИ в разных сегментах контролируемой зоны допускается в случае обеспечения дополнительной защиты канала связи одним из следующих способов:

- с использованием протокола ГОСТ TLS, реализуемого установленными на СВТ с СУБД программными средствами СКЗИ класса КС3 (например, средствами утилиты *stunnel* из состава СКЗИ «КриптоПро CSP»).
- с использованием протокола ГОСТ TLS, реализуемого программно-аппаратными средствами СКЗИ класса КС3 и располагаемых в одной серверной стойке с СУБД (при этом канал связи между данным СКЗИ и СУБД должен быть выделенным).

4.3 Требования к персоналу, обслуживающему СКЗИ

В организации, эксплуатирующей СКЗИ, должен быть назначен Администратор безопасности, на которого возлагаются задачи по установке и настройке программно-аппаратных средств, общесистемного и специального ПО, эксплуатируемого совместно с СКЗИ, по установке и первичной настройке СКЗИ, по организации работ по использованию СКЗИ, выработке соответствующих инструкций для пользователей, а также контроля за соблюдением требований по безопасности.

Допускается назначать группу лиц (например, Администратора безопасности СКЗИ и Системного администратора), распределяя между ними приведённые в настоящем документе обязанности Администратора безопасности, а также возлагать при эксплуатации клиентских компонентов исполнения 2 ПК «КриптоПро Ключ» на Пользователя задачи по настройке программной среды МУ и установке СКЗИ в соответствии с эксплуатационной документацией на СКЗИ, с учётом модели угроз и нарушителя ИС, в которой применяется СКЗИ, и с учётом

нормативных документов, определяющих создание такой ИС. В этом случае порядок распределения обязанностей, функций и полномочий между ответственными лицами должен быть зафиксирован в эксплуатационной документации ИС.

Администратор безопасности не должен иметь возможность доступа к конфиденциальной информации пользователей.

Правом доступа к рабочим местам с установленными СКЗИ должны обладать только определённые для эксплуатации лица, прошедшие соответствующую подготовку. Администратор безопасности должен ознакомить каждого пользователя, применяющего СКЗИ, с документацией на СКЗИ, а также с другими нормативными документами, созданными на её основе.

Ролевая модель ПК «КриптоПро Ключ» включает следующие роли:

1. Пользователь;
2. Оператор;
3. Администратор безопасности (Администратор);
4. Системный Администратор.

Ролевая модель лиц, взаимодействующих с СКЗИ, с описанием их задач и выполняемых функций приведена в документе «ЖТЯИ.00118-01 96 01. Общее описание».

Для Пользователей и Операторов дополнительно реализовано разграничение доступа на уровне функций СКЗИ, зависящее от выбранного сценария авторизации ИС и от выбранного способа аутентификации пользователя, подробнее см. «Руководство разработчика» (для используемых SDK и ОС).

4.4 Инициализация и ввод СКЗИ в эксплуатацию

4.4.1 Требования к установке СКЗИ, общесистемного и специального ПО

При установке СКЗИ «КриптоПро CSP» и «КриптоПро JCP», входящих в комплектацию исполнений 1 и 3-5 ПК «КриптоПро Ключ», а также общесистемного и специального ПО на технические средства следует руководствоваться требованиями к установке, приведенными в эксплуатационной документации на СКЗИ, используемое в составе серверного или клиентского компонента ПК (в частности, разделом «Требования по установке СКЗИ, общесистемного и специального ПО на ПЭВМ» Правил пользования на СКЗИ «КриптоПро CSP» и разделами «Меры по обеспечению защиты от НСД», «Меры обеспечения безопасности функционирования рабочих мест со встроенными СКЗИ» Правил пользования на СКЗИ «КриптоПро JCP»).

Администратор безопасности должен сконфигурировать ОС, в среде которой планируется использовать СКЗИ, и осуществлять периодический контроль сделанных

настроек в соответствии с требованиями эксплуатационной документации на используемое СКЗИ «КриптоПро CSP» и «КриптоПро JCP».

При установке клиентских компонентов, входящих в комплектацию исполнений 1 и 3-5 ПК «КриптоПро Ключ», — МП «КриптоКлюч», МП «КриптоПро Ключ» и МП, разработанных на основе фреймворков «КриптоКлюч SDK» и «КриптоПро Ключ SDK» — должны выполняться следующие требования:

1. Перед установкой СКЗИ необходимо проверить целостность полученных установочных модулей СКЗИ в соответствии с инструкциями раздела «Контроль целостности мобильных приложений» документа «ЖТЯИ.00118-01 91 01. Руководство администратора».

2. На МУ, предназначенных для работы с СКЗИ, допустимо использовать только лицензионное ПО.

3. На МУ не должны устанавливаться средства разработки ПО, отладчики и трассировщики. Если средства отладки приложений и/или трассировки вызовов нужны для технологических потребностей организации, то их использование должно быть санкционировано Администратором безопасности. При этом должны быть реализованы меры, исключающие возможность использования этих средств для редактирования кода и памяти СКЗИ и приложений, использующих СКЗИ, а также для просмотра кода и памяти СКЗИ и приложений, использующих СКЗИ, в процессе обработки СКЗИ защищаемой информации и/или при загруженной ключевой информации.

4. При использовании ОС iOS с операционной системой не должно быть произведено операций, расширяющих возможности root-доступа, т.н. Jailbreak.

5. При использовании ОС Android должно быть запрещено использование Accessibility Service, МУ не должно иметь root-прав.

Администратор безопасности должен сконфигурировать ОС iOS или Android, в среде которой планируется использовать МП, и осуществлять периодический контроль сделанных настроек в соответствии с требованиями:

- на МУ должна быть установлена только одна ОС;
- не допускается использовать нестандартные, измененные или отладочные версии ОС;
- необходимо исключить возможность загрузки и использования ОС, отличной от предусмотренной штатной работой;
- в случае поддержки в мобильной ОС многопользовательского режима в системе регистрируется один пользователь, обладающий правами администратора, на которого возлагается обязанность конфигурировать ОС, настраивать безопасность ОС, а также конфигурировать МУ, на которое установлена ОС;

- необходимо исключить возможность удаленного управления, администрирования и модификации ОС и её настроек;

- необходимо регулярно устанавливать пакеты обновления безопасности ОС;

- при использовании СКЗИ на МУ, подключённых к общедоступным сетям связи, с целью исключения возможности НСД к системным ресурсам используемых ОС, к ПО, в окружении которого функционируют СКЗИ, и к компонентам СКЗИ со стороны указанных сетей должны использоваться дополнительные методы и средства защиты (например, организация VPN-сетей или подключение через локальные сети, использующие средства межсетевых экранов, IPS/IDS и т.п.);

- в случае подключения МУ с установленным СКЗИ к общедоступным сетям передачи данных, необходимо исключить возможность открытия и исполнения файлов и скриптовых объектов (например, JavaScript, VBScript, ActiveX), полученных из общедоступных сетей передачи данных, без проведения соответствующих проверок на предмет содержания в них программных закладок и вредоносного ПО, загружаемых из сети;

- необходимо организовать и использовать систему аудита, организовать регулярный анализ результатов аудита;

- необходимо организовать и использовать комплекс мероприятий антивирусной защиты;

- должны быть отключены радиоканалы, неиспользуемые в работе СКЗИ.

4.4.2 Установка СКЗИ

Установка ПАКМ «КриптоПро HSM», входящего в комплектацию исполнения 1 ПК «КриптоПро Ключ», должна выполняться в соответствии с требованиями и инструкциями эксплуатационной документации на ПАКМ (в частности, в соответствии с документом «Инструкция по использованию»).

Установка СКЗИ «КриптоПро CSP»/«КриптоПро JCP», входящих в комплектацию исполнений 1 и 3-5 ПК «КриптоПро Ключ», должна выполняться в соответствии с требованиями и инструкциями эксплуатационной документации на используемую версию СКЗИ (в частности, в соответствии с документом «Руководство администратора безопасности» для используемой ОС для СКЗИ «КриптоПро CSP» и документом «Инструкция по использованию» для СКЗИ «КриптоПро JCP»).

Установка серверных программных компонентов, входящих в комплектацию исполнения 1 ПК «КриптоПро Ключ», и необходимого дополнительного ПО должна выполняться Администратором безопасности в соответствии с требованиями и инструкциями документа «ЖТЯИ.00118-01 91 01. Руководство администратора».

Установка МП «КриптоКлюч», МП «КриптоПро Ключ» и МП, разработанных на основе фреймворков «КриптоКлюч SDK» и «КриптоПро Ключ SDK», входящих в состав исполнения 2

ПК «КриптоПро Ключ», на МУ должна выполняться Администратором безопасности в соответствии с требованиями и инструкциями эксплуатационной документации на используемое МП.

Перед установкой ПО СКЗИ необходимо убедиться в целостности полученных дистрибутивов в соответствии с п. 5.3.

4.4.3 Настройки и параметры СКЗИ

Настройка и инициализация ПАКМ «КриптоПро HSM», входящего в комплектацию исполнения 1 ПК «КриптоПро Ключ», должны выполняться в соответствии с требованиями и инструкциями эксплуатационной документации на ПАКМ (в частности, в соответствии с документом «Инструкция по использованию»).

После установки СКЗИ «КриптоПро CSP»/«КриптоПро JCP», входящих в комплектацию исполнений 1 и 3-5 ПК «КриптоПро Ключ», должны быть выполнены настройки ПО в соответствии с требованиями и инструкциями эксплуатационной документации на используемую версию СКЗИ (в частности, в соответствии с документами «Руководство администратора безопасности» для используемой ОС и «Правила пользования» для используемого СКЗИ).

После установки серверных программных компонентов, входящих в комплектацию исполнения 1 ПК «КриптоПро Ключ», должны быть выполнены настройки ПО в соответствии с требованиями и инструкциями документа «ЖТЯИ.00118-01 91 01. Руководство администратора».

4.4.4 Ввод СКЗИ в эксплуатацию

Ввод СКЗИ (исполнение 1 ПК «КриптоПро Ключ») в эксплуатацию осуществляется Администратором безопасности.

При вводе СКЗИ в эксплуатацию соответствующая информация должна быть внесена в раздел «СВЕДЕНИЯ О ЗАКРЕПЛЕНИИ ИЗДЕЛИЯ ПРИ ЭКСПЛУАТАЦИИ» Формуляра на СКЗИ.

Необходимость и порядок фиксации факта ввода в эксплуатацию исполнений 2-5 СКЗИ определяется эксплуатирующей организацией в зависимости от требований, предъявляемых к ИС.

5 Требования по обеспечению безопасности при эксплуатации СКЗИ

5.1 Общие требования по защите от НСД

Защита информации от НСД должна обеспечиваться на всех технологических этапах обработки информации и во всех режимах функционирования, в том числе при проведении ремонтных и регламентных работ.

Защита информации от НСД должна предусматривать контроль эффективности используемых средств защиты от НСД. Этот контроль должен периодически выполняться Администратором безопасности на основе требований документации на средства защиты от НСД.

5.2 Требования к аутентификации и разграничению доступа

При использовании СКЗИ необходимо разработать и применять политику назначения и смены паролей (в т.ч. пин-кодов).

Для исполнений 1 и 3-5 политика должна удовлетворять требованиям, приведенным в эксплуатационной документации на СКЗИ, входящие в комплектацию соответствующих исполнений (в частности, в разделе «Требования к аутентификации и разграничению доступа» Правил пользования на ПАКМ «КриптоПро HSM», разделе «Требования по обеспечению защиты от НСД» Правил пользования на СКЗИ «КриптоПро CSP» и разделе «Меры по обеспечению защиты от НСД» Правил пользования на СКЗИ «КриптоПро JCP»).

