

Утвержден

RU.54819512.00021-01 34 01-ЛУ

Программный комплекс
«ОТР. Универсальный сервер криптографической защиты
информации»
(ОТР.УСКЗИ)

RU.54819512.00021-01

РУКОВОДСТВО АДМИНИСТРАТОРА

Листов: 77

АННОТАЦИЯ

В данном программном документе приведено описание установки, настройки и эксплуатации программного комплекса «ОТР. Универсальный сервер криптографической защиты информации» (далее по тексту – ПК «ОТР.УСКЗИ», Изделие).

Документ предназначен для администратора Изделия.

СОДЕРЖАНИЕ

АННОТАЦИЯ	2
1. ВВЕДЕНИЕ	5
1.1. Область применения	5
1.2. Возможности системы	5
1.3. Требования к пользователям	5
1.4. Условные обозначения и сокращения	5
2. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ	7
2.1. Виды деятельности, функции	7
2.2. Минимальные программные и аппаратные требования к системе	7
2.3. Минимальные сетевые требования	7
3. ПРЕДВАРИТЕЛЬНЫЕ НАСТРОЙКИ СИСТЕМЫ.....	8
3.1. Установка и настройка СУБД Postgres.....	8
3.2. Установка и настройка Docker	9
3.2.1. Установка Docker	9
3.2.2. Установка Docker-compose	10
3.3. Установка и настройка Криптопро CSP	10
3.4. Установка JAVA (openJdk-11).....	11
3.5. Контроль целостности	11
4. ПОЛНОЕ ОПИСАНИЕ НАСТРОЕК СЕРВИСОВ	13
4.1. Сервис расчета хеша и случайных последовательностей байт (message-digest-service)	13
4.2. Сервис проверки сертификатов и CRL (certificate-verify-service)	14
4.3. Сервис создания подписи (create-signature-service)	17
4.4. Сервис проверки подписи (verify-signature-service).....	21
4.5. Мост интеграции kafka с сервисом подписания (create-signature-bridge).....	25
4.6. Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge).....	26
4.7. Backend-модуль сервиса управления (control-module-backend)	27
5. НАСТРОЙКА СЕРВИСОВ	31
5.1. Сервис расчета хеша и случайных последовательностей байт (message-digest-service)	31
5.2. Сервис проверки сертификатов и CRL (certificate-verify-service)	31
5.3. Сервис создания подписи (create-signature-service)	31
5.4. Сервис проверки подписи (verify-signature-service).....	32
5.5. Мост интеграции kafka с сервисом подписания (create-signature-bridge).....	33
5.6. Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge).....	33
5.7. Backend-модуль сервиса управления (control-module-backend)	33
5.8. Настройка логирования в ELK-стек (в каждом сервисе)	35
6. НАСТРОЙКА ВСПОМОГАТЕЛЬНЫХ СЕРВИСОВ	38
6.1. Настройка необходимых подсетей	38
6.2. Настройка вспомогательных компонент.....	38
7. ПОРЯДОК ЗАПУСКА.....	39
8. РАБОТА С КЛЮЧАМИ И СЕРТИФИКАТАМИ	40
ПРИЛОЖЕНИЕ 1. ФАЙЛЫ ЗАПУСКА ВСПОМОГАТЕЛЬНЫХ КОМПОНЕНТ.....	41
ПРИЛОЖЕНИЕ 2. ПЕРЕЧЕНЬ ФАЙЛОВ ИЗДЕЛИЯ ДЛЯ КОНТРОЛЯ	44
Сервис расчета хеша и случайных последовательностей байт (message-digest-service)	44
Сервис проверки сертификатов и CRL (certificate-verify-service).....	48

Сервис создания подписи (create-signature-service).....	54
Сервис проверки подписи (verify-signature-service)	59
Мост интеграции kafka с сервисом подписания (create-signature-bridge)	65
Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge)	68
Backend-модуль сервиса управления (control-module-backend).....	72
СПИСОК ИЗМЕНЕНИЙ	77

1. ВВЕДЕНИЕ

1.1. Область применения

В документе приведено описание процессов установки и настройки, а также эксплуатации ПК «ОТР.УСКЗИ».

1.2. Возможности системы

Раздел в рамках данного документа не предусмотрен.

1.3. Требования к пользователям

Администраторы, выполняющие работы по настройке Изделия должны обладать правами администратора на уровне операционной системы.

1.4. Условные обозначения и сокращения

Условные обозначения и сокращения, используемые в рамках документа, перечислены в таблице 1.

Таблица 1. Условные обозначения и сокращения

№ п/п	Термин	Содержание
1.	Закрытый ключ	Ключ электронной подписи. Уникальная последовательность символов, предназначенная для создания электронной подписи.
2.	ПК «ОТР.УСКЗИ»	Программный комплекс «ОТР. Универсальный сервер криптографической защиты информации».
3.	Объект	Объекты системы: документы, значения справочников, настройки и т.д.
4.	Параметры	Данные полей документа, справочников и т.п.
5.	ПС	Прикладная система
6.	СКП (сертификат)	Квалифицированный сертификат ключа проверки электронной подписи
7.	УЦ	Удостоверяющий центр.
8.	УЭП	Усовершенствованная квалифицированная электронная подпись. Обеспечивает юридическую значимость документооборота путем добавления к электронной подписи информации, необходимой для её локальной проверки и валидности сертификата на момент подписи.
9.	ЭП	Квалифицированная электронная подпись.
10.	ЭФ	Экранная форма.

Элементы оформления

Название файлов выделяются в тексте следующим образом: setup.exe.

Сведения, которые указывают на особенности данного описания, оформлены в виде примечаний, начинающихся со слова «**Примечание**». Примечание может предшествовать целой главе, разделу, подразделу или следовать непосредственно за элементом, к которому оно относится.

Сведения, имеющие критический характер для работы системы или пользователя, оформлены в виде примечаний, которые начинаются со слова «**Внимание!**». Как правило, эта информация имеет предупреждающий характер.

2. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ

2.1. Виды деятельности, функции

ПК «ОТР.УСКЗИ» – программный комплекс, предоставляющий автоматическом режиме сервисы формирования и проверки электронной подписи (далее – ЭП), а также доведения ЭП до усовершенствованной.

Взаимодействие с Изделием должно осуществляться от имени служебных пользователей.

2.2. Минимальные программные и аппаратные требования к системе

Минимальные требования к аппаратному обеспечению указаны в таблице 2.

Таблица 2. Минимальные требования к аппаратному обеспечению

Параметр	Значение
ЦП	8 ядра
ОЗУ	32 GB
Жесткий диск	150 GB

Требования к программному обеспечению указаны в таблице 3.

Таблица 3. Требование к программному обеспечению

Параметр	Значение
Операционная система	RHEL, релиз 7 CentOS, релиз 7
СУБД	PostgreSQL 12 и выше
JRE (Java SE Runtime Environment)	11 и выше
CSP (cryptography service provider)	СКЗИ «КриптоПро CSP» версия 4.0 исполнения 1-Base и 2-Base (для ОС 64 bit)

2.3. Минимальные сетевые требования

Для всех серверов, участвующие в обмене данными с Изделием должна быть настроена маршрутизация, а также открыты порты, указанные в качестве входящих в конфигурации сервисов.

3. ПРЕДВАРИТЕЛЬНЫЕ НАСТРОЙКИ СИСТЕМЫ

Изделие представляет собой набор микросервисов, разворачиваемых в среде RHEL с использованием JVM. В качестве БД используется СУБД Postgres версии 12.

К операционной системе должен быть примонтирован раздел «/opt», в который будут устанавливаться основные компоненты.

Требуется наличие доступа к сети интернет (для установки дополнительных пакетов ОС и БПО).

3.1. Установка и настройка СУБД Postgres

Необходимо установить БД Postgres версии 12.

Для этого необходимо выполнить следующие шаги:

1. Выполнить команды установки Postgres:

```
yum install -y https://download.postgresql.org/pub/repos/yum/reposrms/EL-7-
x86_64/pgdg-redhat-repo-latest.noarch.rpm
yum install -y postgresql12-server
```

2. Создать каталог, в котором будут размещены табличные пространства и назначить ему права для пользователя postgres:

```
mkdir -p /opt/pgdata
chown -R postgres:postgres /opt/pgdata
chmod -R 750 /opt/pgdata/
```

3. Далее необходимо установить переменную окружения «PGDATA», в которой будет указано, куда устанавливаются файл табличного пространства Postgres и прочие конфигурационные файлы:

- 3.1. Открыть на редактирование настройки службы:

```
systemctl edit postgresql-12.service
```

- 3.2. Внести туда следующие строки и сохранить:

```
[Service]
Environment=PGDATA=/opt/pgdata
```

- 3.3. Обновить службы:

```
systemctl daemon-reload
```

4. Инициализировать базу данных:

```
/usr/pgsql-12/bin/postgresql-12-setup initdb
```

5. Выполнить настройки доступа:

- 5.1. В файле «/opt/pgdata/pg_hba.conf» добавить строчку «all» для соответствующих подсетей, например:

```
host all all 192.168.0.0/24 md5
```

- 5.2. В файле «/opt/pgdata/postgresql.conf» добавить разрешение доступа с адресов. Для этого найти параметр «listen_addresses», раскомментировать его и установить значение «*»

```
listen_addresses = '*'
```

- 5.3. Также в файле «/opt/pgdata/postgresql.conf» возможно выполнить настройки оптимизации (найти параметры и обновить значения):

```
max_connections = 500
shared_buffers = 1GB
```

Для подбора оптимизационных параметров можно воспользоваться ресурсом <https://pgtune.leopard.in.ua/#/>

6. Выполнить команды запуска сервиса БД:

```
systemctl enable postgresql-12
systemctl start postgresql-12
```

7. Пользователю ОС postgres задать пароль (например, postgres):

```
passwd postgres
```

8. Зайти под пользователем postgres (su postgres) в ОС и задать пароль пользователю БД. Для этого выполнить команды:

```
psql
ALTER USER postgres PASSWORD 'postgres' ;
```

9. В файле «`/opt/pgdata/pg_hba.conf`» изменить тип подключения для локальных подключений:

```
host all all 127.0.0.1/32 md5
host all all ::1/128 md5
```

10. Перезапустить службу БД:

```
systemctl restart postgresql-12
```

Далее необходимо через pgAdmin создать БД «crypto» в кодировке «ru_RU.UTF-8» и выполнить в ней импорт базовой схемы (restore):

```
CREATE DATABASE crypto
WITH
  OWNER = postgres
  ENCODING = 'UTF8'
  TEMPLATE = template0
  LC_COLLATE = 'ru_RU.UTF-8'
  LC_CTYPE = 'ru_RU.UTF-8'
  TABLESPACE = pg_default
  CONNECTION LIMIT = -1;
```

3.2. Установка и настройка Docker

Для работы общесистемного ПО необходимо установить средство контейнеризации Docker и менеджер конфигурации запуска контейнеров Docker-compose.

3.2.1. Установка Docker

Необходимо выполнить следующие шаги:

1. Выполнить удаление предыдущих версий, если они есть. Для этого выполнить команду:

```
yum remove docker \
           docker-client \
           docker-client-latest \
           docker-common \
           docker-latest \
           docker-latest-logrotate \
           docker-logrotate \
           docker-engine
```

- Установить репозиторий, из которого будет устанавливаться Docker. Для этого выполнить команды:

```
yum install -y yum-utils
yum-config-manager \
  --add-repo \
  https://download.docker.com/linux/centos/docker-ce.repo
```

- Установить основной движок Docker. Для этого выполнить команду:

```
yum install -y docker-ce docker-ce-cli containerd.io
```

- Выполнить настройку расположения docker. Для этого:

- Создать каталог «`/opt/docker`»:

```
mkdir -p /opt/docker
```

- Отредактировать/создать файл «`/etc/docker/daemon.json`»:

```
{
  "graph": "/opt/docker",
  "storage-driver": "overlay"
}
```

- Выполнить команду автоматического запуска docker при старте системы и запустить docker:

```
systemctl enable docker
systemctl start docker
```

Убедиться в корректности установки можно, выполнив команду запуска тестового контейнера docker:

```
docker run hello-world
```

3.2.2. Установка Docker-compose

Для установки Docker-compose необходимо выполнить следующие шаги:

- Выполнить команду загрузки Docker-compose

```
curl -L "https://github.com/docker/compose/releases/download/1.27.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

- Выполнить команду установки разрешений

```
chmod +x /usr/local/bin/docker-compose
```

- Выполнить команду создания ссылки:

```
ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

Проверить корректность установки можно выполнив команду:

```
docker-compose version
```

3.3. Установка и настройка Криптопро CSP

Для установки Криптопро CSP необходимо выполнить следующие шаги:

- Скачать с официального сайта архив с Криптопро версии 4.0.9963 (Abel) для операционной системы RadHat (rpm-пакет)
- Скопировать данный архив на сервер и распаковать при помощи команды:

```
tar -zxvf ./linux-amd64.tgz
```

3. Выполнить установку вспомогательных пакетов. Для этого выполнить команды:

```
yum install -y lsb-core-noarch
yum clean all
```

4. Перейти в разархивированный каталог и выполнить команду:

```
sh ./install.sh kcl
```

5. Выйти из каталога, удалить его и архив с дистрибутивом
6. Выполнить команду обновления переменной окружения PATH

```
PATH=/opt/cproccsp/bin/amd64:/opt/cproccsp/sbin/amd64:$PATH
```

7. Скопировать на сервер каталог с корневыми и промежуточными сертификатами (для тех личных сертификатов, которыми предполагается серверное подписание) и осуществить их импорт.