При установке пароля на вход через веб-интерфейс рекомендуется использовать механизм генерации парольных фраз.

Для клиентских компонентов, входящих в состав исполнения 2 ПК «КриптоПро Ключ», политика должна удовлетворять следующим требованиям:

- Для используемой мобильной ОС должна быть включена поддержка парольного входа. Необходимо задать сложность пароля и настройки для удаления данных в случае неправильного ввода пароля, соответствующие политике безопасности.
- Используемые (задаваемые вручную) пароли должны соответствовать следующим требованиям:
    - длина пароля не менее 8 символов;
    - периодичность смены пароля не более 6 месяцев;
    - в числе символов пароля обязательно должны присутствовать буквы в верхнем и нижнем регистрах, цифры и специальные символы (@, #, $, &, *, % и т.п.);

- пароль не должен включать в себя легко вычисляемые сочетания символов (имена, фамилии и т. д.), а также общепринятые сокращения (USER, ALEX и т. д.);
- при смене пароля новое значение должно отличаться от предыдущего не менее чем в 4-х позициях;
- после 10 неверных попыток ввода пароля пользователем при входе в ОС система должна блокироваться на 1 час (при наличии возможности соответствующей настройки в ОС).

- Личный пароль пользователь не имеет права сообщать никому.

- Пароли для аутентификации пользователей на носителях, работающих в режиме активного вычислителя с защитой канала между носителем и СКЗИ по протоколу SESPAKE, должны удовлетворять требованиям, приведенным в разделе «Требования по обеспечению защиты от НСД» Правил пользования на СКЗИ «КриптоПро CSP».

Необходимо ограничить использование паролей по умолчанию и сохранение (в т.ч. с получением доступа по отпечатку пальца или сканированию лица, например, средствами TouchID/FaceID) паролей, используемых в работе СКЗИ:

- в исполнении 2 (МП/SDK в режиме как распределенного, так и автономного хранения ключей ЭП) использование пароля по умолчанию и долговременное хранение (в т.ч. с получением доступа по отпечатку пальца или сканированию лица) паролей, используемых в работе СКЗИ, следует запретить с учётом модели угроз и нарушителя ИС, в которой применяется СКЗИ, а также с учетом нормативных документов, определяющих создание такой ИС; при этом нормативные документы, определяющие создание ИС (включая Техническое задание на разработку и проведение исследований ИС или ее компонентов), могут устанавливать дополнительные требования к построению ключевой системы с проведением установленным порядком ее исследований;

- в исполнении 5 сохранение в системе паролей, используемых в работе СКЗИ, допускается только при условии однопользовательского применения ПЭВМ, в противном случае необходимо запретить сохранение паролей, используемых в работе СКЗИ;

- в случае использования пароля по умолчанию или долговременного хранения пароля на устройстве к этому устройству с установленным СКЗИ предъявляются требования как к ключевому носителю (см. п. 2.4.3).

При эксплуатации СКЗИ запрещено оставлять без контроля вычислительные средства, на которых эксплуатируется СКЗИ (в том числе ключевые носители) после ввода ключевой информации либо иной конфиденциальной информации. В периоды неиспользования ПЭВМ/МУ должно быть заблокировано, разблокировка устройства должна осуществляться при помощи системного механизма аутентификации.

5.3 Требования по обеспечению целостности СКЗИ

При использовании ПАКМ «КриптоПро HSM», входящего в комплектацию исполнения 1 ПК «КриптоПро Ключ», следует руководствоваться требованиями по обеспечению целостности СКЗИ, приведенными в эксплуатационной документации на используемый ПАКМ (в частности, в разделе «Требования по обеспечению целостности» Правил пользования на ПАКМ «КриптоПро HSM»).

Контролем целостности должны быть охвачены файлы серверных компонентов СКЗИ, указанные в «ЖТЯИ.00118-01 91 01. Руководство Администратора».

При использовании исполнений 1 и 3-5 ПК «КриптоПро Ключ», включающие в свой состав СКЗИ «КриптоПро CSP» или «КриптоПро JCP», следует руководствоваться требованиями по обеспечению целостности СКЗИ, приведенными в эксплуатационной документации на используемую версию СКЗИ «КриптоПро CSP» или «КриптоПро JCP» (в частности, в разделах «Требования по установке СКЗИ, общесистемного и специального ПО на ПЭВМ» и «Требования по криптографической защите» Правил пользования на СКЗИ «КриптоПро CSP» и разделе «Требования по проведению контроля целостности» Правил пользования на СКЗИ «КриптоПро JCP»).

При использовании клиентских компонентов, входящих в состав исполнения 2 ПК «КриптоПро Ключ», должны выполняться следующие требования:

1. При разработке МП на базе SDK в данном МП должен быть обеспечен пусковой, периодический и регламентный контроль целостности модулей МП в соответствии с «Руководством разработчика» (для используемых SDK и ОС).

Если в результате контроля целостности обнаруживается нарушение целостности, выполнение МП криптографических функций должно быть заблокировано, а пользователю должно быть выдано соответствующее сообщение с рекомендацией обратиться к Администратору безопасности. Администратор безопасности должен проанализировать причину, приведшую к нарушению целостности, и в случае необходимости переустановить МП.

2. Механизм контроля целостности МП должен быть реализован в соответствии с описанием, приведённым в «Руководстве разработчика» (для используемых SDK и ОС).

3. Контроль целостности МП перед его установкой должен проводиться в соответствии с «ЖТЯИ.00118-01 91 01. Руководство Администратора».

Установка клиентских компонентов на рабочем месте пользователя может быть осуществлена только в случае подтверждения целостности полученных установочных модулей приложения, модулей СКЗИ и эксплуатационной документации в соответствии с эксплуатационной документацией на СКЗИ, входящее в состав клиентского компонента.

После завершения процесса установки должны быть выполнены действия, необходимые для осуществления периодического контроля целостности установленного ПО

СКЗИ, а также его окружения в соответствии с эксплуатационной документацией на СКЗИ, входящее в состав клиентского компонента.

При развёртывании программных компонентов СКЗИ в замкнутой программной среде (ЗПС) ОС СН Astra Linux SE должны выполняться следующие требования:

1. В процессе ввода в эксплуатацию Администратору безопасности рекомендуется установить ОС в минимально необходимой для функционирования СКЗИ конфигурации, после чего Администратор должен включить и настроить в соответствии с эксплуатационной документацией на ОС СН Astra Linux SE все необходимые встроенные средства защиты, установить программные компоненты СКЗИ и программные средства, относящиеся непосредственно к эксплуатации СКЗИ, и включить режим ЗПС ОС.

2. Установка программных компонентов СКЗИ и программных средств, относящихся непосредственно к эксплуатации СКЗИ, должна выполняться в соответствии с положениями, приведёнными в разделе «Развертывание КриптоПро Ключ» документа «ЖТЯИ.00118-01 91 01. Руководство Администратора».

3. Для ПО, устанавливаемого в ОС СН Astra Linux SE, должны выполняться требования, изложенные в эксплуатационной документации на ОС СН Astra Linux SE.

4. Запрещается устанавливать ПО, не относящееся к функционированию инфраструктуры организации, эксплуатирующей СКЗИ, а в отношении устанавливаемого ПО должен выполняться комплекс проверочных мероприятий, предусмотренный политикой безопасности данной организации и включающий в том числе проверку с использованием АВС, проверку отсутствия уязвимостей, опубликованных в общедоступных источниках, и проверку установки лицензионного ПО.

5.4 Порядок обеспечения работоспособности СКЗИ

5.4.1 Ограничение срока непрерывного функционирования СКЗИ

При использовании ПАКМ «КриптоПро HSM», входящего в комплектацию исполнения 1 ПК «КриптоПро Ключ», периодичность перезагрузки ПАКМ определяется в соответствии с п. «Ежесуточное нагрузочное тестирование ПАКМ» Правил пользования на ПАКМ «КриптоПро HSM».

При использовании исполнений 1 и 3-5 ПК «КриптоПро Ключ», включающих в свой состав СКЗИ «КриптоПро CSP» или «КриптоПро JCP», периодичность перезагрузки технических средств с установленным СКЗИ определяется в соответствии с разделом «Требования по криптографической защите» Правил пользования на СКЗИ «КриптоПро CSP» и разделом «Требования по защите от НСД» Правил пользования на СКЗИ «КриптоПро JCP».

При использовании исполнений 2 необходимо не реже чем 1 раз в 7 дней осуществлять перезагрузку технических средств с установленными компонентами ПК «КриптоПро Ключ».

5.4.2 Журналирование и аудит

Необходимо настроить систему аудита ПАКМ «КриптоПро HSM», входящего в комплектацию исполнения 1 ПК «КриптоПро Ключ», в соответствии с эксплуатационной документацией на ПАКМ.

На технических средствах с установленными серверными (исполнение 1 ПК «КриптоПро Ключ») и клиентскими (исполнения 3-5 ПК «КриптоПро Ключ») компонентами СКЗИ необходимо организовать и использовать систему аудита, а также регулярный анализ результатов аудита. В настройках ОС, отвечающих за ведение журналов событий, необходимо установить режим архивирования журнала при его заполнении.

Настройка журналирования серверных компонентов СКЗИ должна выполняться в соответствии с инструкциями документа «ЖТЯИ.00118-01 91 01. Руководство администратора».

Журнал аудита необходимо хранить в полном объеме за период, покрывающий срок действия технологических ключей (в т.ч. ключевой пары дополнительной защиты ключей Пользователей при использовании исполнения 2 в режиме распределенного хранения ключей ЭП), хранящихся на серверном компоненте.

5.4.3 Восстановление работоспособности СКЗИ

При восстановлении работоспособности ПАКМ «КриптоПро HSM», входящего в комплектацию исполнения 1 ПК «КриптоПро Ключ», следует руководствоваться требованиями и инструкциями, приведенными в эксплуатационной документации на используемый ПАКМ (в частности, разделом «Требования по выполнению технического обслуживания и ремонта ПАКМ» Правил пользования на ПАКМ «КриптоПро HSM»).

В случае нарушения работоспособности СКЗИ вследствие ошибки проверки целостности компонентов СКЗИ Администратор безопасности должен выявить причину и обстоятельства нарушения целостности СКЗИ и переустановить СКЗИ в соответствии с инструкцией по установке, описанной в эксплуатационной документации (см. 4.4.2).

Для исключения сбоев технических средств с установленными компонентами СКЗИ, вызванных отключением электропитания, необходимо обеспечить электропитание сервера от источника бесперебойного питания достаточной мощности. Как минимум, мощности батарей источника бесперебойного питания должно хватать на время достаточное для корректного автоматического завершения работы сервера.

6 Требования по обеспечению безопасности при выводе СКЗИ из эксплуатации и передаче в ремонт

6.1 Ремонт СКЗИ

В случае необходимости проведения ремонтных и регламентных работ ПАКМ «КриптоПро HSM», входящего в комплектацию исполнения 1 ПК «КриптоПро Ключ», следует руководствоваться требованиями по обеспечению безопасности при передаче СКЗИ в ремонт, приведенными в эксплуатационной документации на ПАКМ (в частности, разделом «Требования по выполнению технического обслуживания и ремонта ПАКМ» Правил пользования на ПАКМ «КриптоПро HSM»).

В случае необходимости проведения ремонтных и регламентных работ аппаратной части СФ исполнений 2-5 ПК «КриптоПро Ключ» необходимо обеспечить невозможность доступа нарушителя к ключевой информации, содержащейся в СФ. Конкретный перечень мер должен быть определён исходя из условий эксплуатации СКЗИ.