- 7.1. Для каждого корневого сертификата выполнить команду

```
certmgr -install -store mRoot -file имя_файла_корневого_сертификата
```

- 7.2. Для каждого промежуточного сертификата выполнить команду

```
certmgr -install -store mCa -file имя_файла_промежуточного_сертификата
```

8. Скопировать контейнеры с закрытыми ключами в каталог (где вместо root указать имя пользователя, под которым запускаются сервисы и установить owner соответствующего пользователя):

```
/var/opt/cproccsp/keys/root
```

9. Зайти под пользователем, под которым запускаются сервисы и выполнить импорт сертификатов из закрытых ключей, для этого выполнить команду:

```
csptestf -absorb -certs -autoprov
```

Если команда вызывает ошибку, что csptestf не найден, то предварительно выполнить команду обновления переменной окружения PATH

```
PATH=/opt/cproccsp/bin/amd64:/opt/cproccsp/sbin/amd64:$PATH
```

10. В случае необходимости установки лицензии, это можно сделать, выполнив команду:

```
cpconfig -license -set номер_лицензии
```

3.4. Установка JAVA (openJdk-11)

Для установки JAVA необходимо:

1. Выполнить команду:

```
yum install -y java-11-openjdk-headless
```

2. Выставить JAVA по умолчанию:

```
alternatives --config java
```

и в списке выбрать новую java java-11-openjdk

3.5. Контроль целостности

Контролю целостности подлежат исполняемые библиотеки java (jar-библиотеки, перечисленные в Приложении «2. Перечень файлов изделия для контроля»).

Контроль целостности реализован по следующей схеме:

- При сборке проекта к каждой jar-библиотеке применяются следующие действия:
 - Считается контрольная сумма (в формате «Adler-32»), по всем файлам, входящим в состав jar-библиотеки;
 - Рассчитанная контрольная сумма зашифровывается при помощи закрытого RSA-ключа (закрытый ключ хранится только на сборочном конвейере);
 - Зашифрованная контрольная сумма располагается внутри jar-библиотеки по пути «META-INF/checksum.sig»;
- При старте сервиса для каждой jar-библиотеки осуществляется контроль на соответствия контрольным суммам, сохранённым при сборке на сборочном конвейере. Для этого применяются следующие действия:
 - Считается контрольная сумма (в формате «Adler-32») по всем файлам, входящим в состав jar-библиотеки;
 - Осуществляется проверка наличия в jar-библиотеке наличия зашифрованной контрольной суммы;
 - Из файла «META-INF/checksum.sig» извлекается зашифрованная контрольная сумма;
 - Осуществляется расшифрование контрольной суммы при помощи открытого RSA-ключа, соответствующего закрытому RSA-ключу, которым контрольная сумма была зашифрована;
 - Осуществляется сравнение посчитанной и сохраненной контрольных сумм.

4. ПОЛНОЕ ОПИСАНИЕ НАСТРОЕК СЕРВИСОВ

В разделе представлено полное описание возможных настроек сервисов. Описание необходимых настроек представлено в следующем разделе.

Представлены следующие сервисы:

- Сервис расчета хеша и случайных последовательностей байт (message-digest-service)
- Сервис проверки сертификатов и CRL (certificate-verify-service)
- Сервис создания подписи (create-signature-service)
- Сервис проверки подписи (verify-signature-service)
- Мост интеграции kafka с сервисом подписания (create-signature-bridge)
- Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge)
- Backend-модуль сервиса управления (control-module-backend)

4.1. Сервис расчета хеша и случайных последовательностей байт (message-digest-service)

Конфигурация сервиса расположена в файле «config/application.yml»:

```

grpc:
  port: 6565
s3:
  endpoint: ${S3_ENDPOINT:https://play.min.io}
  accessKey: ${S3_ACCESS_KEY:Q3AM3UQ867SPQQA43P2F}
  secretKey: ${S3_SECRET_KEY:zuf+tfteSlswRu7BJ86wekitnifILbZam1KYY3TG}

provider:
  provider-type: ${PROVIDER_NAME:BC}
  default-cryptopro-provider-type: ${DEFAULT_CRYPTOPRO_PROVIDER_TYPE:GOST-2012-256}

service:
  digest:
    buffersize: ${DIGEST_BUFFER_SIZE:1048576}

server:
  port: 8080
  maxInboundMessageSize: ${MAX_INBOUND_MESSAGE_SIZE:52428800}
  servlet:
    application-display-name: message-digest-service

logging:
  config: ${LOGBACK_CONFIG_PATH:}
  level:
    service: ${LOGGING_LEVEL_ROOT:ERROR}

management:
  endpoints:
    web:
      exposure:
        include: "*"
  endpoint:
    metrics:
      enabled: true
    prometheus:
      enabled: true
  metrics:
    expiration: ${METRICS_EXPIRATION:600}
    bufferlength: ${METRICS_BUFFERLENGTH:10}
    export:

```

```

prometheus:
  enabled: true
tags:
  application: ${server.servlet.application-display-name}

```

Описание значимых параметров:

Параметр	Описание	Варианты значений
grpc.port	Порт, на котором сервис принимает gRPC запросы	
server.port	Порт, на котором сервис принимает HTTP запросы	
s3.endpoint	Адрес S3 хранилища	
s3.accessKey	Реквизиты доступа к хранилищу	
s3.secretKey	Реквизиты доступа к хранилищу	
provider.provider-type	Тип используемого провайдера	<ul style="list-style-type: none"> – «BC» - использование крипто провайдера BouncyCastle – «FSSP-CRYPTOPRO» - использование крипто провайдера КриптоПРО CSP
provider.default-cryptopro-provider-type	Провайдер по умолчанию для выполнения общих операций (получение ключей, случайной последовательности). Если не задан или задан «DEFAULT», то используется ГОСТ 2012 256	<ul style="list-style-type: none"> – «DEFAULT» – «GOST-2001» – «GOST-2012-256» – «GOST-2012-512»
service.digest.bufferSize	Размер буфера чтения для внутренних операций	
server.maxInboundMessageSize	Максимальный размер входящих сообщений	

4.2. Сервис проверки сертификатов и CRL (certificate-verify-service)

Конфигурация сервиса расположена в файле «config/application.yml»:

```

grpc:
  port: 6565
certverifier:
  cache:
    checkConnectionPeriodInSec: ${CHECK_CONNECTION_PERIOD:300}
    invalidateCertCache: ${INVALIDATE_CERTIFICATE_CACHE:invalidate_cert_cache}

```

```

invalidateCrlCache: ${INVALIDATE_CRL_CACHE:invalidate_crl_cache}
ttlInSec: ${CACHE_TTL:1800}
message:
  dateformat: yyyy-MM-dd HH:mm:ss.SSSXXX

provider:
  provider-type: ${PROVIDER_NAME:BC}
  default-cryptopro-provider-type: ${DEFAULT_CRYPTOPRO_PROVIDER_TYPE:GOST-2012-256}

spring:
  main:
    allow-bean-definition-overriding: true

cryptoserver:
  dao:
    configuration:
      host: ${DATABASE_HOST:fssp-dev-crypto.otr.ru}
      port: ${DATABASE_PORT:5432}
      database: ${DATABASE:crypto}
      schema: ${DATABASE_SCHEMA:crypto}
      user: ${DATABASE_USER:postgres}
      password: ${DATABASE_USER_PASSWORD:Postgres1}

logging:
  config: ${LOGBACK_CONFIG_PATH:}
  level:
    service: ${LOGGING_LEVEL_ROOT:ERROR}

management:
  endpoints:
    web:
      exposure:
        include: "*"
  endpoint:
    metrics:
      enabled: true
  prometheus:
    enabled: true
  metrics:
    expiration: ${METRICS_EXPIRATION:600}
    bufferlength: ${METRICS_BUFFERLENGTH:10}
    export:
      prometheus:
        enabled: true
  tags:
    application: ${server.servlet.application-display-name}

server:
  port: 8080
  servlet:
    application-display-name: certificate-verify-service

```

Описание значимых параметров:

Параметр	Описание	Варианты значений
grpc.port	Порт, на котором сервис принимает gRPC запросы	
server.port	Порт, на котором сервис принимает HTTP запросы	

certverifier.cache.checkConnectionPeriodInSec	Кеш. Таймаут проверки подключения к БД	
certverifier.cache.invalidateCertCache	Кеш. Наименование очереди событий изменений таблицы Сертификатов в БД	
certverifier.cache.invalidateCrlCache	Кеш. Наименование очереди событий изменений таблицы CRL в БД	
certverifier.cache.ttlInSec	Кеш. Время жизни записи в кеше.	
certverifier.message.dateFormat	Формат отображения даты в логировании	
provider.provider-type	Тип используемого провайдера	<ul style="list-style-type: none"> – «BC» - использование крипто провайдера Bouncy Castle – «FSSP-CRYPTOPRO» - использование крипто провайдера КриптоПРО CSP
provider.default-cryptopro-provider-type	Провайдер по умолчанию для выполнения общих операций (получение ключей, случайной последовательности). Если не задан или задан «DEFAULT», то используется ГОСТ 2012 256	<ul style="list-style-type: none"> – «DEFAULT» – «GOST-2001» – «GOST-2012-256» – «GOST-2012-512»
cryptoserver.dao.configuration.host	БД. Хост БД	
cryptoserver.dao.configuration.port	БД. Порт БД	
cryptoserver.dao.configuration.database	БД. Наименование БД	
cryptoserver.dao.configuration.schema	БД. Схема в БД	
cryptoserver.dao.configuration.user	БД. Пользователь БД	
cryptoserver.dao.configuration.password	БД. Пароль пользователя БД	

4.3. Сервис создания подписи (create-signature-service)

Конфигурация сервиса расположена в файле «config/application.yml»:

```

grpc:
  port: 6565
s3:
  endpoint: ${S3_ENDPOINT:https://play.min.io}
  accessKey: ${S3_ACCESS_KEY:Q3AM3UQ867SPQQA43P2F}
  secretKey: ${S3_SECRET_KEY:zuf+tfteSlswRu7BJ86wekitnifILbZam1KYY3TG}

provider:
  provider-type: ${PROVIDER_NAME:BC}
  default-cryptopro-provider-type: ${DEFAULT_CRYPTOPRO_PROVIDER_TYPE:GOST-2012-256}

store:
  type: ${PROVIDER_STORE:PKCS12}
  store-file: ${STORE_FILE:./test1.p12}
  store-password: ${STORE_PASSWORD:Qwerty123}
  cache-private-key-handle: ${CACHE_PRIVATE_KEY_HANDLE:false}
  cache:
    expireAfterWrite: ${SYSTEM_STORE_CACHE_EXPIRE_AFTER_WRITE:300}
    expireAfterAccess: ${SYSTEM_STORE_CACHE_EXPIRE_AFTER_ACCESS:300}

server:
  port: 8080
  maxInboundMessageSize: ${MAX_INBOUND_MESSAGE_SIZE:52428800}
  servlet:
    application-display-name: create-signature-service

certificate:
  verify:
    service:
      address: ${VERIFY_CERTIFICATE_ENDPOINT:fssp-dev-crypto:6560}
      cacheExpireAfterAccess: ${VERIFY_CERT_SERVICE_CACHE_EXPIRE_AFTER_ACCESS:30}
      cacheExpireAfterWrite: ${VERIFY_CERT_SERVICE_CACHE_EXPIRE_AFTER_WRITE:300}
      cacheKeyDateTemplate: ${VERIFY_CERT_SERVICE_CACHE_KEY_DATE_TEMPLATE:yyyy-MM-
dd-HH}

certificates:
  mapping:
    path: ${PATH_TO_CERTIFICATE_MAPPING;}

create:
  signature:
    service:
      buffersize: ${SIGNATURE_BUFFER_SIZE:1048576}

tsp:
  service:
    url: ${TSP_SERVICE_URL:http://fssp-dev-crypto:8090/api/v1/tsp/}
    type: ${TSP_SERVICE_TYPE:INTERNAL}
  property:
    certificate-fingerprint:
    ${TSP_CERTIFICATE_FINGER_PRINT:e73f2e5b5516063bd908eb9a17cfaf78f9ae672e}
    security-policyid: ${TSP_SECURITY_POLICY_ID:1.2.3.4.5.6.7}
    sequence-name: ${TSP_SEQUENCE_NAME:tsp_cert_sequence_1}
    idKpTimeStamping: ${TSP_ID_KP_TIMESTAMPING:1.3.6.1.5.5.7.3.8}
    keyEntryPassword: ${TSP_KEY_ENTRY_PASSWORD;}

cryptoserver:
  dao:
    configuration:
      host: ${DATABASE_HOST:fssp-dev-crypto.otr.ru}
      port: ${DATABASE_PORT:5432}