6.2 Вывод СКЗИ из эксплуатации

При выводе СКЗИ из эксплуатации соответствующая информация должна быть внесена в раздел «СВЕДЕНИЯ О ЗАКРЕПЛЕНИИ ИЗДЕЛИЯ ПРИ ЭКСПЛУАТАЦИИ» Формуляра на СКЗИ.

Приложение 1. Перечень методов интерфейса REST API серверных компонентов, использование которых при разработке систем на основе ПК «КриптоПро Ключ» возможно без дополнительных тематических исследований

Конечная точка health

Для всех сервисов ниже доступна конечная точка **health**, позволяющая получить информацию о доступности сервиса.

Конечная точка	Метод	Описание
https://<hostname>/<AppName>/health	get	Получение информации о состоянии сервиса

Сервис Подписи

В таблице 1 приведены относительные URL-адреса методов REST API. Абсолютный URL-адрес методов REST API имеет вид:

https://<hostname>/<AppName>/api, где

- hostname — DNS-имя сервера, на котором развёрнут экземпляр Сервиса Подписи
- AppName – имя веб-приложения Сервиса Подписи (по умолчанию SignServer)

Таблица 1 — Перечень методов интерфейса REST API Сервиса Подписи

Метод	Описание	Ограничения на использование метода
Конечная точка Policy		
policy	Метод получает политику Сервиса Подписи.	
policy/ex	Метод получает расширенную политику Сервиса Подписи.	
Конечная точка Requests		
requests v2/requests	Метод получает (get) список запросов на сертификат или создает (post) неподписанный запрос на сертификат Пользователя (без ключа проверки ЭП).	
requests/client	Метод создает или обновляет запрос на сертификат Пользователя, в том числе добавляет сгенерированный ключ проверки ЭП.	Разрешено использовать только при соответствии алгоритма хэш-функции и алгоритма электронной подписи/ключа проверки электронной подписи, указанных в передаваемом запросе на сертификат Пользователя (параметр CertRequestBase64), алгоритмам ГОСТ Р 34.11-2012 и ГОСТ Р 34.10-2012 соответственно.
requests/{req_id} v2/requests/{req_id}	Метод получает (get) или удаляет (post/delete) запрос на сертификат Пользователя, соответствующий	

	указанному идентификатору.	
requests/{req_id}/status v2/requests/{req_id}/status	Метод изменяет статус (принимает или отклоняет) указанного запроса на сертификат Пользователя.	
requests/revokerequests/{req_id}/status v2/requests/revokerequests/{req_id}/status	Метод изменяет статус (принимает или отклоняет) указанного запроса на отзыв сертификата Пользователя.	
requests/{req_id}/content v2/requests/{req_id}/content	Метод получает печатную форму содержимого указанного запроса на сертификат Пользователя.	
requests/revokerequests/{req_id}/content v2/requests/revokerequests/{req_id}/content	Метод получает печатную форму содержимого указанного запроса на отзыв сертификата Пользователя.	
Конечная точка Certificates		
certificates v2/certificates	Метод получает (get) список доступных сертификатов или устанавливает (post) сертификат Пользователя.	Разрешено использование для установки сертификата только при соответствии алгоритма хэш-функции и алгоритма электронной подписи/ключа проверки электронной подписи, указанных в передаваемом сертификате Пользователя (параметр Certificate), алгоритмам ГОСТ Р 34.11-2012 и ГОСТ Р 34.10-2012 соответственно.
certificates/{cert_id} v2/certificates/{cert_id}	Метод получает (get) или удаляет (post/delete) сертификат Пользователя, соответствующий указанному идентификатору.	
v2/certificates/{cert_id}/status	Метод изменяет статус указанного сертификата Пользователя.	
certificates/{cert_id}/default v2/certificates/{cert_id}/default	Метод назначает указанный сертификат Пользователя в качестве сертификата по умолчанию.	
certificates/{cert_id}/friendlyname v2/certificates/{cert_id}/friendlyname	Метод устанавливает дружественное имя для указанного сертификата Пользователя.	
certificates/{cert_id}/revokerequests v2/certificates/{cert_id}/revokerequests	Метод получает список запросов на отзыв указанного сертификата Пользователя.	
certificates/{cert_id}/content v2/certificates/{cert_id}/content	Метод получает печатную форму содержимого указанного сертификата Пользователя.	
Конечная точка Signature		
v2/signature	Метод создает операцию ЭП документа.	В случае распределенного формирования ЭП параметр DocumentsDskSignVersion должен иметь значение «1». Разрешено использовать только без указания параметра «Hash» в списке SignatureParams. Разрешено использовать при указании в качестве параметра «HashAlgorithm» из списка SignatureParams

		только следующих значений OID-идентификаторов: • szOID_CP_GOST_R3411_12_256_R3410 ("1.2.643.7.1.1.3.2") • szOID_CP_GOST_R3411_12_512_R3410 ("1.2.643.7.1.1.3.3").
v2/enhance	Метод создает операцию усовершенствования ЭП.	Разрешено использовать только без указания параметра «Hash» в списке SignatureParams. Разрешено использовать при указании в качестве параметра «HashAlgorithm» из списка SignatureParams только следующих значений OID-идентификаторов: • szOID_CP_GOST_R3411_12_256_R3410 ("1.2.643.7.1.1.3.2") • szOID_CP_GOST_R3411_12_512_R3410 ("1.2.643.7.1.1.3.3")
Конечные точки Decryption и Encryption		
documents/decrypt/parse v1/decryption/parse v2/decryption/parse v2/parse	Метод выполняет поиск подходящих сертификатов для расшифрования.	
v2/encryption	Метод создает операцию зашифрования документа.	Разрешено использовать только при соответствии алгоритма ключа обмена каждого из сертификатов передаваемого списка сертификатов алгоритмам ГОСТ Р 34.11-2012 и ГОСТ Р 34.10-2012.
v2/decryption	Метод создает операцию расшифрования документа.	
Конечная точка Hash		
v2/hash	Метод получает значение хэш-функции для документа.	Разрешено использовать при указании в качестве параметра «HashAlgorithm» из списка SignatureParams только следующих значений OID-идентификаторов: • szOID_CP_GOST_R3411_12_256_R3410 ("1.2.643.7.1.1.3.2") • szOID_CP_GOST_R3411_12_512_R3410 ("1.2.643.7.1.1.3.3")
Конечная точка Version		
version v2/version	Метод получает версию ПО.	

Сервис Ключ Lite

В таблице 2 приведены относительные URL-адреса методов REST API. Абсолютный URL-адрес методов REST API имеет вид:

https://<hostname>/<AppName>/api, где

- hostname — DNS-имя сервера, на котором развёрнут экземпляр Сервиса Ключ Lite
- AppName – имя веб-приложения Сервиса Ключ Lite

Таблица 2 — Перечень методов интерфейса REST API Сервиса Ключ Lite

Функция	**Описание**	**Ограничения на использование функции**
signatures/presign	Метод предварительно обрабатывает документ и создает операцию ЭП.	Разрешено использовать только при соответствии алгоритма хэш-функции и алгоритма электронной подписи/ключа проверки электронной подписи, указанных в передаваемом сертификате (параметр RawCertificate), алгоритмам ГОСТ Р 34.11-2012 и ГОСТ Р 34.10-2012. Разрешено использовать только без указания параметра «Hash» в списке SignatureParams. Разрешено использовать при указании в качестве параметра «HashAlgorithm» из списка SignatureParams только следующих значений OID-идентификаторов: • szOID_CP_GOST_R3411_12_256_R3410 ("1.2.643.7.1.1.3.2") • szOID_CP_GOST_R3411_12_512_R3410 ("1.2.643.7.1.1.3.3")
signatures/postsign	Метод завершает операцию ЭП.	Разрешено использовать только без указания параметра «Hash» в списке SignatureParams. Разрешено использовать при указании в качестве параметра «HashAlgorithm» из списка SignatureParams только следующих значений OID-идентификаторов: • szOID_CP_GOST_R3411_12_256_R3410 ("1.2.643.7.1.1.3.2") • szOID_CP_GOST_R3411_12_512_R3410 ("1.2.643.7.1.1.3.3")
signatures/postsign (patch) signatures/enhance	Метод выполняет операцию усовершенствования ЭП.	Разрешено использовать только без указания параметра «Hash» в списке SignatureParams. Разрешено использовать при указании в качестве параметра «HashAlgorithm» из списка SignatureParams только следующих значений OID-идентификаторов: • szOID_CP_GOST_R3411_12_256_R3410 ("1.2.643.7.1.1.3.2") • szOID_CP_GOST_R3411_12_512_R3410 ("1.2.643.7.1.1.3.3")
policy	Метод получает политики Сервиса Ключ Lite.	
hash	Метод получает значение хэш-функции для документа.	

Сервис Аудита

В таблице 3 приведены относительные URL-адреса методов REST API. Абсолютный URL-адрес методов REST API имеет вид:

https://<hostname>/<AppName>/api, где

- hostname — DNS-имя сервера, на котором развёрнут экземпляр Сервиса Аудита
- AppName – имя веб-приложения Сервиса Аудита

Таблица 3 — Перечень методов интерфейса REST API Сервиса Аудита

Функция	Описание	Ограничения на использование функции
Конечная точка Audit		
audit	Метод получает записи событий аудита в заданном диапазоне.	
Конечная точка Reports		
reports/policy	Метод получает информацию о доступных плагинах формирования отчетов.	

Сервис Обработки Документов

В таблице 4 приведены относительные URL-адреса методов REST API. Абсолютный URL-адрес методов REST API имеет вид:

https://<hostname>/<AppName>/api, где

- hostname — DNS-имя сервера, на котором развёрнут экземпляр Сервиса Обработки Документов
- AppName – имя веб-приложения Сервиса Обработки Документов

Таблица 4 — Перечень методов интерфейса REST API Сервиса Обработки Документов

Функция	Описание	Ограничения на использование функции
Конечная точка Policy		
policy	Метод получает политики Сервиса Обработки Документов.	
Конечная точка Documents		
documents/pack documents/multipartpack	Метод загружает на сервер пакет документов.	
documents/{id}/info	Метод получает информацию о загруженном документе.	
documents/info	Метод получает информацию о загруженных документах.	
documents/hash/{hash}/info	Метод получает информацию о загруженном документе по хэш-значению.	
documents/{id}/content	Метод получает содержимое загруженного документа.	
documents/content/download	Метод получает содержимое нескольких загруженных документов в формате multipart/form-data.	

documents/{id}	Метод обновляет информацию (patch) или удаляет (delete) документ.	
documents/{id}/converted_content	Метод получает сконвертированный документ в формате PDF для отображения пользователю в веб-интерфейсе.	
documents/{id}/converted_content/{startPage}	Метод получает сконвертированный документ в формате PDF с указанием страницы для отображения пользователю в веб-интерфейсе.	
documents/{id}/preview	Метод получает сконвертированный документ в формате HTML для отображения пользователю в веб-интерфейсе	
documents/{id}/converted_content/info	Метод получает информацию о сконвертированном документе.	
documents/{id}/converted_content	Метод получает сконвертированный документ для отображения пользователю в веб-интерфейсе.	
documents/{id}/supported_convertations	Метод получает информацию о доступных для документа способах конвертации.	

Центр Идентификации

В таблице 5 приведены относительные URL-адреса методов REST API. Абсолютный URL-адрес методов REST API имеет вид:

https://<hostname>/<AppName>/api, где

- hostname — DNS-имя сервера, на котором развёрнут экземпляр Центра Идентификации
- AppName – имя веб-приложения Центра Идентификации (по умолчанию STS)

Таблица 5 — Перечень методов интерфейса REST API Центра Идентификации

Функция	**Описание**	**Ограничения на использование функции**
<u>Сервис Управления Пользователями (UMS)</u>		
Конечная точка User		
Операции с Пользователями		
ums/user	Метод выполняет регистрацию Пользователя с добавлением в группу (post) или получает информацию о Пользователе (get).	
ums/user/register	Метод выполняет регистрацию Пользователя с добавлением в группу по умолчанию.	
ums/user/isunique	Метод проверяет уникальность данных Пользователя.	
ums/user/{id}	Метод получает информацию о Пользователе (get), назначает отображаемое имя и изменяет состояние блокировки Пользователя (post/patch) или удаляет Пользователя (delete).	