```

```

database: ${DATABASE:crypto}
schema: ${DATABASE_SCHEMA:crypto}
user: ${DATABASE_USER:postgres}
password: ${DATABASE_USER_PASSWORD:Postgres1}

integration:
context:
header: ${INTEGRATION_CONTEXT_HEADER:Request-Context}

management:
endpoints:
web:
exposure:
include: "*"
endpoint:
metrics:
enabled: true
prometheus:
enabled: true
metrics:
expiration: ${METRICS_EXPIRATION:600}
bufferlength: ${METRICS_BUFFERLENGTH:10}
export:
prometheus:
enabled: true
tags:
application: ${server.servlet.application-display-name}

logging:
config: ${LOGBACK_CONFIG_PATH:}
level:
service: ${LOGGING_LEVEL_ROOT:ERROR}

```

Перечень сертификатов для подписания расположен в файле «config/certificates.json»:

```

{
  "test_256_old": {
    "fingerprint": "423c97121f54af760f36313d3db3f1cfc0f48c4a",
    "password": "Qwerty123"
  },
  "test_512": {
    "fingerprint": "423c97121f54af760f36313d3db3f1cfc0f48c4a",
    "password": "Qwerty123"
  },
  "test_256": {
    "fingerprint": "4eceff5ed2789688d382f01190fa3477a9669041",
    "password": ""
  }
}

```

Описание файла «config/certificates.json»:

- Наименование (test_256_old, test_512 и т.д.) - название алиаса, по которому к сервису будут обращаться другие сервисы за подписанием
- fingerprint - отпечаток сертификата в сервисе
- password - пароль контейнера, если есть

Описание значимых параметров:

Параметр	Описание	Варианты значений
grpc.port	Порт, на котором сервис принимает gRPC	

	запросы	
server.port	Порт, на котором сервис принимает HTTP запросы	
s3.endpoint	Адрес S3 хранилища	
s3.accessKey	Реквизиты доступа к хранилищу	
s3.secretKey	Реквизиты доступа к хранилищу	
provider.provider-type	Тип используемого провайдера	<ul style="list-style-type: none"> – «BC» - использование крипто провайдера Bouncy Castle – «FSSP-CRYPTOPRO» - использование крипто провайдера КриптоПРО CSP
provider.default-cryptopro-provider-type	Провайдер по умолчанию для выполнения общих операций (получение ключей, случайной последовательности). Если не задан или задан «DEFAULT», то используется ГОСТ 2012 256	<ul style="list-style-type: none"> – «DEFAULT» – «GOST-2001» – «GOST-2012-256» – «GOST-2012-512»
store.type	Тип хранилища контейнеров с закрытыми ключами	<ul style="list-style-type: none"> – PKCS12 - в качестве хранилища используется файл в формате PKCS12 – SYSTEM - в качестве хранилища используется системное хранилище Операционной системы
store.store-file	Расположение файла с хранилищем, если выбран тип «PKCS12»	
store.store-password	Пароль от файла хранилища, если выбран тип «PKCS12»	
store.cache-private-key-handle	Кешировать ли указатель на закрытый ключ, если выбран тип	

	«PKCS12»	
store.cache.expireAfterWrite	Сброс кэша n секунд.	
store.cache.expireAfterAccess	Удаления записей, которые простаивали в течение n секунд.	
server.maxInboundMessageSize	Максимальный размер входящих сообщений	
certificate.verify.service.address	Адрес сервиса проверки сертификатов	
certificates.mapping.path	Расположение файла «config/certificates.json» с перечнем возможных сертификатов	
create.signature.service.bufferSize	Размер буфера чтения для внутренних операций	
create.signature.service.cache.removeEntryTime	Кеш результатов проверки сертификата. Удаления записей, которые простаивали в течение n секунд	
create.signature.service.cache.clear	Кеш результатов проверки сертификата. Сброс кэша n секунд.	
tsp.service.url	Адрес сервиса службы меток времени	Используется, если «tsp.service.type» – EXTERNAL
tsp.service.type	Тип формирования меток времени	<ul style="list-style-type: none"> • EXTERNAL – использование внешнего сервиса формирования меток времени • INTERNAL – используется встроенный в сервис механизм формирования меток времени
tsp.property.certificate-fingerprint	Отпечаток сертификата службы меток времени, которым необходимо	Используется, если «tsp.service.type» – INTERNAL

	подписывать ответы службы	
tsp.property.security-policyid	OID политики безопасности, который должен включаться в формируемые сервисом метки времени	Используется, если «tsp.service.type» – INTERNAL
tsp.property.sequence-file	Расположение файла «sequence.txt»	Используется, если «tsp.service.type» – INTERNAL
tsp.property.idKpTimeStamping	OID политики меток времени	Используется, если «tsp.service.type» – INTERNAL
tsp.property.keyEntryPassword	Пароль от хранилища/ключа	Используется, если «tsp.service.type» – INTERNAL
cryptoserver.dao.configuration.host	БД. Хост БД	
cryptoserver.dao.configuration.port	БД. Порт БД	
cryptoserver.dao.configuration.database	БД. Наименование БД	
cryptoserver.dao.configuration.schema	БД. Схема в БД	
cryptoserver.dao.configuration.user	БД. Пользователь БД	
cryptoserver.dao.configuration.password	БД. Пароль пользователя БД	
integration.context.header	Имя заголовка http-запроса, в котором будет передаваться информация о контексте	

4.4. Сервис проверки подписи (verify-signature-service)

Конфигурация сервиса расположена в файле «config/application.yml»:

```
s3:
  endpoint: ${S3_ENDPOINT:https://play.min.io}
  accessKey: ${S3_ACCESS_KEY:Q3AM3UQ867SPQQA43P2F}
  secretKey: ${S3_SECRET_KEY:zuf+tfteSlswRu7BJ86wekitnifI1bZam1KYY3TG}

provider:
  provider-type: ${PROVIDER_NAME:FSSP-CRYPTOPRO}
  default-cryptopro-provider-type: ${DEFAULT_CRYPTOPRO_PROVIDER_TYPE:GOST-2012-256}

store:
  type: ${PROVIDER_STORE:SYSTEM}
  store-file: ${STORE_FILE:./test1.p12}
  store-password: ${STORE_PASSWORD:}
  cache-private-key-handle: ${CACHE_PRIVATE_KEY_HANDLE:false}
  cache:
```

```

    expireAfterWrite: ${SYSTEM_STORE_CACHE_EXPIRE_AFTER_WRITE:300}
    expireAfterAccess: ${SYSTEM_STORE_CACHE_EXPIRE_AFTER_ACCESS:30}

server:
  port: 8080
  maxInboundMessageSize: ${MAX_INBOUND_MESSAGE_SIZE:52428800}
  servlet:
    application-display-name: verify-signature-service

grpc:
  port: 6565

certificate:
  verify:
    service:
      address: ${VERIFY_CERTIFICATE_ENDPOINT:fssp-dev-crypto.otr.ru:6560}
      cacheExpireAfterAccess: ${VERIFY_CERT_SERVICE_CACHE_EXPIRE_AFTER_ACCESS:30}
      cacheExpireAfterWrite: ${VERIFY_CERT_SERVICE_CACHE_EXPIRE_AFTER_WRITE:300}
      cacheKeyDateTemplate: ${VERIFY_CERT_SERVICE_CACHE_KEY_DATE_TEMPLATE:yyyy-MM-
dd-HH}

  verify:
    signature:
      service:
        buffersize: ${SIGNATURE_BUFFER_SIZE:1048576}

tsp:
  service:
    url: ${TSP_SERVICE_URL:http://fssp-dev-crypto:8090/api/v1/tsp/}
    type: ${TSP_SERVICE_TYPE:INTERNAL}
  property:
    certificate-fingerprint:
    ${TSP_CERTIFICATE_FINGER_PRINT:e73f2e5b5516063bd908eb9a17cfaf78f9ae672e}
    security-policyid: ${TSP_SECURITY_POLICY_ID:1.2.3.4.5.6.7}
    sequence-name: ${TSP_SEQUENCE_NAME:tsp_cert_sequence_1}
    idKpTimeStamping: ${TSP_ID_KP_TIMESTAMPING:1.3.6.1.5.5.7.3.8}
    keyEntryPassword: ${TSP_KEY_ENTRY_PASSWORD:}

cryptoserver:
  dao:
    configuration:
      host: ${DATABASE_HOST:fssp-dev-crypto.otr.ru}
      port: ${DATABASE_PORT:5432}
      database: ${DATABASE:crypto}
      schema: ${DATABASE_SCHEMA:crypto}
      user: ${DATABASE_USER:postgres}
      password: ${DATABASE_USER_PASSWORD:Postgres1}

integration:
  context:
    header: ${INTEGRATION_CONTEXT_HEADER:Request-Context}

management:
  endpoints:
    web:
      exposure:
        include: "*"
  endpoint:
    metrics:
      enabled: true
    prometheus:
      enabled: true
  metrics:
    expiration: ${METRICS_EXPIRATION:600}
    bufferlength: ${METRICS_BUFFERLENGTH:10}

```

```

export:
  prometheus:
    enabled: true
  tags:
    application: ${server.servlet.application-display-name}

spring:
  main:
    allow-bean-definition-overriding: true

logging:
  config: ${LOGBACK_CONFIG_PATH:./config/logback-spring.xml}
  level:
    service: ${LOGGING_LEVEL_ROOT:ERROR}
    
```

Описание значимых параметров:

Параметр	Описание	Варианты значений
grpc.port	Порт, на котором сервис принимает gRPC запросы	
server.port	Порт, на котором сервис принимает HTTP запросы	
s3.endpoint	Адрес S3 хранилища	
s3.accessKey	Реквизиты доступа к хранилищу	
s3.secretKey	Реквизиты доступа к хранилищу	
provider.provider-type	Тип используемого провайдера	<ul style="list-style-type: none"> – «BC» - использование крипто провайдера BouncyCastle – «FSSP-CRYPTOPRO» - использование крипто провайдера КриптоПРО CSP
provider.default-cryptopro-provider-type	Провайдер по умолчанию для выполнения общих операций (получение ключей, случайной последовательности). Если не задан или задан «DEFAULT», то используется ГОСТ 2012 256	<ul style="list-style-type: none"> – «DEFAULT» – «GOST-2001» – «GOST-2012-256» – «GOST-2012-512»
store.type	Тип хранилища контейнеров с закрытыми ключами	<ul style="list-style-type: none"> – PKCS12 - в качестве хранилища используется файл в формате PKCS12 – SYSTEM - в качестве хранилища используется системное хранилище Операционной системы
store.store-file	Расположение файла с	

	хранилищем, если выбран тип «PKCS12»	
store.store-password	Пароль от файла хранилища, если выбран тип «PKCS12»	
store.cache-private-key-handle	Кешировать ли указатель на закрытый ключ, если выбран тип «PKCS12»	
store.cache.expireAfterWrite	Сброс кэша n секунд.	
store.cache.expireAfterAccess	Удаления записей, которые простаивали в течение n секунд.	
server.maxInboundMessageSize	Максимальный размер входящих сообщений	
certificate.verify.service.address	Адрес сервиса проверки сертификатов	
tsp.service.url	Адрес сервиса службы меток времени	Используется, если «tsp.service.type» – EXTERNAL
tsp.service.type	Тип формирования меток времени	<ul style="list-style-type: none"> • EXTERNAL – использование внешнего сервиса формирования меток времени • INTERNAL – используется встроенный в сервис механизм формирования меток времени
tsp.property.certificate-fingerprint	Отпечаток сертификата службы меток времени, которым необходимо подписывать ответы службы	Используется, если «tsp.service.type» – INTERNAL
tsp.property.security-policyid	OID политики безопасности, который должен включаться в формируемые сервисом метки времени	Используется, если «tsp.service.type» – INTERNAL
tsp.property.sequence-file	Расположение файла «sequence.txt»	Используется, если «tsp.service.type» – INTERNAL
tsp.property.idKpTimeStamping	OID политики меток	Используется, если

	времени	«tsp.service.type» INTERNAL	–
tsp.property.keyEntryPassword	Пароль от хранилища/ключа	Используется, «tsp.service.type» INTERNAL	если –
cryptoserver.dao.configuration.host	БД. Хост БД		
cryptoserver.dao.configuration.port	БД. Порт БД		
cryptoserver.dao.configuration.database	БД. Наименование БД		
cryptoserver.dao.configuration.schema	БД. Схема в БД		
cryptoserver.dao.configuration.user	БД. Пользователь БД		
cryptoserver.dao.configuration.password	БД. Пароль пользователя БД		
integration.context.header	Имя заголовка http-запроса, в котором будет передаваться информация о контексте		
verify.signature.service.bufferSize	Размер буфера чтения для внутренних операций		

4.5. Мост интеграции kafka с сервисом подписания (create-signature-bridge)

Конфигурация сервиса расположена в файле «config/application.yml»:

```
kafka:
  bootstrapAddress: ${KAFKA_URL:localhost:9093}
  concurrency: ${CONCURRENCY:1}
  request:
    topic: ${CREATE_SIGNATURE_REQUESTS_TOPIC:create-signature-request-topic}
    group: ${CREATE_SIGNATURE_REQUEST_GROUP:create-signature-service}
  response:
    topic: ${CREATE_SIGNATURE_RESPONSES_TOPIC:create-signature-response-topic}

create:
  signature:
    service:
      address: ${CREATE_SIGNATURE_URL:fssp-dev-crypto:8101}

control:
  module:
    rules:
      address: ${CONTROL_MODULE_RULES_URL:fssp-dev-crypto:8121}
      cacheExpireAfterAccess: ${CONTROL_MODULE_RULES_CACHE_EXPIRE_AFTER_ACCESS:30}
      cacheExpireAfterWrite: ${CONTROL_MODULE_RULES_CACHE_EXPIRE_AFTER_WRITE:300}

server:
  port: 8080
  servlet:
    application-display-name: create-signature-bridge

logging:
```

```

config: ${LOGBACK_CONFIG_PATH:}
level:
  service: ${LOGGING_LEVEL_ROOT:INFO}

```

Описание значимых параметров:

Параметр	Описание	Варианты значений
server.port	Порт, на котором сервис принимает HTTP запросы	
kafka.bootstrapAddress	Адрес сервиса kafka	
kafka.concurrency	Количество потоков обработки сервиса	
kafka.request.topic	Топик запросов	
kafka.request.group	Группа запросов	
kafka.response.topic	Топик ответов	
create.signature.service.address	Адрес сервиса формирования подписи	
control.module.rules.address	Адрес Backend-модуля сервиса управления	
control.module.rules.cacheExpireAfterAccess	Кеш запросов к сервису правил. Время жизни записи после последнего обращения	
control.module.rules.cacheExpireAfterWrite	Кеш запросов к сервису правил. Время жизни записи после первого обращения	

4.6. Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge)

Конфигурация сервиса расположена в файле «config/application.yml»:

```

kafka:
  bootstrapAddress: ${KAFKA_URL:localhost:9093}
  concurrency: ${CONCURRENCY:1}
  request:
    topic: ${VERIFY_SIGNATURE_REQUESTS_TOPIC:verify-signature-request-topic}
    group: ${VERIFY_SIGNATURE_REQUEST_GROUP:verify-signature-service}
  response:
    topic: ${VERIFY_SIGNATURE_RESPONSES_TOPIC:verify-signature-response-topic}

verify:
  signature:
    service:
      address: ${VERIFY_SIGNATURE_URL:fssp-dev-crypto:8201}

```

```

control:
  module:
    rules:
      address: ${CONTROL_MODULE_RULES_URL:fssp-dev-crypto:8121}
      cacheExpireAfterAccess: ${CONTROL_MODULE_RULES_CACHE_EXPIRE_AFTER_ACCESS:30}
      cacheExpireAfterWrite: ${CONTROL_MODULE_RULES_CACHE_EXPIRE_AFTER_WRITE:300}

  server:
    port: 8080
    servlet:
      application-display-name: verify-signature-bridge

  logging:
    config: ${LOGBACK_CONFIG_PATH:}
    level:
      service: ${LOGGING_LEVEL_ROOT:INFO}

```

Описание значимых параметров:

Параметр	Описание	Варианты значений
server.port	Порт, на котором сервис принимает HTTP запросы	
kafka.bootstrapAddress	Адрес сервиса kafka	
kafka.concurrency	Количество потоков обработки сервиса	
kafka.request.topic	Топик запросов	
kafka.request.group	Группа запросов	
kafka.response.topic	Топик ответов	
verify.signature.service.address	Адрес сервиса формирования подписи	
control.module.rules.address	Адрес Backend-модуля сервиса управления	

4.7. Backend-модуль сервиса управления (control-module-backend)

Конфигурация сервиса расположена в файле «config/application.yml»:

```

grpc:
  port: 6565
server:
  port: 8080
  maxInboundMessageSize: ${MAX_INBOUND_MESSAGE_SIZE:52428800}
  servlet:
    application-display-name: control-module

rules:
  create:
    path: ${PATH_TO_CREATE_RULES:}
  verify:
    path: ${PATH_TO_VERIFY_RULES:}

```



```

"executionMessageBeforeSendingToSmev2": {
  "signatureType": "WSSECURITY",
  "certificate": "test_256",
  "createWsSecurityRule": {
    "actor": "http://smev.gosuslugi.ru/actors/smev"
  }
}
}

```

Конфигурация правил для проверки подписи расположена в файле «config/verifySignatureRules.yml»:

```

{
  "executionBeforeAcceptMessageFromSmev3": {
    "signatureType": "XML",
    "verifyXmlSignatureRule": {
      "signatureFormat": "XMLDSIG"
    }
  },
  "executionDocumentAttachCheck": {
    "signatureType": "BINARY",
    "verifyBinarySignatureRule": {
      "signatureFormat": "CADES_C"
    }
  },
  "executionDocumentApproveCheck": {
    "signatureType": "XML",
    "verifyXmlSignatureRule": {
      "signatureFormat": "XADES_C"
    }
  },
  "executionBeforeAcceptMessageFromSmev2": {
    "signatureType": "WSSECURITY",
    "verifyWsSecurityRule": {
      "actor": "http://smev.gosuslugi.ru/actors/recipient"
    }
  }
}

```

Описание значимых параметров «config/application.yml»:

Параметр	Описание	Варианты значений
grpc.port	Порт, на котором сервис принимает gRPC запросы	
server.port	Порт, на котором сервис принимает HTTP запросы	
server.maxInboundMessageSize	Максимальный размер входящих сообщений	
rules.create.path	Путь к файлу с конфигурацией правил на формирование подписи	
rules.verify.path	Путь к файлу с конфигурацией правил на проверку подписи	

Описание значимых параметров «config/createSignatureRules.yml»

Каждое правило имеет имя, например, `createAuthoritySignature`, `createDetachedSignature`, `createBinarySignature`, `createWsSecuritySignature`.

Формат описания правил следующий:

- **signatureType** - тип ЭП, одно из значений:
 - XML (Подпись семейства XMLDsig/XAdES)
 - BINARY (Подпись семейства CMS/CAAdES)
 - WSSECURITY (Подпись формата WS Security X509 Token Profile)
- **certificate** - алиас сертификата, который должен использоваться при формировании ЭП
- **createXmlSignatureRule** - параметры создания подписи типа XML:
 - **signatureFormat** - формат XML подписи, одно из значений XMLDSIG, XADES_BES, XADES_T, XADES_C. Определяет формат создаваемой ЭП;
 - **embeddedMode** - описание режима встраивания подписи в подписываемый документ:
 - **attached** - boolean флаг, должна ли быть подпись встроена в XML документ
 - **embeddingPath** - XPath выражение, которое используется если флаг `attached` установлен в true. Подпись при встраивании встраивается как дочерний элемент для элемента, заданного этим XPath выражением
 - **xmlParts** - используется при частичном подписании XML документа. Содержит массив, описывающий ID всех элементов, которые должны быть подписаны в подписываемом XML, и трансформеры, которые должны быть применены к этим элементам, в виде base64 строки
- **createBinarySignatureRule** - параметры создания бинарной подписи
 - **signatureFormat** - одно из значений CMS, CADES_BES, CADES_T, CADES_C. Определяет формат создаваемой ЭП;
- **createWsSecurityRule** - параметры создания WS-Security подписи:
 - **actor** - значение атрибута `actor`, которое должно быть установлено при формировании подписи
 - **xmlParts** - используется при частичном подписании XML документа. Содержит массив, описывающий ID всех элементов, которые должны быть подписаны в подписываемом XML, и трансформеры, которые должны быть применены к этим элементам, в виде base64 строки

Описание значимых параметров «`config/verifySignatureRules.yml`»

Каждое правило имеет имя, например, `verifyXmlDSig`, `verifyCMS`, `verifyCadesBes`, `verifyWsSecurity`.

Формат описания правил следующий:

- **signatureType** - тип ЭП, одно из значений:
 - XML (Подпись семейства XMLDsig/XAdES)
 - BINARY (Подпись семейства CMS/CAAdES)
 - WSSECURITY (Подпись формата WS Security X509 Token Profile)
- **verifyXmlSignatureRule** - параметры проверки подписи типа XML:
 - **signatureFormat** - формат XML подписи, одно из значений XMLDSIG, XADES_BES, XADES_T, XADES_C. Определяет формат, которому должна соответствовать проверяемая ЭП;
- **verifyBinarySignatureRule** - параметры проверки подписи типа BINARY:
 - **signatureFormat** - одно из значений CMS, CADES_BES, CADES_T, CADES_C. Определяет формат, которому должна соответствовать проверяемая ЭП;
- **verifyWsSecurityRule** - параметры проверки подписи типа WS-Security:
 - **actor** – значение атрибута `actor`. Поэтому атрибуту, среди всех WS-Security подписей, выбирается та, которая должна проверяться

5. НАСТРОЙКА СЕРВИСОВ

Необходимо создать каталог, в который скопировать дистрибутивы сервисов. Для этого выполнить команду:

```
mkdir -p /opt/crypto/stand/
```

Сервисы будут располагаться в данном каталоге.

5.1. Сервис расчета хеша и случайных последовательностей байт (message-digest-service)

Файл «./config/application.yml»:

- В параметре «provider-type» указать «FSSP-CRYPTOPRO»
- Группа «s3»:
 - endpoint = http://localhost:9100
 - accessKey = АКIAIOSFODNN7EXAMPLE
 - secretKey = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
- В параметре «logging.config» указать «./config/logback-spring.xml»
- В параметре «server.port» установить значение «8085»
- В параметре «grpc.port» установить значение «6565»

5.2. Сервис проверки сертификатов и CRL (certificate-verify-service)

Файл «./config/application.yml»:

- В параметре «provider-type» указать «FSSP-CRYPTOPRO»
- В разделе «cryptoserver.dao.configuration» указать параметры подключения к БД:
 - host = IP или имя хоста, на котором запущена БД (localhost можно использовать, если в настройках pg_hba.conf для локальных подключений указано md5)
 - port = 5432
 - database = crypto
 - schema = crypto
 - user = postgres
 - password = postgres
- В параметре «logging.config» указать «./config/logback-spring.xml»
- В параметре «server.port» установить значение «8081»
- В параметре «grpc.port» установить значение «6561»

5.3. Сервис создания подписи (create-signature-service)

Файл «./config/application.yml»:

- В параметре «provider-type» указать «FSSP-CRYPTOPRO»
- В параметре «store.type» указать «SYSTEM»
- Группа «s3»:
 - endpoint = http://localhost:9100
 - accessKey = АКIAIOSFODNN7EXAMPLE
 - secretKey = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
- Очистить значение параметре «store.store-password»

- В параметре «certificate.verify.service.address» указать значение «localhost:6561»
- В параметре «certificates.mapping.path» указать значение «./config/certificates.json»
- В параметре «tsp.service.type» указать значение «INTERNAL»
- В параметре «tsp.property.certificate-fingerprint» установить значение «e73f2e5b5516063bd908eb9a17cfaf78f9ae672e»
- В параметре «tsp.property.sequence-name» установить значение «tsp_cert_sequence_1»
- В параметре «logging.config» указать «./config/logback-spring.xml»
- В разделе «cryptoserver.dao.configuration» указать параметры подключения к БД:
 - host = IP или имя хоста, на котором запущена БД (localhost можно использовать, если в настройках pg_hba.conf для локальных подключений указано md5)
 - port = 5432
 - database = crypto
 - schema = crypto
 - user = postgres
 - password = postgres
- В параметре «server.port» установить значение «8083»
- В параметре «grpc.port» установить значение «6563»

В каталоге «./config» создать файл «certificates.json» (если он отсутствует) со значени-
ем:

```
{
  "test_256_old": {
    "fingerprint": "423c97121f54af760f36313d3db3f1cfc0f48c4a",
    "password": "Qwerty123"
  },
  "test_512": {
    "fingerprint": "423c97121f54af760f36313d3db3f1cfc0f48c4a",
    "password": "Qwerty123"
  },
  "test_256": {
    "fingerprint": "4eceff5ed2789688d382f01190fa3477a9669041",
    "password": ""
  }
}
```

5.4. Сервис проверки подписи (verify-signature-service)

Файл «./config/application.yml»:

- В параметре «provider-type» указать «FSSP-CRYPTOPRO»
- Группа «s3»:
 - endpoint = http://localhost:9100
 - accessKey = AKIAIOSFODNN7EXAMPLE
 - secretKey = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
- В параметре «certificate.verify.service.address» указать значение «localhost:6561»
- В параметре «tsp.service.type» указать значение «INTERNAL»
- В параметре «tsp.property.certificate-fingerprint» установить значение «e73f2e5b5516063bd908eb9a17cfaf78f9ae672e»
- В параметре «tsp.property.sequence-name» установить значение «tsp_cert_sequence_1»
- В разделе «cryptoserver.dao.configuration» указать параметры подключения к БД:
 - host = IP или имя хоста, на котором запущена БД (localhost можно использовать, если в настройках pg_hba.conf для локальных подключений указано md5)
 - port = 5432
 - database = crypto

- schema = crypto
- user = postgres
- password = postgres
- В параметре «logging.config» указать «./config/logback-spring.xml»
- В параметре «server.port» установить значение «8089»
- В параметре «grpc.port» установить значение «6569»