Настройка различительного имени Пользователя		
ums/user/{id}/dn	Метод получает список компонент (get) или устанавливает (post) различительное имя Пользователя.	
ums/user/dn/isunique	Метод проверяет уникальность компонентов различительного имени Пользователя.	
Назначение состояния подтвержденности УЗ		
ums/user/{id}/approve	Метод устанавливает информацию о подтвержденности УЗ Пользователя.	
Настройка групп Пользователей		
ums/user/{id}/group	Метод получает название (get) или назначает (post) группу, в которой состоит Пользователь.	
Настройка логина Пользователя		
ums/user/{id}/login/local	Метод устанавливает/изменяет (post) или удаляет (delete) локальный логин Пользователя, предназначенный для аутентификации средствами ЦИ СКЗИ.	
ums/user/{id}/login	Метод получает (get), устанавливает (post) или удаляет (delete) внешний (локальный) логин Пользователя, используемый для аутентификации средствами ЦИ доверенной системы.	
Настройка пароля Пользователя		
ums/user/{id}/password	Метод сбрасывает пароль Пользователя.	
ums/user/{id}/password/print	Метод сбрасывает пароль Пользователя и получает новый пароль в печатной форме.	
Настройка номера телефона Пользователя		
ums/user/{id}/phones	Метод получает (get) или добавляет (post) номер телефона Пользователя.	
ums/user/phones/{phone}/isunique	Метод проверяет уникальность номера телефона Пользователя.	
ums/user/{id}/phones/{phone}/confirm	Метод подтверждает указанный номер телефона Пользователя.	
ums/user/{id}/phones/{phone}/requireconfirm	Метод запрашивает отправку одноразового пароля для подтверждения номера телефона Пользователя.	
ums/user/{id}/phones/{phone}/submitconfirm	Метод подтверждает номер телефона Пользователя при помощи кода подтверждения (одноразового пароля).	
ums/user/{id}/phones/{phone}/primary	Метод устанавливает приоритет номера телефона Пользователя для идентификации (входа).	
ums/user/{id}/phones/{phone}/notification	Метод устанавливает номер телефона Пользователя для получения уведомлений.	
ums/user/{id}/phones/{phone}/secondaryauth	Метод назначает номер телефона Пользователя для вторичной аутентификации и/или подтверждения операций.	
ums/user/{id}/phones/{phone}	Метод удаляет номер телефона Пользователя.	

ums/user/{id}/phonenumber	Метод получает (get), устанавливает (post) или удаляет (delete) номер телефона Пользователя, используемый для входа, вторичной аутентификации и/или подтверждения операций.	
Настройка адреса электронной почты Пользователя		
ums/user/{id}/emails	Метод получает (get) список адресов или добавляет (post) адрес электронной почты Пользователя.	
ums/user/emails/{email}/isunique	Метод проверяет уникальность адреса электронной почты Пользователя.	
ums/user/{id}/emails/{email}/confirm	Метод подтверждает адрес электронной почты Пользователя.	
ums/user/{id}/emails/{email}/requireconfirm	Метод запрашивает отправку одноразового пароля для подтверждения адреса электронной почты Пользователя.	
ums/user/{id}/emails/{email}/submitconfirm	Метод подтверждает адрес электронной почты Пользователя при помощи кода подтверждения (одноразового пароля).	
ums/user/{id}/emails/{email}/primary	Метод устанавливает приоритет адреса электронной почты Пользователя для идентификации (входа).	
ums/user/{id}/emails/{email}/notification	Метод устанавливает адрес электронной почты Пользователя для получения уведомлений.	
ums/user/{id}/emails/{email}/secondaryauth	Метод устанавливает адрес электронной почты Пользователя для вторичной аутентификации.	
ums/user/{id}/emails/{email}	Метод удаляет адрес электронной почты Пользователя.	
ums/user/{id}/email	Метод получает (get), устанавливает (post) или удаляет (delete) адрес электронной почты Пользователя, используемый для входа, вторичной аутентификации и/или подтверждения операций.	
Настройка push[5]-адресов Пользователя		
ums/user/{id}/pushaddress/{activeOnly}	Метод получает информацию о push-адресах Пользователя.	
ums/user/{id}/pushaddress	Метод получает (get) или удаляет (delete) информацию о push-адресе.	
ums/user/{id}/pushaddress/restore	Метод восстанавливает push-адрес для оповещения Пользователя.	
ums/user/{id}/pushaddress/notification/{notification}	Метод включает или отключает оповещение Пользователя посредством push-уведомлений.	
Настройка протокола OATH		
ums/user/{id}/oath	Метод получает информацию (get), устанавливает (post) или удаляет (delete) информацию о токене	

[5] В качестве сервиса отправки PUSH-уведомлений рекомендуется использовать российские сервисы PUSH-уведомлений.

	аутентификации (OTP) Пользователя.	
ums/user/{id}/oath/sync	Метод выполняет синхронизацию данных токена аутентификации (OTP) Пользователя.	
ums/user/{id}/oath/app	Метод выполняет создание данных для приложения-аутентификатора.	
Общие настройки аутентификации		
ums/user/{id}/authmethod	Метод получает информацию о методах аутентификации Пользователя.	
ums/user/{id}/authmethod/idonly	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации Identification Only (только идентификация).	
ums/user/{id}/authmethod/password	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с помощью пароля.	
ums/user/{id}/authmethod/cert	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с использованием сертификата.	
ums/user/{id}/authmethod/external	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации через внешний ЦИ.	
ums/user/{id}/authmethod/otpviasms	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с помощью одноразовых паролей, направляемых по SMS.	
ums/user/{id}/authmethod/otpviaemail	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с помощью одноразовых паролей, направляемых на электронную почту.	
ums/user/{id}/authmethod/oath	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с использованием OTP-токенов.	
ums/user/{id}/authmethod/mydss	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации через МП.	
ums/user/rawauthdata	Метод получает подробную информацию о доступных методах аутентификации Пользователя.	
Настройка аутентификации с помощью МП		
ums/user/{id}/mydss/assign	Метод назначает Пользователю существующее МУ.	
ums/user/{id}/mydss/init	Метод создает аутентификационные данные (QR-код с K_init) для первой аутентификации через МП.	
ums/user/{id}/mydss/init/get	Метод получает инициализационные данные токена аутентификации (QR-кода с K_init).	
ums/user/{id}/mydss/init/sendqr	Метод отправляет аутентификационные данные (QR-код с K_init) на электронную почту.	
ums/user/{id}/mydss/init/resendotp	Метод повторно отправляет код активации для первой	В случае необходимости указать использование распределенного ключа для

	аутентификации через МП.	аутентификации параметр «AuthKeyType» должен иметь значение Cryptokey2. Разрешено использовать только без указания значения CryptoKey параметра «AuthKeyType».
ums/user/{id}/mydss/init/delete	Метод удаляет аутентификационные данные (QR-код с K_init).	
ums/user/{id}/mydss/verify/get	Метод получает аутентификационные данные (QR-код с Nonce).	
ums/user/{id}/mydss	Метод получает информацию (get) или удаляет (delete) информацию о ключах аутентификации для МП.	
ums/user/{id}/mydss/lockout	Метод блокирует или разблокирует МУ Пользователя.	
ums/user/{id}/mydss/delete	Метод удаляет указанный ключ аутентификации для МП.	
Настройка политики подтверждения операций		
ums/user/{id}/operationpolicy	Метод получает (get) или устанавливает (post) политику подтверждения операций Пользователя.	
Настройка политики доступа к операциям		
ums/user/{id}/accesspolicy	Метод получает (get) или устанавливает (post) политику доступа к операциям для Пользователя.	
Настройка политики оповещения		
ums/user/{id}/notificationpolicy/events	Метод получает список событий, оповещения о которых может получить Пользователь.	
ums/user/{id}/notificationpolicy ums/user/{id}/notifications/policy	Метод получает (get) или устанавливает (post) политику оповещений Пользователя.	
Блокировка/разблокировка Пользователя		
ums/user/{id}/lockout ums/user/{id}/lockoutex	Метод блокирует или разблокирует УЗ Пользователя.	
ums/user/{id}/grants/revoke	Метод отзывает права приложений на доступ к ресурсам (OAuth).	
Управление лицензией модуля доступа КриптоПро Cloud CSP		
ums/user/{id}/cloudcsp/serverlicense	Метод получает информацию о состоянии лицензии (get), активирует (post/patch) или деактивирует (delete) лицензию модуля доступа КриптоПро Cloud CSP.	
Конечная точка Users		
ums/users	Метод получает список Пользователей по заданным фильтрам.	
ums/usersbyrdns	Метод получает список Пользователей по компонентам различительных имен.	
Конечная точка Authtokens		
ums/authntokens	Метод получает список средств аутентификации по заданным фильтрам.	

Конечная точка Policy		
ums/policy	Метод получает политику Сервиса Управления Пользователями.	
Конечная точка GroupPolicy		
ums/groupPolicy/{groupName}	Метод получает политики группы пользователей на ЦИ.	
Конечная точка SelfUser		
Операции с Пользователями		
self/register/policy	Метод получает политики самостоятельной регистрации Пользователя.	
self/register	Метод выполняет запрос на самостоятельную регистрацию Пользователя.	
self/info	Метод получает информацию о Пользователе.	
self/displayName	Метод устанавливает или изменяет отображаемое имя Пользователя.	
Настройка различительного имени Пользователя		
self/dn	Метод получает список компонент (get) или устанавливает (post) различительное имя Пользователя.	
Настройка групп Пользователей		
self/group	Метод получает имя группы, в которой состоит Пользователь.	
Настройка логина Пользователя		
self/login/local	Метод устанавливает/изменяет (post) или удаляет (delete) локальный логин УЗ Пользователя, предназначенный для аутентификации средствами ЦИ СКЗИ.	
self/login	Метод получает (get), устанавливает/изменяет (post) или удаляет (delete) внешний логин пользователя, предназначенный для аутентификации средствами Стороннего ЦИ доверенной системы.	
Настройка пароля Пользователя		
self/password/generate	Метод запрашивает генерацию пароля Пользователя средствами сервера.	
self/password/change	Метод изменяет пароль Пользователя.	
Настройка номера телефона Пользователя		
self/phones	Метод получает (get) список номеров или добавляет (post) номер телефона Пользователя в УЗ.	
self/phones/{phone}/confirm	Метод подтверждает указанный номер телефона Пользователя.	
self/phones/{phone}/requirecconfirm	Метод запрашивает отправку одноразового пароля для подтверждения указанного номера телефона Пользователя.	
self/phones/{phone}/submitcconfirm	Метод подтверждает указанный номер телефона Пользователя при помощи одноразового пароля.	
self/phones/{phone}/primary	Метод устанавливает приоритет	

	указанного номера телефона Пользователя в качестве логина для идентификации.	
self/phones/{phone}/notification	Метод устанавливает указанный номер телефона Пользователя в качестве номера для получения уведомлений.	
self/phones/{phone}/secondaryauth	Метод устанавливает указанный номер телефона Пользователя в качестве номера для вторичной аутентификации и/или подтверждения операций.	
self/phones/{phone}	Метод удаляет указанный номер телефона Пользователя из УЗ.	
self/phonenumber	Метод получает (get), устанавливает (post) или удаляет (delete) номер телефона Пользователя, используемый для входа, вторичной аутентификации и/или подтверждения операций.	
Настройка адреса электронной почты Пользователя		
self/emails	Метод получает (get) список адресов или добавляет (post) адрес электронной почты Пользователя.	
self/emails/{email}/confirm	Метод подтверждает указанную электронную почту Пользователя.	
self/emails/{email}/requireconfirm	Метод создает операцию для подтверждения указанной электронной почты Пользователя с использованием одноразового кода.	
self/emails/{email}/submitconfirm	Метод отправляет код для подтверждения указанной электронной почты Пользователя.	
self/emails/{email}/primary	Метод устанавливает приоритет указанной электронной почты Пользователя.	
self/emails/{email}/notification	Метод настраивает возможность отправки оповещений на указанную электронную почту Пользователя.	
self/emails/{email}/secondaryauth	Метод устанавливает указанную электронную почту Пользователя как контакт для отправки одноразовых кодов для вторичной аутентификации.	
self/emails/{email}	Метод удаляет указанную электронную почту Пользователя из УЗ.	
self/email	Метод получает (get), устанавливает (post) или удаляет (delete) электронную почту Пользователя, используемую для входа, вторичной аутентификации и/или подтверждения операций.	
Настройка push-адресов Пользователя		
self/pushaddress/{activeOnly}	Метод получает информацию о push-адресах Пользователя.	
self/pushaddress	Метод получает (get) или удаляет (delete) информацию об указанном push-адресе.	
self/pushaddress/notification/{notification}	Метод включает или отключает оповещение Пользователя	

	посредством push-уведомлений.	
self/pushaddress/restore	Метод восстанавливает push-адрес для оповещения Пользователя.	
Настройка протокола OATH		
self/oath	Метод получает информацию (get), устанавливает (post) или удаляет (delete) информацию о токене аутентификации (OTP) Пользователя.	
self/oath/app	Метод выполняет создание данных для приложения-аутентификатора.	
self/oath/sync	Метод выполняет синхронизацию данных токена аутентификации (OTP) Пользователя.	
Общие настройки аутентификации		
self/authmethod	Метод получает информацию о методах аутентификации Пользователя.	
self/authmethod/idonly	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации Identification Only (только идентификация).	
self/authmethod/password	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с помощью пароля.	
self/authmethod/cert	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с использованием сертификата.	
self/authmethod/external	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации через внешний ЦИ.	
self/authmethod/otpviasms	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с помощью одноразовых паролей, направляемых по SMS.	
self/authmethod/otpviaemail	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с помощью одноразовых паролей, направляемых на электронную почту.	
self/authmethod/oath	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации с использованием OTP-токенов.	
self/authmethod/mydss	Метод назначает (post) или отключает (delete) Пользователю метод аутентификации через МП.	
self/rawauthdata	Метод получает подробную информацию о доступных методах аутентификации Пользователя.	
Настройка аутентификации с помощью МП		
self/mydss/init	Метод создает аутентификационные данные (QR-код с K_init) для первой аутентификации через МП.	В случае необходимости указать использование распределенного ключа для аутентификации параметр «AuthKeyType» должен иметь значение Cryptokey2. Разрешено использовать

		только без указания значения CryptoKey параметра «AuthKeyType».
self/mydss/init/get	Метод получает инициализационные данные токена аутентификации (QR-кода с K_init).	
self/mydss/init/resendotp	Метод повторно отправляет код активации для первой аутентификации через МП.	В случае необходимости указать использование распределенного ключа для аутентификации параметр «AuthKeyType» должен иметь значение Cryptokey2. Разрешено использовать только без указания значения CryptoKey параметра «AuthKeyType».
self/mydss/init/delete	Метод удаляет аутентификационные данные (QR-код с K_init).	
self/mydss/verify/get	Метод получает аутентификационные данные (QR-код с Nonce).	
self/mydss	Метод получает информацию (get) или удаляет (delete) информацию о ключах аутентификации для МП.	
self/mydss/lockout	Метод блокирует или разблокирует МУ Пользователя.	
self/mydss/delete	Метод удаляет указанный ключ аутентификации для МП.	
Настройка политики подтверждения операций		
self/operationpolicy	Метод получает (get) или устанавливает (post) политику подтверждения операций Пользователя.	
Настройка политики доступа к операциям		
self/accesspolicy	Метод получает (get) или устанавливает (post) политику доступа к операциям для Пользователя.	
Настройка политики оповещения		
self/notificationpolicy/events	Метод получает список событий, оповещения о которых может получить Пользователь.	
self/notificationpolicy	Метод получает (get) или устанавливает (post) политику оповещений Пользователя.	
Настройка протокола OAuth		
self/grants	Метод получает список приложений, запросивших права на доступ к ресурсам (OAuth).	
self/grants/revoke	Метод отзывает права приложений на доступ к ресурсам (OAuth)	

Сервис Операций		
Конечная точка Operations		
Получение активных операций пользователя		
operations	Метод получает информацию о текущих (активных) операциях Пользователя, ожидающих	

	подтверждения через мобильное устройство.	
operations/all	Метод получает информацию о текущих (активных) операциях Пользователя.	
Получение сведений об операции пользователя		
operations/{op_id}	Метод получает информацию об указанной операции Пользователя.	
Конечная точка Confirmation		
v2.0/confirmation v2.0/confirmation/cert	Метод инициирует подтверждение операции Пользователя.	
Другое		
Получение сведений об операциях и уведомлениях		
transactions	Метод получает список операций, ожидающих подтверждения.	
transactions/all	Метод получает список всех операций.	
transactions/info	Метод получает информацию об операции.	
notifications	Метод получает доступные Пользователю уведомления.	
Управление сценариями авторизации по протоколу OAuth		
oauth/authorize oauth/authorize/certificate	Метод позволяет ИС инициировать процедуру авторизации с аутентификацией Пользователя.	
oauth/endsession	Метод завершает сессию и освобождает токен доступа.	
oauth/token oauth/token/cert	Метод позволяет ИС получить маркер доступа для сценариев «без аутентификации ИС», «с аутентификацией ИС по предварительно распределенному секрету» и «с аутентификацией ИС по сертификату».	
oauth/revocation	Метод отзывает маркер доступа.	
Конфигурация OpenID Connect		
.well-known/openid-configuration	Метод получает получить адреса точек обработки этапов протокола Oauth.	
.well-known/openid-configuration/jwks	Метод получает данные сертификатов, используемых для проверки подписи токенов.	

Приложение 2. Перечень методов интерфейсов фреймворков «КриптоПро Ключ SDK» и «КриптоКлюч SDK», использование которых при разработке систем на основе ПК «КриптоПро Ключ» возможно без дополнительных тематических исследований

КриптоПро Ключ SDK

Таблица 1 — Перечень методов интерфейсов фреймворка **«КриптоПро Ключ SDK»**

Метод	***Описание***	***Ограничения на использование метода***
Инициализация фреймворка		
initBioRng (Android) initRNG (iOS)	Метод открывает окно биологического датчика случайных чисел.	Должен быть вызван перед началом использования остальных методов интерфейса фреймворка.
init (Android) _init (iOS)	Метод инициализации библиотеки.	
isInitialized (Android)	Метод проверки инициализации.	
setCustomLocale (Android)	Метод смены языка.	
setExternalLogger (Android) setLogger (iOS)	Метод устанавливает уровень журналирования.	
setHttpAllTimeOut (Android) setRequestTimeouts (iOS)	Метод установки time-out взаимодействия с сервером.	
setTicketSavePath (Android)	Метод установки директории для записи ticket-файлов.	
getTicketSavePath (Android)	Метод получения директории для записи ticket-файлов.	
checkIntegrity	Метод проведения контроля целостности.	
Аутентификация пользователя, управление ключами аутентификации		
init (Android) normalInit (iOS)	Метод для online регистрации пользователя на сервере. Создает неподтвержденное мобильное устройство (без привязки) в КриптоПро Ключ с получением ключей аутентификации к нему.	В случае использования режима распределенного хранения ключей ЭП разрешено использовать только при указании значения «PASSWORD» в параметре keyProtectionType. Разрешается использование функции с передачей в качестве параметра keyProtectionType значения «BIOMETRIC» только при работе с ключами с автономной защитой и при выполнении требований раздела 5.2. Разрешается использование параметра password только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).

kinit	Метод для online регистрации пользователя на сервере посредством kinit. На сервере создается неподтвержденное мобильное устройство.	В случае использования режима распределенного хранения ключей ЭП разрешено использовать только при указании значения «PASSWORD» в параметре keyProtectionType. Разрешается использование функции с передачей в качестве параметра keyProtectionType значения «BIOMETRIC» только при работе с ключами с автономной защитой и при выполнении требований раздела 5.2. Разрешается использование параметра password только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
addNewDevice	Метод отправляет запрос на регистрацию нового устройства пользователя.	В случае использования режима распределенного хранения ключей ЭП разрешено использовать только при указании значения «PASSWORD» в параметре keyProtectionType. Разрешается использование функции с передачей в качестве параметра keyProtectionType значения «BIOMETRIC» только при работе с ключами с автономной защитой и при выполнении требований раздела 5.2. Разрешается использование параметра password только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
renameAuth	Метод переименования ключей аутентификации.	Разрешено использовать только с указанием значения параметра newName, отличающегося от наименований остальных ключей аутентификации пользователя при их наличии.
removeAuth	Метод удаления ключей аутентификации.	
removeAuthLocal (Android) removeLocalAuth (iOS)	Метод удаления ключей аутентификации без обращения на сервер.	
getAuthList	Метод получения ключей аутентификации.	
confirmNewDevice	Метод подтверждения запроса на добавление нового	Разрешается использование параметра silent только при

	устройства.	использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
checkStatus	Проверка статуса запроса на добавление нового устройства.	
scanQR (Android) scanAndAddQR (iOS)	Метод загрузки данных, переданных в виде QR-кода.	Разрешается использование метода scanQR только с передачей в качестве параметра base64Qr значения null.
currentProtectionType (iOS)	Метод получения способа защиты ключей.	
setPassAuth	Метод для ввода пин-кода *pw* на ключ аутентификации (сессию работы приложения).	Разрешается использование параметра password только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
Verify (Android) verifyDevice (iOS)	Метод подтверждения присоединения мобильного устройства к учетной записи.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
verifyByCert	Метод подтверждения привязки устройства к учетной записи при помощи сертификата.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
confirm	Метод подтверждения установки ключей аутентификации.	
changePassAuth	Метод смены пин-кода *pw* ключей аутентификации.	В случае использования режима распределенного хранения ключей ЭП разрешено использовать только при указании значения «PASSWORD» в параметре keyProtectionType. Разрешается использование функции с передачей в качестве параметра keyProtectionType значения «BIOMETRIC» только при работе с ключами с автономной защитой и при выполнении требований раздела 5.2. Разрешается использование параметров oldPassword и newPassword только при использовании ключей с автономной защитой для поддержки недолговременного хранения

		пин-кода *pw* (подробнее см. описание silent-режима).
Управление сертификатами пользователя		
setCert	Метод установки сертификата.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
getCertList	Метод получения списка запросов на сертификаты и списка сертификатов ключей проверки ЭП.	
setNameCert	Метод установки названия сертификата ключа проверки ЭП.	
setDefaultCert	Метод установки сертификата ключа проверки ЭП сертификатом по умолчанию.	
revokeCert	Метод отзыва сертификата ключа проверки ЭП.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
deleteCert	Метод удаления сертификата или запроса на сертификат ключа проверки ЭП.	Разрешается использование параметров silent и pinCode только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
deleteReqCert (iOS)	Метод удаления запроса на сертификат ключа проверки ЭП.	Разрешается использование параметров silent и pinCode только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
createUnsignedCert (iOS)	Метод создания неподписанного запроса на сертификат ключа проверки ЭП.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
getClientCert (Android) signAndSendCertificateRequest (iOS)	Метод создания ключа ЭП на мобильном устройстве или внешнем носителе, подписание запроса на сертификат и отправка подписанного запроса на сервер для синхронизации.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
createKeyAndSignRequest (Android)	Метод создания ключа ЭП на мобильном устройстве или	Разрешается использование параметра silent только при