5.5. Мост интеграции kafka с сервисом подписания (create-signature-bridge)

Файл «./config/application.yml»:

- В параметре «create.signature.service.address» указать «localhost:6563»
- В параметре «control.module.rules.address» указать «localhost:6570»
- В параметре «kafka.bootstrapAddress» указать «localhost:9093»
- В параметре «kafka.request.topic» указать «CreateSignatureRequestTopic»
- В параметре «kafka.request.group» указать «create-signature-group»
- В параметре «kafka.response.topic» указать «CreateSignatureResponseTopic»
- В параметре «logging.config» указать «./config/logback-spring.xml»
- В параметре «server.port» установить значение «8082»

5.6. Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge)

Файл «./config/application.yml»:

- В параметре «verify.signature.service.address» установить значение «localhost:6569»
- В параметре «control.module.rules.address» указать «localhost:6570»
- В параметре «kafka.bootstrapAddress» указать «localhost:9093»
- В параметре «kafka.request.topic» указать «VerifySignatureRequestTopic»
- В параметре «kafka.request.group» указать «verify-signature-group»
- В параметре «kafka.response.topic» указать «VerifySignatureResponseTopic»
- В параметре «logging.config» указать «./config/logback-spring.xml»
- В параметре «server.port» установить значение «8088»

5.7. Backend-модуль сервиса управления (control-module-backend)

Файл «./config/application.yml»:

- Группа «rules»:
 - create.path = ./config/createSignatureRules.yml
 - verify.path = ./config/verifySignatureRules.yml
- В параметре «logging.config» указать «./config/logback-spring.xml»
- В параметр «server.port» установить значение «8090»
- В параметр «grpc.port» установить значение «6570»

В каталоге «./config» создать файл «createSignatureRules.yml» со значением:

```
{
  "executionBeforeSendingToSnev3": {
```



```

    },
    "executionBeforeAcceptMessageFromSmev2": {
        "signatureType": "WSSECURITY",
        "verifyWsSecurityRule": {
            "actor" : "http://smev.gosuslugi.ru/actors/recipient"
        }
    }
}
}

```

5.8. Настройка логирования в ELK-стек (в каждом сервисе)

В каждом сервисе необходимо отредактировать файл «config/logback-spring.xml». Необходимо привести его к следующему виду:

```

<?xml version="1.0" encoding="UTF-8"?>
<configuration scan="true" scanPeriod="2 seconds">
  <property name="KAFKA_SERVER" value="localhost:9093"/>
  <property name="defaultPattern"
    value="%date{dd.MM.yyyy HH:mm:ss.SSS} %magenta([%thread]) [%-5level]
[%logger{36}] [%X{invokerUser}] [%X{processInstanceId}]
[%X{systemOriginator}/%X{systemOriginatorVersion}] - %msg%n"/>
  <springProperty name="rootLogLevel" source="logging.level.service"/>
  <springProperty name="application" source="server.servlet.application-display-
name"/>

  <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">
    <encoder>
      <pattern>${defaultPattern}</pattern>
    </encoder>
  </appender>

  <appender name="FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>logs/${application}.log</file>
    <rollingPolicy
class="ch.qos.logback.core.rolling.SizeAndTimeBasedRollingPolicy">
      <!-- daily rollover -->
      <fileNamePattern>logs/${application}-%d{yyyy-MM-
dd}_%i.zip</fileNamePattern>
      <maxHistory>${log.maxHistory:-7}</maxHistory>
      <maxFileSize>100MB</maxFileSize>
      <totalSizeCap>700MB</totalSizeCap>
    </rollingPolicy>

    <encoder>
      <pattern>${defaultPattern}</pattern>
      <charset>UTF-8</charset>
    </encoder>
  </appender>

  <if condition='isDefined("KAFKA_SERVER") '>
    <then>
      <appender
class="com.github.danielwegener.logback.kafka.KafkaAppender"
name="Kafka"
      <encoder
class="net.logstash.logback.encoder.LoggingEventCompositeJsonEncoder">
        <providers>
          <version/>
          <!--<timestamp>-->
          <!--<fieldName>timestamp</fieldName>-->
          <!--<pattern>yyyy-MM-dd HH:mm:ss,SSS</pattern>-->
          <!--<timeZone>UTC</timeZone>-->
          <!--</timestamp>-->
          <logLevel>

```


– Изменено название топика: *<topic>rouz-logs-ov</topic>*

6. НАСТРОЙКА ВСПОМОГАТЕЛЬНЫХ СЕРВИСОВ

6.1. Настройка необходимых подсетей

Для запуска сервисов необходимо создать сеть при помощи команды:

```
docker network create \  
  --subnet=10.111.0.0/16 \  
  --gateway=10.111.0.1 \  
  kafka-elk
```

6.2. Настройка вспомогательных компонент

Необходимо выполнить следующую последовательность действий:

1. Создать каталог, в котором, будет располагаться конфигурация вспомогательных сервисов - minio, kafka, ELK (Elasticsearch, Logstash, Kibana):

```
mkdir -p /opt/crypto/utility/  
mkdir -p /opt/crypto/utility/data/kafka-data/  
mkdir -p /opt/crypto/utility/data/minio/  
mkdir -p /opt/crypto/utility/data/zookeeper-data/
```

2. В данном каталоге «/opt/crypto/utility/» разместить файлы из Приложения 1
3. В файле «/opt/crypto/utility/docker-compose.yml» указать имя текущего хоста в строке (вместо «fssp-ov-crypto-lab-rhel.pds.otr.ru»):

```
- KAFKA_CFG_ADVERTISED_LISTENERS=CLIENT://kafka:9092,EXTERNAL://fssp-ov-crypto-lab-  
rhel.pds.otr.ru:9093
```

7. ПОРЯДОК ЗАПУСКА

Необходимо запустить сервисы в следующем порядке:

1. Запустить вспомогательные сервисы (./start.sh в каталоге /opt/crypto/utility/)
2. Запустить Изделие в следующем порядке (./start.sh в соответствующем сервисе /opt/crypto/stand/*):
 - 2.1. Сервис расчета хеша и случайных последовательностей байт (message-digest-service)
 - 2.2. Сервис проверки сертификатов и CRL (certificate-verify-service)
 - 2.3. Сервис создания подписи (create-signature-service)
 - 2.4. Сервис проверки подписи (verify-signature-service)
 - 2.5. Backend-модуль сервиса управления (control-module-backend)
 - 2.6. Мост интеграции kafka с сервисом подписания (create-signature-bridge)
 - 2.7. Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge)

8. РАБОТА С КЛЮЧАМИ И СЕРТИФИКАТАМИ

Для работы Изделия должны использоваться ключи, СКП, выпущенные аккредитованным удостоверяющим центром (далее – УЦ).

ПРИЛОЖЕНИЕ 1. ФАЙЛЫ ЗАПУСКА ВСПОМОГАТЕЛЬНЫХ КОМПОНЕНТ

Конфигурационный файл docker-compose.yml:

```

version: '3.5'
services:
  minio-service:
    image: "minio/minio"
    restart: "unless-stopped"
    expose:
      - "9000"
    ports:
      - "9100:9000"
    volumes:
      - "${PWD}/data/minio:/data"
    environment:
      MINIO_ACCESS_KEY: "AKIAIOSFODNN7EXAMPLE"
      MINIO_SECRET_KEY: "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
    command: server /data
    networks:
      - kafka-elk

  zookeeper:
    container_name: zookeeper
    image: 'docker.io/bitnami/zookeeper:3-debian-10'
    restart: "unless-stopped"
    ports:
      - '2181:2181'
    volumes:
      - 'zookeeper_data:/bitnami'
    environment:
      - ALLOW_ANONYMOUS_LOGIN=yes
    networks:
      - kafka-elk

  kafka:
    container_name: kafka
    image: 'docker.io/bitnami/kafka:2-debian-10'
    restart: "unless-stopped"
    ports:
      - '9092:9092'
      - '9093:9093'
    volumes:
      - 'kafka_data:/bitnami'
    environment:
      - KAFKA_CFG_ZOOKEEPER_CONNECT=zookeeper:2181
      - ALLOW_PLAINTEXT_LISTENER=yes
      - KAFKA_CFG_LISTENER_SECURITY_PROTOCOL_MAP=CLIENT:PLAINTEXT,EXTERNAL:PLAINTEXT
      - KAFKA_CFG_LISTENERS=CLIENT://:9092,EXTERNAL://:9093
      - KAFKA_CFG_ADVERTISED_LISTENERS=CLIENT://kafka:9092,EXTERNAL://fssp-ov-
crypto-lab-rhel.pds.otr.ru:9093
      - KAFKA_INTER_BROKER_LISTENER_NAME=CLIENT
    depends_on:
      - zookeeper
    networks:
      - kafka-elk

#####
#           logstash           #
#                               #
#####
logstash:

```

```

container_name: logstash
image: docker.elastic.co/logstash/logstash:7.8.0
networks:
- kafka-elk
environment:
  XPACK_MONITORING_ENABLED: "false"
ports:
- "5045-5055:5045-5055"
restart: unless-stopped
volumes:
- ./config-elk/pipelines.yml:/usr/share/logstash/config/pipelines.yml:ro
- ./config-elk/pipelines:/usr/share/logstash/config/pipelines:ro
depends_on:
- "kafka"
#####
#                               #
#       elasticsearch           #
#                               #
#####
elasticsearch:
  image: docker.elastic.co/elasticsearch/elasticsearch:7.8.0
  container_name: elasticsearch
  environment:
    ES_JAVA_OPTS: -Xms512m -Xmx512m
    discovery.type: single-node
    xpack.license.self_generated.type: trial
  ulimits:
    memlock:
      soft: -1
      hard: -1
  volumes:
  - ./config-elk/data01:/usr/share/elasticsearch/data
  restart: unless-stopped
  ports:
  - 9200:9200
  networks:
  - kafka-elk
#####
#                               #
#       kibana                  #
#                               #
#####
kibana:
  image: docker.elastic.co/kibana/kibana:7.8.0
  container_name: kibana
  ports:
  - 5601:5601
  restart: unless-stopped
  networks:
  - kafka-elk
  links:
  - elasticsearch
  environment:
    ELASTICSEARCH_URL: http://elasticsearch:9200

volumes:
  zookeeper_data:
    driver: local
    driver_opts:
      type: none
      o: bind
      device: "${PWD}/data/zookeeper-data"
  kafka_data:
    driver: local
    driver_opts:

```

```
type: none
o: bind
device: "${PWD}/data/kafka-data"

networks:
  kafka-elk:
  external:
  name: kafka-elk
```

Файл запуска сервисов:

```
#!/bin/sh
#docker-compose pull &&
#docker rmi $(docker images --filter 'dangling=true' -q --no-trunc) &&
docker-compose up -d
```

Файл остановки сервисов:

```
#!/bin/sh
docker-compose down -v
```

Архив, содержащий структуру каталога utility:



utility.zip

ПРИЛОЖЕНИЕ 2. ПЕРЕЧЕНЬ ФАЙЛОВ ИЗДЕЛИЯ ДЛЯ КОНТРОЛЯ

Сервис расчета хеша и случайных последовательностей байт (message-digest-service)

Таблица 4. Перечень файлов Изделия подлежащих контролю целостности

№ п/п.	Имя файла
1.	./lib/FastInfoset-1.2.16.jar
2.	./lib/HdrHistogram-2.1.12.jar
3.	./lib/LatencyUtils-2.0.3.jar
4.	./lib/accessors-smart-1.2.jar
5.	./lib/android-json-0.0.20131108.vaadin1.jar
6.	./lib/animal-sniffer-annotations-1.17.jar
7.	./lib/annotations-13.0.jar
8.	./lib/annotations-2.13.8.jar
9.	./lib/annotations-4.1.1.4.jar
10.	./lib/apiguardian-api-1.1.0.jar
11.	./lib/arns-2.13.8.jar
12.	./lib/asm-5.0.4.jar
13.	./lib/aspectjweaver-1.9.5.jar
14.	./lib/assertj-core-3.16.1.jar
15.	./lib/auth-2.13.8.jar
16.	./lib/aws-core-2.13.8.jar
17.	./lib/aws-query-protocol-2.13.8.jar
18.	./lib/aws-xml-protocol-2.13.8.jar
19.	./lib/bcpkix-jdk15on-1.65.jar
20.	./lib/bcprov-jdk15on-1.65.jar
21.	./lib/byte-buddy-1.9.10.jar
22.	./lib/byte-buddy-agent-1.9.10.jar
23.	./lib/checker-qual-2.8.1.jar
24.	./lib/classmate-1.3.4.jar
25.	./lib/commons-codec-1.13.jar
26.	./lib/commons-collections-3.2.2.jar
27.	./lib/commons-compiler-3.0.7.jar
28.	./lib/commons-lang3-3.9.jar
29.	./lib/commons-net-3.6.jar
30.	./lib/cryptoserver-bouncycastle-extension-1.0-SNAPSHOT.jar