	внешнем носителе, подписание запроса на сертификат и отправка подписанного запроса на сервер для синхронизации.	использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
signRequest[6]	Метод создания ключа ЭП на мобильном устройстве или внешнем носителе и подписание запроса на сертификат без отправки на сервер.	
sendSignRequest (Android) sendClientSignedCertificate (iOS)	Метод отправки подписанного запроса на сертификат на сервер для синхронизации.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
installCertificate	Метод установки сертификата в ключевой контейнер на мобильном устройстве или внешнем носителе и отправка его на сервер для синхронизации.	
installCertificateExternal	Метод установки сертификата в мобильное устройство с внешнего носителя (NFC) и отправка его на сервер для синхронизации.	Разрешается использование параметра silent только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
changeKeyPin (Android) changeRutokenPin (iOS)	Метод смены ПИН-кода внешнего носителя (Рутокен).	
forgetKeyPin (Android) removeRutokenPin (iOS)	Метод удаления сохранённого ПИН-кода внешнего носителя (Рутокен).	
checkIfInstalled	Метод проверки установки сертификата в ключевой контейнер на мобильном устройстве.	
checkIfAccessibleOnThisDevice	Метод проверки доступности ключа ЭП (существования ключевого контейнера).	
Операции ЭП		
signMT	Метод подтверждения операции, созданной на сервере.	Разрешается использование параметров silent и pinCode только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
signMO	Метод подтверждения операции, инициированной в мобильном	Разрешается использование параметров silent и pinCode

[6] Данный метод в ОС iOS поддерживает генерацию ключей только с автономной защитой. Для создания запроса сертификата ключа ЭП в режиме распределенного хранения ключей ЭП необходимо использовать метод signAndSendCertificateRequest.

	приложении.	только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
deferredRequest	Метод формирования запроса на подпись без отправки на сервер.	
Управление документами		
uploadDocument	Загрузка документа на сервер.	
downloadDocument	Загрузка документов из сервера.	
getDocumentInfo (Android) downloadDocumentInfo (iOS)	Загрузка информации о документе с сервера.	
downloadDocuments (iOS)	Загрузка списка документов из сервера.	
downloadPdfRawDocument (iOS)	Загрузка PDF представления документа из сервера.	
downloadPdfPreviewDocument (iOS)	Загрузка предпросмотра PDF представления документа из сервера.	
saveDocuments	Загрузка документов из сервера и сохранение в песочнице приложения.	
getListSavedDocuments (Android) listSavedDocument (iOS)	Получение списка сохраненных документов.	
cleanUpDocuments	Удаление сохраненных документов.	
cleanUpDocuments (Android) cleanUpDocument (iOS)	Удаление сохраненного документа.	
Управление ключами ЭП		
listKeys	Получение списка ключей ЭП, установленных на данном мобильном устройстве.	
listExternalKeys	Получение списка ключей ЭП, установленных на внешнем носителе.	
deleteKeyPair	Удаление ключа ЭП, соответствующего определенному сертификату или запросу на сертификат.	
deleteSigningKeyInfo (Android) deleteKeyRecord (iOS)	Удаление сведений о ключе ЭП из SDK.	
deleteKey (iOS)	Удаление информации о ключе, подписанном на устройстве.	
getKey (iOS)	Получение информации о ключе по входным id.	
isExportable (iOS)	Проверка экспортируемости ключа.	
getTitle (iOS)	Получение текста о месте хранения сертификата.	
getAll (iOS)	Получение всех ключевых контейнеров.	
getKey (iOS)	Получение ключа ЭП.	
getContainerFullName (iOS)	Получение полного имени ключевого контейнера.	
isExist (iOS)	Проверка существования ключа ЭП.	

Получение политик		
getOperations	Возвращает список операций, ожидающих действий от пользователя.	
getHistoryOperations	Получение истории операций пользователя на сервисе.	
getParamsDSS	Получение параметров взаимодействия с сервисом.	
getUserDevices	Получение сведений об устройствах пользователя.	
getCaParams	Получение политики сервиса Сервиса Подписи.	
updateDeviceInfo	Обновляет сведения об устройстве пользователя.	

КриптоКлюч SDK

Таблица 2 — Перечень методов интерфейсов фреймворка **«КриптоКлюч SDK»**

Метод	***Описание***	***Ограничения на использование метода***
Класс CertificatesManager. Класс для управления сертификатами.		
createCertificate	Создание запроса на сертификат.	
listCertificates	Получение списка сертификатов и запросов на сертификат.	
listExternalCertificates (iOS)	Получение списка сертификатов со внешнего носителя.	
deleteCertificate	Удаление сертификата, запроса на сертификат.	
setCertificate	Установка сертификата.	
setCertificateFriendlyName	Установка имени сертификата для отображения.	
setDefaultCertificate	Установка сертификата, который будет использоваться по умолчанию.	
revokeCertificate	Запрос на отзыв сертификата.	
Класс CertificatesManagerNonQual. Класс для управления сертификатами для УНЭП.		
signCertificateRequest[7] (Android) sign[7] (iOS)	Подпись запроса на сертификат и отправка в Сервис Подписи.	Разрешается использование параметра pin только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
installCertificate (Android) install (iOS)	Установка сертификата в хранилище приложения.	Разрешается использование параметра pin только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
isCertificateInstalled (Android) checkIfInstalled (iOS)	Проверка установки сертификата в хранилище приложения.	
isCertificateAccessibleOnThisDevice (Android) checkIfAccessibleOnThisDevice (iOS)	Проверка доступности конкретного сертификата для использования на текущем устройстве.	
checkIsOnExternalMedia (iOS)	Проверка установки сертификата на внешнем носителе.	
Класс CKey. Класс для служебных методов.		
initRNG	Открытие окна ДСЧ.	Должен быть вызван перед началом использования остальных методов интерфейса фреймворка.
Init (Android)	Инициализация SDK.	Разрешается использование с передачей в составе

[7] Перед вызовом указанных методов необходимо вызвать метод UsersManager.submitPassword для предъявления пин-кода pw на ключ аутентификации пользователя.

Initialize (iOS)		параметра configuration флага debugMode (iOS) или в качестве параметра RootCertificateType значения Development (Android) только в тестовых целях.
initNonQual (Android)	Инициализация SDK в режиме УНЭП.	Разрешается использование с передачей в качестве параметра RootCertificateType значения Development (Android) только в тестовых целях.
getVersion (Android) version (iOS)	Текущая версия библиотеки.	
getAppearance (Android) appearance (iOS)	Внешний вид интерфейса.	
setHandlesBackgroundTimeout[8] (Android) handlesBackgroundTimeout[8] (iOS)	Установка времени нахождения приложения в фоновом (свернутом) режиме, в течение которого не требуется повторный ввод пин-кода *pw*.	
setLogLevel	Установка уровня журнала события.	
setAlternativeLogger (Android) setLogger (iOS)	Задача внешнего журнала события.	
destroy[9] (Android) reset[9] (iOS)	Завершение использования SDK.	
clearCache	Удаление скаченных документов из кэша приложения.	
getAvailableAPIVersions	Получение список доступных версий API для конкретного класса.	
verifyCkeyIntegrity (Android) verifyControlSum (iOS)	Метод проведения контроля целостности.	
Класс DevicesManager. Класс для управления устройствами пользователя.		
listDevices	Получение с ЦИ списка всех своих устройств.	
revoke	Отзыв (удаление) устройства и соответствующего ему ключа в рамках своей УЗ.	
processAwaitingDevice	Подтверждение или отклонение добавления ключа на новое устройство.	
Класс DevicesManagerNonQual. Класс для управления устройствами пользователя для УНЭП.		
approve	Подтверждение добавления ключа ЭП на новое устройство.	
reject	Отклонение добавления ключа ЭП на новое устройство.	
Класс KeysManagerNonQual. Класс для управления ключами пользователя.		
listKeys	Перечисление всех ключей на устройстве.	

[8] Устанавливаемое время не должно превышать время кэширования пин-кода pw в рабочем состоянии.

[9] После вызова этих методов необходимо выполнить инициализацию библиотеки заново.

createKeyPair	Создание новой ключевой пары для данного пользователя.	Разрешается использование параметра keySource с указанием значения pin только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
getKeysSourceIdentifier (Android)	Получение информации о хранилище ключей по сертификату.	
getKeysForUser (Android)	Перечисление всех ключей пользователя на устройстве.	
deleteKeyPair	Удаление ключевой пары.	
Класс OperationsManager. Класс для управления подписанием.		
getOperationsList	Получение списка операций.	
confirmOperation	Подтверждение операции.	
confirmOperation	Подтверждение операции путем отправки заранее подготовленного запроса на подтверждение.	
getOperationsInfo	Получение сведений об операциях.	
getDocumentDescription	Получение информации о документе.	
signDocuments	Подписание документов.	
signDocuments	Подписание документов путем отправки заранее подготовленного запроса на подпись.	
uploadDocument	Загрузка документов на сервер.	
signDocumentsOffline	Формирование запроса на подпись без отправки на сервер.	
getDocumentBinaryData	Получение бинарных данных документа с серверного компонента.	
Класс OperationsManagerNonQual. Класс для управления подписанием для УНЭП.		
confirmOperation	Подтверждение операции.	
getDocumentPreview	Получение превью документа.	
getDocumentRawPDF	Получение «сырого» документа в формате PDF.	
signDocuments	Подписание документов путем отправки заранее подготовленного запроса на подпись.	
signDocumentsOffline	Формирование запроса на подпись без отправки на сервер.	
signDocuments	Подписание документов.	
Класс PolicyManager. Класс для работы с политиками сервера.		
getParams (Android) getPolicy (iOS)	Запрос параметров сервера.	
getSignParams (Android) getSignServerParams (iOS)	Запрос с сервера КриптоПро Ключ параметров подписания: список профилей ЭП, параметры Удостоверяющих Центров и другое.	
Класс UsersManager. Класс для управления и выполнения действий с УЗ пользователей.		

rename	Переименование пользователя в хранилище SDK.	Разрешено использовать только с указанием значения параметра newName, отличающегося от наименований остальных ключей аутентификации пользователя при их наличии.
listStorage (Android) users (iOS)	Перечисление доступных объектов пользователей.	
delete	Удаление пользователя из хранилища SDK.	
updateStatus	Обновление статуса пользователя информацией с сервера.	
createUserWithInitQR	Создание "неподтвержденной" учётной записи с получением ключей аутентификации (K_{auth}, K_{HMAC}) к ней с использованием QR-кода.	Разрешается использование только с передачей в качестве параметра QRCodeKinit значения «nil» (iOS)/«null» (Android). Разрешается использование с передачей в качестве параметра requirePassword значения «false» только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
createUserWithApproval	Создание запроса на добавление устройства к учётной записи.	Разрешается использование с передачей в качестве параметра requirePassword значения «false» только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
createUser	Создание "неподтвержденной" учетной записи с получением ключей аутентификации к ней.	Разрешается использование с передачей в качестве параметра requirePassword значения «false» только при использовании ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
acceptAccountChanges	Подтверждение присоединения мобильного устройства к учётной записи.	
checkApprovalStatus	Проверка статуса запроса на добавление устройства к учётной записи.	
submitPassword	Предъявление пин-кода на ключ аутентификации K_{auth}.	
changePassword	Изменение пин-кода на ключе аутентификации K_{auth}.	
revoke	Отзыв ключей аутентификации на сервере. Делает ключи недействительными.	