31.	./lib/cryptoserver-cryptopro-jna-support-1.0-SNAPSHOT.jar
32.	./lib/cryptoserver-cryptopro-provider-1.0-SNAPSHOT.jar
33.	./lib/cryptoserver-grpc-api-1.0-SNAPSHOT.jar
34.	./lib/cryptoserver-grpc-support-1.0-SNAPSHOT.jar
35.	./lib/cryptoserver-integrity-base-1.0-SNAPSHOT.jar
36.	./lib/cryptoserver-logging-commons-1.0-SNAPSHOT.jar
37.	./lib/cryptoserver-message-digest-services-1.0-SNAPSHOT.jar
38.	./lib/cryptoserver-metrics-commons-1.0-SNAPSHOT.jar
39.	./lib/cryptoserver-s3-services-1.0-SNAPSHOT.jar
40.	./lib/cryptoserver-utilities-1.0-SNAPSHOT.jar
41.	./lib/de.huxhorn.sulky.ulid-8.2.0.jar
42.	./lib/error_prone_annotations-2.3.4.jar
43.	./lib/eventstream-1.0.1.jar
44.	./lib/execution-model-6.0.1.jar
45.	./lib/failureaccess-1.0.1.jar
46.	./lib/grpc-api-1.28.0.jar
47.	./lib/grpc-context-1.28.0.jar
48.	./lib/grpc-core-1.28.0.jar
49.	./lib/grpc-netty-1.28.0.jar
50.	./lib/grpc-netty-shaded-1.28.0.jar
51.	./lib/grpc-protobuf-1.28.0.jar
52.	./lib/grpc-protobuf-lite-1.28.0.jar
53.	./lib/grpc-services-1.28.0.jar
54.	./lib/grpc-spring-boot-starter-3.5.3.jar
55.	./lib/grpc-stub-1.28.0.jar
56.	./lib/gson-2.8.6.jar
57.	./lib/guava-28.0-jre.jar
58.	./lib/hamcrest-2.2.jar
59.	./lib/hibernate-validator-6.1.5.Final.jar
60.	./lib/http-client-spi-2.13.8.jar
61.	./lib/istack-commons-runtime-3.0.8.jar
62.	./lib/j2objc-annotations-1.3.jar
63.	./lib/jackson-annotations-2.11.0.jar
64.	./lib/jackson-core-2.11.0.jar
65.	./lib/jackson-databind-2.11.0.jar
66.	./lib/jackson-datatype-jdk8-2.11.0.jar
67.	./lib/jackson-datatype-jsr310-2.11.0.jar
68.	./lib/jackson-module-parameter-names-2.11.0.jar

69.	./lib/jakarta.activation-api-1.2.2.jar
70.	./lib/jakarta.annotation-api-1.3.5.jar
71.	./lib/jakarta.validation-api-2.0.2.jar
72.	./lib/jakarta.xml.bind-api-2.3.3.jar
73.	./lib/janino-3.0.7.jar
74.	./lib/javax.activation-api-1.2.0.jar
75.	./lib/javax.annotation-api-1.3.2.jar
76.	./lib/jaxb-api-2.3.1.jar
77.	./lib/jaxb-runtime-2.3.2.jar
78.	./lib/jboss-logging-3.3.2.Final.jar
79.	./lib/jna-5.6.0.jar
80.	./lib/jna-platform-5.6.0.jar
81.	./lib/json-path-2.4.0.jar
82.	./lib/json-smart-2.3.jar
83.	./lib/jsonassert-1.5.0.jar
84.	./lib/jsr305-3.0.2.jar
85.	./lib/jul-to-slf4j-1.7.30.jar
86.	./lib/junit-jupiter-5.6.2.jar
87.	./lib/kafka-clients-1.0.0.jar
88.	./lib/kotlin-reflect-1.3.72.jar
89.	./lib/kotlin-stdlib-1.3.72.jar
90.	./lib/kotlin-stdlib-common-1.3.72.jar
91.	./lib/kotlin-stdlib-jdk7-1.3.72.jar
92.	./lib/kotlin-stdlib-jdk8-1.3.72.jar
93.	./lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
94.	./lib/log4j-api-2.13.2.jar
95.	./lib/log4j-to-slf4j-2.13.2.jar
96.	./lib/logback-classic-1.2.3.jar
97.	./lib/logback-core-1.2.3.jar
98.	./lib/logback-kafka-appender-0.2.0-RC2.jar
99.	./lib/logstash-logback-encoder-5.3.jar
100.	./lib/lombok-1.18.12.jar
101.	./lib/lz4-java-1.4.jar
102.	./lib/micrometer-core-1.5.4.jar
103.	./lib/micrometer-registry-prometheus-1.5.4.jar
104.	./lib/mockito-junit-jupiter-3.3.3.jar
105.	./lib/netty-buffer-4.1.45.Final.jar
106.	./lib/netty-codec-4.1.45.Final.jar

107.	./lib/netty-codec-http-4.1.45.Final.jar
108.	./lib/netty-codec-http2-4.1.45.Final.jar
109.	./lib/netty-codec-socks-4.1.45.Final.jar
110.	./lib/netty-common-4.1.45.Final.jar
111.	./lib/netty-handler-4.1.45.Final.jar
112.	./lib/netty-handler-proxy-4.1.45.Final.jar
113.	./lib/netty-resolver-4.1.45.Final.jar
114.	./lib/netty-tcnative-boringssl-static-2.0.25.Final.jar
115.	./lib/netty-transport-4.1.45.Final.jar
116.	./lib/netty-transport-native-epoll-4.1.49.Final-linux-x86_64.jar
117.	./lib/netty-transport-native-unix-common-4.1.49.Final.jar
118.	./lib/nio-multipart-parser-1.1.0.jar
119.	./lib/nio-stream-storage-1.1.3.jar
120.	./lib/objenesis-2.6.jar
121.	./lib/opentest4j-1.2.0.jar
122.	./lib/perfmark-api-0.19.0.jar
123.	./lib/profiles-2.13.8.jar
124.	./lib/proto-google-common-protos-1.17.0.jar
125.	./lib/protobuf-java-3.11.3.jar
126.	./lib/protobuf-java-util-3.11.3.jar
127.	./lib/protocol-core-2.13.8.jar
128.	./lib/reactive-streams-1.0.3.jar
129.	./lib/reactor-core-3.3.5.RELEASE.jar
130.	./lib/reactor-kotlin-extensions-1.0.0.RELEASE.jar
131.	./lib/reactor-netty-0.9.7.RELEASE.jar
132.	./lib/regions-2.13.8.jar
133.	./lib/s3-2.13.8.jar
134.	./lib/sdk-core-2.13.8.jar
135.	./lib/simpleclient-0.8.1.jar
136.	./lib/simpleclient_common-0.8.1.jar
137.	./lib/slf4j-api-1.7.25.jar
138.	./lib/snakeyaml-1.26.jar
139.	./lib/snappy-java-1.1.4.jar
140.	./lib/spring-aop-5.2.6.RELEASE.jar
141.	./lib/spring-beans-5.2.6.RELEASE.jar
142.	./lib/spring-boot-2.3.0.RELEASE.jar
143.	./lib/spring-boot-actuator-2.3.0.RELEASE.jar
144.	./lib/spring-boot-actuator-autoconfigure-2.3.0.RELEASE.jar

145.	./lib/spring-boot-autoconfigure-2.3.0.RELEASE.jar
146.	./lib/spring-boot-configuration-processor-2.3.0.RELEASE.jar
147.	./lib/spring-boot-dependencies-2.3.0.RELEASE.pom
148.	./lib/spring-boot-starter-2.3.0.RELEASE.jar
149.	./lib/spring-boot-starter-actuator-2.3.0.RELEASE.jar
150.	./lib/spring-boot-starter-aop-2.3.0.RELEASE.jar
151.	./lib/spring-boot-starter-json-2.3.0.RELEASE.jar
152.	./lib/spring-boot-starter-logging-2.3.0.RELEASE.jar
153.	./lib/spring-boot-starter-reactor-netty-2.3.0.RELEASE.jar
154.	./lib/spring-boot-starter-web-2.3.0.RELEASE.jar
155.	./lib/spring-boot-starter-webflux-2.3.0.RELEASE.jar
156.	./lib/spring-boot-test-2.3.0.RELEASE.jar
157.	./lib/spring-context-5.2.6.RELEASE.jar
158.	./lib/spring-core-5.2.6.RELEASE.jar
159.	./lib/spring-expression-5.2.6.RELEASE.jar
160.	./lib/spring-jcl-5.2.6.RELEASE.jar
161.	./lib/spring-web-5.2.6.RELEASE.jar
162.	./lib/spring-webflux-5.2.6.RELEASE.jar
163.	./lib/spring-webmvc-5.2.6.RELEASE.jar
164.	./lib/stax-ex-1.8.1.jar
165.	./lib/stax2-api-3.1.4.jar
166.	./lib/txw2-2.3.2.jar
167.	./lib/ulid-6.0.1.jar
168.	./lib/url-connection-client-2.13.8.jar
169.	./lib/utils-2.13.8.jar
170.	./lib/woodstox-core-5.0.3.jar
171.	./lib/xmlsec-2.1.5.jar
172.	./lib/xmlunit-core-2.7.0.jar
173.	./message-digest-service.jar

Сервис проверки сертификатов и CRL (certificate-verify-service)

Таблица 5. Перечень файлов Изделия, подлежащих контролю целостности

№ п/п.	Имя файла
1.	./certificate-verify-service.jar
2.	./lib/FastInfoset-1.2.15.jar
3.	./lib/HdrHistogram-2.1.12.jar

4.	./lib/HikariCP-3.3.1.jar
5.	./lib/LatencyUtils-2.0.3.jar
6.	./lib/accessors-smart-1.2.jar
7.	./lib/android-json-0.0.20131108.vaadin1.jar
8.	./lib/animal-sniffer-annotations-1.17.jar
9.	./lib/annotations-13.0.jar
10.	./lib/annotations-4.1.1.4.jar
11.	./lib/antlr-2.7.7.jar
12.	./lib/apiguardian-api-1.1.0.jar
13.	./lib/asm-7.2.jar
14.	./lib/asm-analysis-7.2.jar
15.	./lib/asm-commons-7.2.jar
16.	./lib/asm-tree-7.2.jar
17.	./lib/aspectjweaver-1.9.5.jar
18.	./lib/assertj-core-3.16.1.jar
19.	./lib/bcpkix-jdk15on-1.65.jar
20.	./lib/bcprov-jdk15on-1.65.jar
21.	./lib/byte-buddy-1.10.10.jar
22.	./lib/byte-buddy-agent-1.9.10.jar
23.	./lib/checker-qual-2.8.1.jar
24.	./lib/classmate-1.5.1.jar
25.	./lib/client-2.1.jar
26.	./lib/common-2.1.jar
27.	./lib/commons-codec-1.13.jar
28.	./lib/commons-collections-3.2.2.jar
29.	./lib/commons-compiler-3.0.7.jar
30.	./lib/commons-io-2.6.jar
31.	./lib/commons-lang3-3.9.jar
32.	./lib/commons-net-3.6.jar
33.	./lib/cryptoserver-cryptopro-jna-support-1.0-SNAPSHOT.jar
34.	./lib/cryptoserver-cryptopro-provider-1.0-SNAPSHOT.jar
35.	./lib/cryptoserver-dao-1.0-SNAPSHOT.jar
36.	./lib/cryptoserver-grpc-api-1.0-SNAPSHOT.jar
37.	./lib/cryptoserver-integrity-base-1.0-SNAPSHOT.jar
38.	./lib/cryptoserver-logging-commons-1.0-SNAPSHOT.jar
39.	./lib/cryptoserver-metrics-commons-1.0-SNAPSHOT.jar
40.	./lib/cryptoserver-model-1.0-SNAPSHOT.jar
41.	./lib/cryptoserver-utilities-1.0-SNAPSHOT.jar

42.	./lib/dom4j-2.1.3.jar
43.	./lib/error_prone_annotations-2.3.2.jar
44.	./lib/failureaccess-1.0.1.jar
45.	./lib/grpc-api-1.28.0.jar
46.	./lib/grpc-context-1.28.0.jar
47.	./lib/grpc-core-1.28.0.jar
48.	./lib/grpc-netty-1.28.0.jar
49.	./lib/grpc-protobuf-1.28.0.jar
50.	./lib/grpc-protobuf-lite-1.28.0.jar
51.	./lib/grpc-services-1.28.0.jar
52.	./lib/grpc-spring-boot-starter-3.5.3.jar
53.	./lib/grpc-stub-1.28.0.jar
54.	./lib/gson-2.8.6.jar
55.	./lib/guava-28.0-jre.jar
56.	./lib/h2-1.4.200.jar
57.	./lib/hamcrest-2.2.jar
58.	./lib/hibernate-commons-annotations-5.1.0.Final.jar
59.	./lib/hibernate-core-5.4.16.Final.jar
60.	./lib/hibernate-types-52-2.1.1.jar
61.	./lib/istack-commons-runtime-3.0.7.jar
62.	./lib/j2objc-annotations-1.3.jar
63.	./lib/jackson-annotations-2.11.0.jar
64.	./lib/jackson-core-2.11.0.jar
65.	./lib/jackson-databind-2.11.0.jar
66.	./lib/jackson-datatype-jdk8-2.11.0.jar
67.	./lib/jackson-datatype-jsr310-2.11.0.jar
68.	./lib/jackson-module-parameter-names-2.11.0.jar
69.	./lib/jakarta.activation-api-1.2.2.jar
70.	./lib/jakarta.annotation-api-1.3.5.jar
71.	./lib/jakarta.el-3.0.3.jar
72.	./lib/jakarta.persistence-api-2.2.3.jar
73.	./lib/jakarta.servlet-api-4.0.3.jar
74.	./lib/jakarta.transaction-api-1.3.3.jar
75.	./lib/jakarta.websocket-api-1.1.2.jar
76.	./lib/jakarta.xml.bind-api-2.3.3.jar
77.	./lib/jandex-2.1.3.Final.jar
78.	./lib/janino-3.0.7.jar
79.	./lib/javassist-3.24.0-GA.jar

80.	./lib/javax-websocket-client-impl-9.4.28.v20200408.jar
81.	./lib/javax-websocket-server-impl-9.4.28.v20200408.jar
82.	./lib/javax.activation-api-1.2.0.jar
83.	./lib/javax.annotation-api-1.3.2.jar
84.	./lib/javax.persistence-api-2.2.jar
85.	./lib/jaxb-api-2.3.1.jar
86.	./lib/jaxb-runtime-2.3.1.jar
87.	./lib/jboss-logging-3.3.2.Final.jar
88.	./lib/jboss-transaction-api_1.2_spec-1.1.1.Final.jar
89.	./lib/jetty-annotations-9.4.28.v20200408.jar
90.	./lib/jetty-client-9.4.28.v20200408.jar
91.	./lib/jetty-continuation-9.4.28.v20200408.jar
92.	./lib/jetty-http-9.4.28.v20200408.jar
93.	./lib/jetty-io-9.4.28.v20200408.jar
94.	./lib/jetty-jndi-9.4.28.v20200408.jar
95.	./lib/jetty-plus-9.4.28.v20200408.jar
96.	./lib/jetty-security-9.4.28.v20200408.jar
97.	./lib/jetty-server-9.4.28.v20200408.jar
98.	./lib/jetty-servlet-9.4.28.v20200408.jar
99.	./lib/jetty-servlets-9.4.28.v20200408.jar
100.	./lib/jetty-util-9.4.28.v20200408.jar
101.	./lib/jetty-webapp-9.4.28.v20200408.jar
102.	./lib/jetty-xml-9.4.28.v20200408.jar
103.	./lib/jna-5.6.0.jar
104.	./lib/jna-platform-5.6.0.jar
105.	./lib/json-path-2.4.0.jar
106.	./lib/json-smart-2.3.jar
107.	./lib/jsonassert-1.5.0.jar
108.	./lib/jsr305-3.0.2.jar
109.	./lib/jul-to-slf4j-1.7.30.jar
110.	./lib/junit-jupiter-5.6.2.jar
111.	./lib/kafka-clients-1.0.0.jar
112.	./lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
113.	./lib/log4j-api-2.13.2.jar
114.	./lib/log4j-to-slf4j-2.13.2.jar
115.	./lib/logback-classic-1.2.3.jar
116.	./lib/logback-core-1.2.3.jar
117.	./lib/logback-kafka-appender-0.2.0-RC2.jar

118.	./lib/logstash-logback-encoder-5.3.jar
119.	./lib/lombok-1.18.12.jar
120.	./lib/lz4-java-1.4.jar
121.	./lib/micrometer-core-1.5.4.jar
122.	./lib/micrometer-registry-prometheus-1.5.4.jar
123.	./lib/mockito-junit-jupiter-3.3.3.jar
124.	./lib/netty-buffer-4.1.45.Final.jar
125.	./lib/netty-codec-4.1.45.Final.jar
126.	./lib/netty-codec-http-4.1.45.Final.jar
127.	./lib/netty-codec-http2-4.1.45.Final.jar
128.	./lib/netty-codec-socks-4.1.45.Final.jar
129.	./lib/netty-common-4.1.45.Final.jar
130.	./lib/netty-handler-4.1.45.Final.jar
131.	./lib/netty-handler-proxy-4.1.45.Final.jar
132.	./lib/netty-resolver-4.1.45.Final.jar
133.	./lib/netty-tcnative-boringssl-static-2.0.25.Final.jar
134.	./lib/netty-transport-4.1.45.Final.jar
135.	./lib/netty-transport-native-epoll-4.1.48.Final-linux-x86_64.jar
136.	./lib/netty-transport-native-unix-common-4.1.48.Final.jar
137.	./lib/nio-multipart-parser-1.1.0.jar
138.	./lib/nio-stream-storage-1.1.3.jar
139.	./lib/objenesis-2.6.jar
140.	./lib/opentest4j-1.2.0.jar
141.	./lib/perfmark-api-0.19.0.jar
142.	./lib/postgresql-42.2.5.jar
143.	./lib/proto-google-common-protos-1.17.0.jar
144.	./lib/protobuf-java-3.11.3.jar
145.	./lib/protobuf-java-util-3.11.3.jar
146.	./lib/r2dbc-h2-0.8.3.RELEASE.jar
147.	./lib/r2dbc-pool-0.8.3.RELEASE.jar
148.	./lib/r2dbc-postgresql-0.8.2.RELEASE.jar
149.	./lib/r2dbc-spi-0.8.0.RELEASE.jar
150.	./lib/reactive-streams-1.0.3.jar
151.	./lib/reactor-core-3.3.4.RELEASE.jar
152.	./lib/reactor-netty-0.9.6.RELEASE.jar
153.	./lib/reactor-pool-0.1.3.RELEASE.jar
154.	./lib/saslprep-1.1.jar
155.	./lib/simpleclient-0.8.1.jar

156.	./lib/simpleclient_common-0.8.1.jar
157.	./lib/slf4j-api-1.7.25.jar
158.	./lib/snakeyaml-1.26.jar
159.	./lib/snappy-java-1.1.4.jar
160.	./lib/spring-aop-5.2.6.RELEASE.jar
161.	./lib/spring-aspects-5.2.6.RELEASE.jar
162.	./lib/spring-beans-5.2.6.RELEASE.jar
163.	./lib/spring-boot-2.3.0.RELEASE.jar
164.	./lib/spring-boot-actuator-2.3.0.RELEASE.jar
165.	./lib/spring-boot-actuator-autoconfigure-2.3.0.RELEASE.jar
166.	./lib/spring-boot-autoconfigure-2.3.0.RELEASE.jar
167.	./lib/spring-boot-configuration-processor-2.3.0.RELEASE.jar
168.	./lib/spring-boot-starter-2.3.0.RELEASE.jar
169.	./lib/spring-boot-starter-actuator-2.3.0.RELEASE.jar
170.	./lib/spring-boot-starter-aop-2.3.0.RELEASE.jar
171.	./lib/spring-boot-starter-data-jpa-2.3.0.RELEASE.jar
172.	./lib/spring-boot-starter-jdbc-2.3.0.RELEASE.jar
173.	./lib/spring-boot-starter-jetty-2.3.0.RELEASE.jar
174.	./lib/spring-boot-starter-json-2.3.0.RELEASE.jar
175.	./lib/spring-boot-starter-logging-2.3.0.RELEASE.jar
176.	./lib/spring-boot-starter-reactor-netty-2.3.0.RELEASE.jar
177.	./lib/spring-boot-starter-security-2.3.0.RELEASE.jar
178.	./lib/spring-boot-starter-web-2.3.0.RELEASE.jar
179.	./lib/spring-boot-starter-webflux-2.3.0.RELEASE.jar
180.	./lib/spring-boot-test-2.3.0.RELEASE.jar
181.	./lib/spring-context-5.2.6.RELEASE.jar
182.	./lib/spring-core-5.2.6.RELEASE.jar
183.	./lib/spring-data-commons-2.3.0.RELEASE.jar
184.	./lib/spring-data-jpa-2.3.0.RELEASE.jar
185.	./lib/spring-data-r2dbc-1.1.0.RELEASE.jar
186.	./lib/spring-data-relational-2.0.0.RELEASE.jar
187.	./lib/spring-expression-5.2.6.RELEASE.jar
188.	./lib/spring-jcl-5.2.6.RELEASE.jar
189.	./lib/spring-jdbc-5.2.6.RELEASE.jar
190.	./lib/spring-orm-5.2.6.RELEASE.jar
191.	./lib/spring-security-config-5.3.2.RELEASE.jar
192.	./lib/spring-security-core-5.3.2.RELEASE.jar
193.	./lib/spring-security-web-5.3.2.RELEASE.jar

194.	./lib/spring-tx-5.2.6.RELEASE.jar
195.	./lib/spring-web-5.2.6.RELEASE.jar
196.	./lib/spring-webflux-5.2.6.RELEASE.jar
197.	./lib/spring-webmvc-5.2.6.RELEASE.jar
198.	./lib/stax-ex-1.8.jar
199.	./lib/stax2-api-3.1.4.jar
200.	./lib/stringprep-1.1.jar
201.	./lib/txw2-2.3.1.jar
202.	./lib/websocket-api-9.4.28.v20200408.jar
203.	./lib/websocket-client-9.4.28.v20200408.jar
204.	./lib/websocket-common-9.4.28.v20200408.jar
205.	./lib/websocket-server-9.4.28.v20200408.jar
206.	./lib/websocket-servlet-9.4.28.v20200408.jar
207.	./lib/woodstox-core-5.0.3.jar
208.	./lib/xmlsec-2.1.5.jar
209.	./lib/xmlunit-core-2.7.0.jar

Сервис создания подписи (create-signature-service)

Таблица 6. Перечень файлов Изделия, подлежащих контролю целостности

№ п/п.	Имя файла
1.	./create-signature-service.jar
2.	./lib/FastInfoset-1.2.16.jar
3.	./lib/HdrHistogram-2.1.12.jar
4.	./lib/LatencyUtils-2.0.3.jar
5.	./lib/accessors-smart-1.2.jar
6.	./lib/android-json-0.0.20131108.vaadin1.jar
7.	./lib/animal-sniffer-annotations-1.17.jar
8.	./lib/annotations-13.0.jar
9.	./lib/annotations-2.13.8.jar
10.	./lib/annotations-4.1.1.4.jar
11.	./lib/antlr-2.7.7.jar
12.	./lib/apiguardian-api-1.1.0.jar
13.	./lib/arns-2.13.8.jar
14.	./lib/asm-5.0.4.jar
15.	./lib/aspectjweaver-1.9.5.jar
16.	./lib/assertj-core-3.16.1.jar

17.	./lib/auth-2.13.8.jar
18.	./lib/aws-core-2.13.8.jar
19.	./lib/aws-query-protocol-2.13.8.jar
20.	./lib/aws-xml-protocol-2.13.8.jar
21.	./lib/bcpkix-jdk15on-1.65.jar
22.	./lib/bcprov-jdk15on-1.65.jar
23.	./lib/byte-buddy-1.9.10.jar
24.	./lib/byte-buddy-agent-1.9.10.jar
25.	./lib/checker-qual-2.8.1.jar
26.	./lib/classmate-1.3.4.jar
27.	./lib/client-2.1.jar
28.	./lib/common-2.1.jar
29.	./lib/commons-codec-1.13.jar
30.	./lib/commons-collections-3.2.2.jar
31.	./lib/commons-compiler-3.0.7.jar
32.	./lib/commons-lang3-3.9.jar
33.	./lib/commons-net-3.6.jar
34.	./lib/cryptoserver-bouncycastle-extension-1.0-SNAPSHOT.jar
35.	./lib/cryptoserver-create-signature-services-1.0-SNAPSHOT.jar
36.	./lib/cryptoserver-cryptopro-jna-support-1.0-SNAPSHOT.jar
37.	./lib/cryptoserver-cryptopro-provider-1.0-SNAPSHOT.jar
38.	./lib/cryptoserver-dao-1.0-SNAPSHOT.jar
39.	./lib/cryptoserver-extend-signature-services-1.0-SNAPSHOT.jar
40.	./lib/cryptoserver-grpc-api-1.0-SNAPSHOT.jar
41.	./lib/cryptoserver-grpc-client-certificate-verify-service-1.0-SNAPSHOT.jar
42.	./lib/cryptoserver-grpc-support-1.0-SNAPSHOT.jar
43.	./lib/cryptoserver-integrity-base-1.0-SNAPSHOT.jar
44.	./lib/cryptoserver-logging-commons-1.0-SNAPSHOT.jar
45.	./lib/cryptoserver-metrics-commons-1.0-SNAPSHOT.jar
46.	./lib/cryptoserver-model-1.0-SNAPSHOT.jar
47.	./lib/cryptoserver-s3-services-1.0-SNAPSHOT.jar
48.	./lib/cryptoserver-santuario-extension-1.0-SNAPSHOT.jar
49.	./lib/cryptoserver-system-store-1.0-SNAPSHOT.jar
50.	./lib/cryptoserver-tsp-client-services-1.0-SNAPSHOT.jar
51.	./lib/cryptoserver-tsp-server-services-1.0-SNAPSHOT.jar
52.	./lib/cryptoserver-utilities-1.0-SNAPSHOT.jar
53.	./lib/dom4j-2.1.3.jar
54.	./lib/error_prone_annotations-2.3.4.jar