getOperationsHistory	Получение истории операций пользователя на сервисе.	
updateDeviceInfo	Обновляет сведения об устройстве пользователя на серверных компонентах.	
renew	Повторная регистрация пользователя на этом же устройстве.	Разрешается использование только для ключей с автономной защитой.
Класс UsersManagerNonQual. Класс для управления и выполнения действий с УЗ пользователей. Используется для работы с УНЭП.		
createUserWithInitQR	Создание "неподтвержденной" учётной записи в Центре Идентификации с получением ключей аутентификации (K_{auth}, K_{HMAC}) к ней с использованием QR-кода.	Разрешается использование только с передачей в качестве параметра QRCodeKinit значения «nil» (iOS)/«null» (Android). Разрешается использование только для ключей с автономной защитой для недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
createUser	Создание "неподтвержденной" учетной записи с получением ключей аутентификации к ней.	Разрешается использование только для ключей с автономной защитой для недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
acceptAccountChanges	Подтверждение присоединения мобильного устройства к учётной записи.	
createUserWithApproval	Создание запроса на добавление устройства к учетной записи с указанием пин-кода *pw* пользователя.	Разрешается использование только для ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
checkApprovalStatus	Проверка статуса подключения устройства.	
store	Сохранение пользователя в долгосрочной памяти с указанием пин-кода *pw* пользователя.	Разрешается использование только для ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).
submitPassword	Предъявление пин-кода на ключи аутентификации.	
changePassword	Сохранение с новым пин-кодом *pw* на ключи аутентификации.	
revoke	Отзыв ключей аутентификации на сервере. Делает ключи недействительными.	
generateRandomPassword	Генерация пин-кода *pw* пользователя.	
renew	Повторная регистрация пользователя на этом же устройстве с указанием пин-кода *pw* пользователя.	Разрешается использование только для ключей с автономной защитой для поддержки недолговременного хранения пин-кода *pw* (подробнее см. описание silent-режима).

Примечание. Фреймворки поддерживают режим работы silent (silent-режим) — режим, в котором пин-код *pw* может быть сохранен в мобильном приложении и быть использован без необходимости повторного предъявления пользователем. Возможность работы в silent-режиме поддерживается только при работе с ключами с автономной защитой и может быть использована, например, в следующих случаях:

- при использовании СКЗИ в компонентах СФ для автоматического выполнения криптографических функций;
- при недолговременном хранении пин-кода *pw* и для однократного его предъявления в течение периода его хранения.

По умолчанию silent-режим отключен.

Применение silent-режима следует ограничить с учётом модели угроз и нарушителя ИС, в которой применяется СКЗИ. При этом нормативные документы, определяющие создание ИС (включая Техническое задание на разработку и проведение исследований ИС или ее компонентов), могут устанавливать дополнительные требования к построению ключевой системы с проведением установленным порядком её исследований (см. разделы 4.1 и 5.2).

Приложение 3. Перечень вызовов, использование которых для реализации TLS-соединения с одно- и двусторонней аутентификацией при разработке систем на основе ПК «КриптоПро Ключ» возможно без дополнительных тематических исследований

Таблица 1 — Перечень методов языка Java для реализации TLS-соединения с одно- и двусторонней аутентификацией под управлением ОС Android

Метод	***Описание***	***Ограничения на использование метода***
Функции установки TLS-соединения и приёма/передачи данных		
initClientSSL объекта ru.CryptoPro.ssl. TLSContext	Функция создаёт и инициализирует объект SSLContext для клиентской стороны TLS-соединения с односторонней аутентификацией.	Разрешена при указании в качестве параметра tlsProvider значения «JTLS». Должна быть вызвана в случае TLS с односторонней аутентификацией. Перед использованием системная переменная «tls_prohibit_disabled_validation» должна быть установлена в значение «True».
initAuthClientSSL объекта ru.CryptoPro.ssl.android. util.TLSContext	Функция создаёт и инициализирует объект SSLContext для клиентской стороны TLS-соединения с двусторонней аутентификацией.	Разрешена при выполнении следующих условий: • должна быть исключена параллельная установка TLS-соединений с двусторонней аутентификацией другими TLS-клиентами и параллельная установка TLS-соединений с односторонней аутентификацией текущим TLS-клиентом; • в качестве tlsProvider и keyStoreProvider должны быть указаны значения «JTLS» и «JCSP» соответственно; Должна быть вызвана в случае TLS с двусторонней аутентификацией. Перед использованием системная переменная «tls_prohibit_disabled_validation» должна быть установлена в значение «True».
getSocketFactory объекта SSLContext	Функция создаёт объект SSLSocketFactory на основе текущего SSLContext.	Разрешена только для объекта SSLContext, полученного разрешённым вызовом initClientSSL или initAuthClientSSL объекта ru.CryptoPro.ssl.android.util.TLSContext
createSocket объекта SSLSocketFactory	Функция создаёт программный сокет с указанием физического адреса сервера.	Разрешена только для объекта SSLSocketFactory, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL и SSLContext.getSocketFactory.
connect объекта SSLSocket	Функция устанавливает связь сокета с сервером по указанному адресу.	Разрешена только для объекта SSLSocket, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.getSocketFactory и SSLSocketFactory.createSocket.
startHandshake объекта SSLSocket	Функция устанавливает TLS-соединение или проводит RENEGOTIATION.	Разрешена только для объекта SSLSocket, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.getSocketFactory и SSLSocketFactory.createSocket.
getOutputStream объекта SSLSocket	Функция получает доступ к буферу OutputStream для отправки данных	Разрешена только для объекта SSLSocket, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или

Метод	***Описание***	***Ограничения на использование метода***
	TLS-серверу.	TLSContext.initAuthClientSSL, SSLContext.getSocketFactory и SSLSocketFactory.createSocket.
getInputStream объекта SSLSocket	Функция получает доступ к буферу InputStream для получения данных от TLS-сервера.	Разрешена только для объекта SSLSocket, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.getSocketFactory и SSLSocketFactory.createSocket.
write объекта OutputStream	Функция отправляет данные TLS-серверу.	Разрешена только для объекта OutputStream, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.getSocketFactory, SSLSocketFactory.createSocket и SSLSocket.getOutputStream. Значение провайдера по умолчанию (параметра "ru.CryptoPro. defaultSSLProv") должно быть равно «JCSP».
read объекта InputStream	Функция получает данные от TLS-сервера.	Разрешена только для объекта InputStream, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL, SSLSocketFactory.createSocket и SSLSocket.getInputStream. Значение провайдера по умолчанию (параметра "ru.CryptoPro. defaultSSLProv") должно быть равно «JCSP».
close объекта SSLSocket	Функция закрывает TLS-соединение.	Разрешена только для объекта SSLSocket, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.getSocketFactory и SSLSocketFactory.createSocket.
createSSLEngine класса SSLContext	Функция создаёт объект SSLEngine на основе текущего SSLContext.	Разрешено использовать только для объекта SSLContext, полученного разрешённым вызовом initClientSSL или initAuthClientSSL объекта ru.CryptoPro.ssl.util.TLSContext. Для полученного объекта SSLEngine нужно установить режим TLS-клиента вызовом его метода setUseClientMode.
setUseClientMode класса SSLEngine	Функция устанавливает роли работы по протоколу TLS (клиент/сервер).	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
isOutboundDone класса SSLEngine	Функция проверяет возможность отправки и отсутствие неотправленных по протоколу TLS данных.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
isInboundDone класса SSLEngine	Функция проверяет возможность получения данных по протоколу TLS.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
beginHandshake класса SSLEngine	Функция устанавливает TLS-соединения или проводит RENEGOTIATION.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
wrap класса SSLEngine	Функция обрабатывает и	Разрешено использовать только для объекта

Метод	*Описание*	*Ограничения на использование метода*
	отправляет данные второй стороне TLS-соединения.	SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
getHandshakeStatus класса SSLEngine	Функция получает текущее состояние процесса установки TLS-соединения.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
getStatus класса SSLEngineResult	Функция получает код ошибки выполнения последнего предыдущего метода SSLEngine.	Разрешено использовать только для объекта SSLEngineResult, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.createSSLEngine и SSLEngine.wrap или SSLEngine.unwrap.
getDelegatedTask класса SSLEngine	Функция получает объект задачи, запуск которой необходим для продолжения установки TLS-соединения.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
unwrap класса SSLEngine	Функция получает и обрабатывает данные от второй стороны TLS-соединения.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine. Разрешается использовать только с проведением дополнительной проверки корректности назначения KU полученного сертификата TLS-сервера. Значение провайдера по умолчанию (параметра «ru.CryptoPro.defaultSSLProv») должно быть равно «JCSP».
closeOutbound класса SSLEngine	Функция приостанавливает отправку любых данных и закрывает TLS-соединение.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
closeInbound класса SSLEngine	Функция приостанавливает отправку и получение любых данных и закрывает TLS-соединение.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
getSession класса SSLEngine	Функция получает доступ к объекту с параметрами текущего соединения SSLSession.	Разрешено использовать только для объекта SSLEngine, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL (или TLSContext.initAuthClientSSL) и SSLContext.createSSLEngine.
getApplicationBufferSize класса SSLSession	Функция получает максимальный размер передаваемых в одном TLS-пакете данных прикладного уровня.	Разрешено использовать только для объекта SSLSession, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.createSSLEngine и SSLEngine.getSession.
getPacketBufferSize класса SSLSession	Функция получает максимальный размер TLS-пакета.	Разрешено использовать только для объекта SSLSession, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.createSSLEngine и SSLEngine.getSession.
getPeerPrincipal класса	Функция получает имя владельца сертификата	Разрешено использовать только для объекта SSLSession, полученного комбинацией

Метод	***Описание***	***Ограничения на использование метода***
SSLSession	(второй стороны TLS-соединения), полученного в рамках обмена Handshake.	разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.createSSLEngine и SSLEngine.getSession.
getPeerCertificates класса SSLSession	Функция получает цепочку сертификатов (второй стороны TLS-соединения), полученную в рамках обмена Handshake.	Разрешено использовать только для объекта SSLSession, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.createSSLEngine и SSLEngine.getSession.
getLocalPrincipal класса SSLSession	Функция получает имя владельца сертификата (данной стороны TLS-соединения), отправленное в рамках обмена Handshake.	Разрешено использовать только для объекта SSLSession, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.createSSLEngine и SSLEngine.getSession.
getLocalCertificates класса SSLSession	Функция получает цепочку сертификатов (данной стороны TLS-соединения), отправленную в рамках обмена Handshake.	Разрешено использовать только для объекта SSLSession, полученного комбинацией разрешённых вызовов TLSContext.initClientSSL или TLSContext.initAuthClientSSL, SSLContext.createSSLEngine и SSLEngine.getSession.

Таблица 2 — Перечень методов языка C++ для реализации TLS-соединения с одно- и двусторонней аутентификацией

Метод	***Описание***	***Ограничения на использование метода***
Функции установки TLS-соединения и приёма/передачи данных		
curl_global_init	Функция инициализации работы библиотеки.	Не является потокобезопасной, должна быть вызвана единожды перед началом использования библиотеки. После завершения использования библиотеки необходимо вызвать curl_global_cleanup
curl_global_cleanup	Функция деинициализации работы библиотеки.	Не является потокобезопасной, должна быть вызвана единожды после завершения использования библиотеки.
curl_easy_init	Функция инициализации сессии работы с библиотекой.	Является потокобезопасной. Полученный данным вызовом хэндл должен быть после завершения работы с ним освобождён вызовом curl_easy_cleanup.
curl_easy_cleanup	Функция освобождения ресурсов, занятых в рамках указанной сессии работы с библиотекой.	Является потокобезопасной.

Метод	***Описание***	***Ограничения на использование метода***
curl_easy_setopt	Функция меняет указанный параметр сессии работы с библиотекой.	Разрешено использовать только с указанным перечнем параметров[10].
curl_easy_getinfo	Функция получения сведений о соединении.	
curl_easy_perform	Функция синхронизируемой передачи данных по заданному дескриптору	Используемый дескриптор CURL должен быть получен разрешённым вызовом *curl_easy_init.* Перед вызовом данной функции с помощью *curl_easy_setopt* и параметров **CURLOPT_STRICT_GOST, CURLOPT_NOPROXY, CURLOPT_REDIR_PROTOCOLS, CURLOPT_FTPSSLAUTH** необходимо: • задать проведение проверки алгоритма ключа проверки ЭП для всех сертификатов в цепочке сертификата сервера; • запретить использовать прокси-сервер при соединении с указанным TLS-сервером; • ограничить возможность redirect-перехода с TLS-соединения на незащищенное; • задать использование протокола TLS для защиты канала FTPS-соединения.

[10] CURLOPT_STRICT_GOST, CURLOPT_DNS_CACHE_TIMEOUT, CURLOPT_SSL_CIPHER_LIST, CURLOPT_PROXY_SSL_CIPHER_LIST, CURLOPT_MAXCONNECTS, CURLOPT_FORBID_REUSE, CURLOPT_FRESH_CONNECT, CURLOPT_HEADER, CURLOPT_NOPROGRESS, CURLOPT_NOBODY, CURLOPT_FAILONERROR, CURLOPT_KEEP_SENDING_ON_ERROR, CURLOPT_UPLOAD, CURLOPT_PUT, CURLOPT_REQUEST_TARGET, CURLOPT_FILETIME, CURLOPT_SERVER_RESPONSE_TIMEOUT, CURLOPT_TFTP_NO_OPTIONS, CURLOPT_TFTP_BLKSIZE, CURLOPT_NETRC, CURLOPT_NETRC_FILE, CURLOPT_TRANSFERTEXT, CURLOPT_TIMECONDITION, CURLOPT_TIMEVALUE, CURLOPT_TIMEVALUE_LARGE, CURLOPT_SSLVERSION, CURLOPT_PROXY_SSLVERSION, CURLOPT_AUTOREFERER, CURLOPT_ACCEPT_ENCODING, CURLOPT_TRANSFER_ENCODING, CURLOPT_FOLLOWLOCATION, CURLOPT_UNRESTRICTED_AUTH, CURLOPT_MAXREDIRS, CURLOPT_POSTREDIR, CURLOPT_POST, CURLOPT_COPYPOSTFIELDS, CURLOPT_POSTFIELDS, CURLOPT_POSTFIELDSIZE, CURLOPT_POSTFIELDSIZE_LARGE, CURLOPT_HTTPPOST, CURLOPT_REFERER, CURLOPT_USERAGENT, CURLOPT_HTTPHEADER, CURLOPT_PROXYHEADER, CURLOPT_HEADEROPT, CURLOPT_HTTP200ALIASES, CURLOPT_COOKIE, CURLOPT_COOKIEFILE, CURLOPT_COOKIEJAR, CURLOPT_COOKIESESSION, CURLOPT_COOKIELIST, CURLOPT_HTTPGET, CURLOPT_HTTP_VERSION, CURLOPT_EXPECT_100_TIMEOUT_MS, CURLOPT_HTTP09_ALLOWED, CURLOPT_HTTPAUTH, CURLOPT_CUSTOMREQUEST, CURLOPT_PROXYPORT, CURLOPT_PROXYAUTH, CURLOPT_SOCKS5_AUTH, CURLOPT_SOCKS5_GSSAPI_NEC,CURLOPT_SOCKS5_GSSAPI_SERVICE, CURLOPT_PROXY_SERVICE_NAME, CURLOPT_SERVICE_NAME, CURLOPT_HEADERDATA, CURLOPT_ERRORBUFFER, CURLOPT_WRITEDATA, CURLOPT_DIRLISTONLY, CURLOPT_APPEND, CURLOPT_FTP_FILEMETHOD, CURLOPT_FTPPORT, CURLOPT_FTP_USE_EPRT, CURLOPT_FTP_USE_EPSV, CURLOPT_FTP_USE_PRET, CURLOPT_FTP_SKIP_PASV_IP, CURLOPT_FTP_ACCOUNT, CURLOPT_FTP_ALTERNATIVE_TO_USER, CURLOPT_FTPSSLAUTH, CURLOPT_FTP_CREATE_MISSING_DIRS, CURLOPT_READDATA, CURLOPT_INFILESIZE, CURLOPT_INFILESIZE_LARGE, CURLOPT_LOW_SPEED_LIMIT, CURLOPT_MAX_SEND_SPEED_LARGE, CURLOPT_MAX_RECV_SPEED_LARGE, CURLOPT_LOW_SPEED_TIME, CURLOPT_URL, CURLOPT_PORT, CURLOPT_TIMEOUT, CURLOPT_TIMEOUT_MS, CURLOPT_CONNECTTIMEOUT, CURLOPT_CONNECTTIMEOUT_MS, CURLOPT_ACCEPTTIMEOUT_MS, CURLOPT_USERPWD, CURLOPT_USERNAME, CURLOPT_PASSWORD, CURLOPT_LOGIN_OPTIONS, CURLOPT_XOAUTH2_BEARER, CURLOPT_POSTQUOTE, CURLOPT_PREQUOTE, CURLOPT_QUOTE, CURLOPT_RESOLVE, CURLOPT_PROGRESSFUNCTION, CURLOPT_XFERINFOFUNCTION, CURLOPT_PROGRESSDATA, CURLOPT_PROXYUSERPWD, CURLOPT_PROXYUSERNAME, CURLOPT_PROXYPASSWORD, CURLOPT_NOPROXY, CURLOPT_RANGE, CURLOPT_RESUME_FROM, CURLOPT_RESUME_FROM_LARGE, CURLOPT_DEBUGFUNCTION, CURLOPT_DEBUGDATA, CURLOPT_HEADERFUNCTION, CURLOPT_WRITEFUNCTION, CURLOPT_READFUNCTION, CURLOPT_SEEKFUNCTION, CURLOPT_SEEKDATA, CURLOPT_IOCTLFUNCTION, CURLOPT_IOCTLDATA, CURLOPT_SSLCERT, CURLOPT_PROXY_SSLCERT, CURLOPT_SSLCERTTYPE, CURLOPT_PROXY_SSLCERTTYPE, CURLOPT_CRLF, CURLOPT_HAPROXYPROTOCOL, CURLOPT_INTERFACE, CURLOPT_LOCALPORT, CURLOPT_LOCALPORTRANGE, CURLOPT_CERTINFO, CURLOPT_PINNEDPUBLICKEY, CURLOPT_PROXY_PINNEDPUBLICKEY, CURLOPT_CAINFO, CURLOPT_PROXY_CAINFO, CURLOPT_TELNETOPTIONS, CURLOPT_BUFFERSIZE, CURLOPT_UPLOAD_BUFFERSIZE, CURLOPT_NOSIGNAL, CURLOPT_PRIVATE, CURLOPT_MAXFILESIZE, CURLOPT_USE_SSL, CURLOPT_IPRESOLVE, CURLOPT_MAXFILESIZE_LARGE, CURLOPT_TCP_NODELAY, CURLOPT_IGNORE_CONTENT_LENGTH, CURLOPT_CONNECT_ONLY, CURLOPT_SOCKOPTFUNCTION, CURLOPT_SOCKOPTDATA, CURLOPT_CLOSESOCKETFUNCTION, CURLOPT_RESOLVER_START_FUNCTION, CURLOPT_RESOLVER_START_DATA, CURLOPT_CLOSESOCKETDATA, CURLOPT_SSL_SESSIONID_CACHE, CURLOPT_HTTP_TRANSFER_DECODING, CURLOPT_HTTP_CONTENT_DECODING, CURLOPT_NEW_FILE_PERMS, CURLOPT_ADDRESS_SCOPE, CURLOPT_PROTOCOLS, CURLOPT_REDIR_PROTOCOLS, CURLOPT_DEFAULT_PROTOCOL, CURLOPT_MAIL_FROM, CURLOPT_MAIL_AUTH, CURLOPT_MAIL_RCPT, CURLOPT_SASL_IR, CURLOPT_RTSP_REQUEST, CURLOPT_RTSP_SESSION_ID, CURLOPT_RTSP_STREAM_URI, CURLOPT_RTSP_TRANSPORT, CURLOPT_RTSP_CLIENT_CSEQ, CURLOPT_INTERLEAVEDATA, CURLOPT_INTERLEAVEFUNCTION, CURLOPT_WILDCARDMATCH, CURLOPT_CHUNK_BGN_FUNCTION, CURLOPT_CHUNK_END_FUNCTION, CURLOPT_FNMATCH_FUNCTION, CURLOPT_CHUNK_DATA, CURLOPT_FNMATCH_DATA, CURLOPT_TCP_KEEPALIVE, CURLOPT_TCP_KEEPIDLE, CURLOPT_TCP_KEEPINTVL, CURLOPT_TCP_FASTOPEN, CURLOPT_UNIX_SOCKET_PATH, CURLOPT_ABSTRACT_UNIX_SOCKET, CURLOPT_PATH_AS_IS, CURLOPT_PIPEWAIT, CURLOPT_CONNECT_TO, CURLOPT_SUPPRESS_CONNECT_HEADERS, CURLOPT_HAPPY_EYEBALLS_TIMEOUT_MS, CURLOPT_DNS_SHUFFLE_ADDRESSES, CURLOPT_DISALLOW_USERNAME_IN_URL, CURLOPT_DOH_URL, CURLOPT_MAXAGE_CONN, CURLOPT_TRAILERFUNCTION, CURLOPT_TRAILERDATA

Метод	***Описание***	***Ограничения на использование метода***
		При установке TLS-соединения с двусторонней аутентификацией перед вызовом данной функции с помощью *curl_easy_setopt* и параметра **CURLOPT_PROXY_SSLCERT** и/или **CURLOPT_SSLCERT** необходимо указать адрес сертификата TLS-клиента в системном хранилище сертификатов.
curl_easy_send	Функция отправляет данные типа raw-data по установленному ранее каналу	Используемый дескриптор CURL должен быть получен вызовом curl_easy_init. Для вызова функции необходимо предварительно вызвать *curl_easy_setopt* (с указанием флаг **CURLOPT_CONNECT_ONLY**) и *curl_easy_perform.*
curl_easy_recv	Функция принимает данные типа raw-data по установленному ранее каналу	Используемый дескриптор CURL должен быть получен вызовом *curl_easy_init.* Для вызова функции необходимо предварительно вызвать *curl_easy_setopt* (с указанием флаг **CURLOPT_CONNECT_ONLY**) и *curl_easy_perform.*
Вспомогательные функции обработки строк и данных		
curl_easy_escape	Функция перевода заданной строки в URL-кодировку	
curl_easy_unescape	Функция перевода заданной URL-кодированной строки в строку символов char	
curl_formadd	Функция добавления новой секции в состав данных типа multipart/formdata, отправляемых по HTTP POST-запросу	
curl_formfree	Функция освобождения ресурсов, выделенных для данных типа multipart/formdata, составленных ранее набором вызовов curl_formadd.	
curl_slist_append	Функция добавляет строку в список строк структуры curl_slist.	
curl_slist_free_all	Функция освобождает все ресурсы, занятые структурой curl_slist.	