55.	./lib/eventstream-1.0.1.jar
56.	./lib/failureaccess-1.0.1.jar
57.	./lib/grpc-api-1.28.0.jar
58.	./lib/grpc-context-1.28.0.jar
59.	./lib/grpc-core-1.28.0.jar
60.	./lib/grpc-netty-1.28.0.jar
61.	./lib/grpc-netty-shaded-1.28.0.jar
62.	./lib/grpc-protobuf-1.28.0.jar
63.	./lib/grpc-protobuf-lite-1.28.0.jar
64.	./lib/grpc-services-1.28.0.jar
65.	./lib/grpc-spring-boot-starter-3.5.3.jar
66.	./lib/grpc-stub-1.28.0.jar
67.	./lib/gson-2.8.6.jar
68.	./lib/guava-28.0-jre.jar
69.	./lib/h2-1.4.200.jar
70.	./lib/hamcrest-2.2.jar
71.	./lib/hibernate-commons-annotations-5.1.0.Final.jar
72.	./lib/hibernate-core-5.4.16.Final.jar
73.	./lib/hibernate-types-52-2.1.1.jar
74.	./lib/hibernate-validator-6.1.5.Final.jar
75.	./lib/http-client-spi-2.13.8.jar
76.	./lib/istack-commons-runtime-3.0.8.jar
77.	./lib/j2objc-annotations-1.3.jar
78.	./lib/jackson-annotations-2.11.0.jar
79.	./lib/jackson-core-2.11.0.jar
80.	./lib/jackson-databind-2.11.0.jar
81.	./lib/jackson-datatype-jdk8-2.11.0.jar
82.	./lib/jackson-datatype-jsr310-2.11.0.jar
83.	./lib/jackson-module-parameter-names-2.11.0.jar
84.	./lib/jakarta.activation-api-1.2.2.jar
85.	./lib/jakarta.annotation-api-1.3.5.jar
86.	./lib/jakarta.validation-api-2.0.2.jar
87.	./lib/jakarta.xml.bind-api-2.3.3.jar
88.	./lib/jandex-2.1.3.Final.jar
89.	./lib/janino-3.0.7.jar
90.	./lib/javassist-3.24.0-GA.jar
91.	./lib/javax.activation-api-1.2.0.jar
92.	./lib/javax.annotation-api-1.3.2.jar

93.	./lib/javax.persistence-api-2.2.jar
94.	./lib/jaxb-api-2.3.1.jar
95.	./lib/jaxb-runtime-2.3.2.jar
96.	./lib/jboss-logging-3.3.2.Final.jar
97.	./lib/jboss-transaction-api_1.2_spec-1.1.1.Final.jar
98.	./lib/jna-5.6.0.jar
99.	./lib/jna-platform-5.6.0.jar
100.	./lib/json-path-2.4.0.jar
101.	./lib/json-smart-2.3.jar
102.	./lib/jsonassert-1.5.0.jar
103.	./lib/jsr305-3.0.2.jar
104.	./lib/jul-to-slf4j-1.7.30.jar
105.	./lib/junit-jupiter-5.6.2.jar
106.	./lib/kafka-clients-1.0.0.jar
107.	./lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
108.	./lib/log4j-api-2.13.2.jar
109.	./lib/log4j-to-slf4j-2.13.2.jar
110.	./lib/logback-classic-1.2.3.jar
111.	./lib/logback-core-1.2.3.jar
112.	./lib/logback-kafka-appender-0.2.0-RC2.jar
113.	./lib/logstash-logback-encoder-5.3.jar
114.	./lib/lombok-1.18.12.jar
115.	./lib/lz4-java-1.4.jar
116.	./lib/micrometer-core-1.5.4.jar
117.	./lib/micrometer-registry-prometheus-1.5.4.jar
118.	./lib/mockito-junit-jupiter-3.3.3.jar
119.	./lib/netty-buffer-4.1.45.Final.jar
120.	./lib/netty-codec-4.1.45.Final.jar
121.	./lib/netty-codec-http-4.1.45.Final.jar
122.	./lib/netty-codec-http2-4.1.45.Final.jar
123.	./lib/netty-codec-socks-4.1.45.Final.jar
124.	./lib/netty-common-4.1.45.Final.jar
125.	./lib/netty-handler-4.1.45.Final.jar
126.	./lib/netty-handler-proxy-4.1.45.Final.jar
127.	./lib/netty-resolver-4.1.45.Final.jar
128.	./lib/netty-tcnative-boringssl-static-2.0.25.Final.jar
129.	./lib/netty-transport-4.1.45.Final.jar
130.	./lib/netty-transport-native-epoll-4.1.49.Final-linux-x86_64.jar

131.	./lib/netty-transport-native-unix-common-4.1.49.Final.jar
132.	./lib/nio-multipart-parser-1.1.0.jar
133.	./lib/nio-stream-storage-1.1.3.jar
134.	./lib/objenesis-2.6.jar
135.	./lib/opentest4j-1.2.0.jar
136.	./lib/perfmark-api-0.19.0.jar
137.	./lib/profiles-2.13.8.jar
138.	./lib/proto-google-common-protos-1.17.0.jar
139.	./lib/protobuf-java-3.11.3.jar
140.	./lib/protobuf-java-util-3.11.3.jar
141.	./lib/protocol-core-2.13.8.jar
142.	./lib/r2dbc-h2-0.8.3.RELEASE.jar
143.	./lib/r2dbc-pool-0.8.3.RELEASE.jar
144.	./lib/r2dbc-postgresql-0.8.2.RELEASE.jar
145.	./lib/r2dbc-spi-0.8.0.RELEASE.jar
146.	./lib/reactive-streams-1.0.3.jar
147.	./lib/reactor-core-3.3.5.RELEASE.jar
148.	./lib/reactor-netty-0.9.7.RELEASE.jar
149.	./lib/reactor-pool-0.1.3.RELEASE.jar
150.	./lib/regions-2.13.8.jar
151.	./lib/s3-2.13.8.jar
152.	./lib/saslprep-1.1.jar
153.	./lib/sdk-core-2.13.8.jar
154.	./lib/simpleclient-0.8.1.jar
155.	./lib/simpleclient_common-0.8.1.jar
156.	./lib/slf4j-api-1.7.25.jar
157.	./lib/snakeyaml-1.26.jar
158.	./lib/snappy-java-1.1.4.jar
159.	./lib/spring-aop-5.2.6.RELEASE.jar
160.	./lib/spring-beans-5.2.6.RELEASE.jar
161.	./lib/spring-boot-2.3.0.RELEASE.jar
162.	./lib/spring-boot-actuator-2.3.0.RELEASE.jar
163.	./lib/spring-boot-actuator-autoconfigure-2.3.0.RELEASE.jar
164.	./lib/spring-boot-autoconfigure-2.3.0.RELEASE.jar
165.	./lib/spring-boot-configuration-processor-2.3.0.RELEASE.jar
166.	./lib/spring-boot-dependencies-2.3.0.RELEASE.pom
167.	./lib/spring-boot-starter-2.3.0.RELEASE.jar
168.	./lib/spring-boot-starter-actuator-2.3.0.RELEASE.jar

169.	./lib/spring-boot-starter-aop-2.3.0.RELEASE.jar
170.	./lib/spring-boot-starter-json-2.3.0.RELEASE.jar
171.	./lib/spring-boot-starter-logging-2.3.0.RELEASE.jar
172.	./lib/spring-boot-starter-reactor-netty-2.3.0.RELEASE.jar
173.	./lib/spring-boot-starter-web-2.3.0.RELEASE.jar
174.	./lib/spring-boot-starter-webflux-2.3.0.RELEASE.jar
175.	./lib/spring-boot-test-2.3.0.RELEASE.jar
176.	./lib/spring-context-5.2.6.RELEASE.jar
177.	./lib/spring-core-5.2.6.RELEASE.jar
178.	./lib/spring-data-commons-2.3.0.RELEASE.jar
179.	./lib/spring-data-r2dbc-1.1.0.RELEASE.jar
180.	./lib/spring-data-relational-2.0.0.RELEASE.jar
181.	./lib/spring-expression-5.2.6.RELEASE.jar
182.	./lib/spring-jcl-5.2.6.RELEASE.jar
183.	./lib/spring-jdbc-5.2.6.RELEASE.jar
184.	./lib/spring-orm-5.2.6.RELEASE.jar
185.	./lib/spring-tx-5.2.6.RELEASE.jar
186.	./lib/spring-web-5.2.6.RELEASE.jar
187.	./lib/spring-webflux-5.2.6.RELEASE.jar
188.	./lib/spring-webmvc-5.2.6.RELEASE.jar
189.	./lib/stax-ex-1.8.1.jar
190.	./lib/stax2-api-3.1.4.jar
191.	./lib/stringprep-1.1.jar
192.	./lib/txw2-2.3.2.jar
193.	./lib/url-connection-client-2.13.8.jar
194.	./lib/utils-2.13.8.jar
195.	./lib/validation-api-2.0.1.Final.jar
196.	./lib/woodstox-core-5.0.3.jar
197.	./lib/xmlsec-2.1.5.jar
198.	./lib/xmlunit-core-2.7.0.jar

Сервис проверки подписи (verify-signature-service)

Таблица 7. Перечень файлов Изделия, подлежащих контролю целостности

№ п/п.	Имя файла
1.	./lib/FastInfoset-1.2.16.jar
2.	./lib/HdrHistogram-2.1.12.jar

3.	./lib/LatencyUtils-2.0.3.jar
4.	./lib/accessors-smart-1.2.jar
5.	./lib/android-json-0.0.20131108.vaadin1.jar
6.	./lib/animal-sniffer-annotations-1.17.jar
7.	./lib/annotations-13.0.jar
8.	./lib/annotations-2.13.8.jar
9.	./lib/annotations-4.1.1.4.jar
10.	./lib/antlr-2.7.7.jar
11.	./lib/apiguardian-api-1.1.0.jar
12.	./lib/arns-2.13.8.jar
13.	./lib/asm-5.0.4.jar
14.	./lib/aspectjweaver-1.9.5.jar
15.	./lib/assertj-core-3.16.1.jar
16.	./lib/auth-2.13.8.jar
17.	./lib/aws-core-2.13.8.jar
18.	./lib/aws-query-protocol-2.13.8.jar
19.	./lib/aws-xml-protocol-2.13.8.jar
20.	./lib/bcpkix-jdk15on-1.65.jar
21.	./lib/bcprov-jdk15on-1.65.jar
22.	./lib/byte-buddy-1.9.10.jar
23.	./lib/byte-buddy-agent-1.9.10.jar
24.	./lib/checker-qual-2.8.1.jar
25.	./lib/classmate-1.3.4.jar
26.	./lib/client-2.1.jar
27.	./lib/common-2.1.jar
28.	./lib/commons-codec-1.13.jar
29.	./lib/commons-collections-3.2.2.jar
30.	./lib/commons-compiler-3.0.7.jar
31.	./lib/commons-lang3-3.9.jar
32.	./lib/commons-net-3.6.jar
33.	./lib/cryptoserver-bouncycastle-extension-1.0-SNAPSHOT.jar
34.	./lib/cryptoserver-cryptopro-jna-support-1.0-SNAPSHOT.jar
35.	./lib/cryptoserver-cryptopro-provider-1.0-SNAPSHOT.jar
36.	./lib/cryptoserver-dao-1.0-SNAPSHOT.jar
37.	./lib/cryptoserver-extend-signature-services-1.0-SNAPSHOT.jar
38.	./lib/cryptoserver-grpc-api-1.0-SNAPSHOT.jar
39.	./lib/cryptoserver-grpc-client-certificate-verify-service-1.0-SNAPSHOT.jar
40.	./lib/cryptoserver-grpc-support-1.0-SNAPSHOT.jar

41.	./lib/cryptoserver-integrity-base-1.0-SNAPSHOT.jar
42.	./lib/cryptoserver-logging-commons-1.0-SNAPSHOT.jar
43.	./lib/cryptoserver-metrics-commons-1.0-SNAPSHOT.jar
44.	./lib/cryptoserver-model-1.0-SNAPSHOT.jar
45.	./lib/cryptoserver-s3-services-1.0-SNAPSHOT.jar
46.	./lib/cryptoserver-santuario-extension-1.0-SNAPSHOT.jar
47.	./lib/cryptoserver-system-store-1.0-SNAPSHOT.jar
48.	./lib/cryptoserver-tsp-client-services-1.0-SNAPSHOT.jar
49.	./lib/cryptoserver-tsp-server-services-1.0-SNAPSHOT.jar
50.	./lib/cryptoserver-utilities-1.0-SNAPSHOT.jar
51.	./lib/cryptoserver-verify-signature-services-1.0-SNAPSHOT.jar
52.	./lib/dom4j-2.1.3.jar
53.	./lib/error_prone_annotations-2.3.4.jar
54.	./lib/eventstream-1.0.1.jar
55.	./lib/failureaccess-1.0.1.jar
56.	./lib/grpc-api-1.28.0.jar
57.	./lib/grpc-context-1.28.0.jar
58.	./lib/grpc-core-1.28.0.jar
59.	./lib/grpc-netty-1.28.0.jar
60.	./lib/grpc-netty-shaded-1.28.0.jar
61.	./lib/grpc-protobuf-1.28.0.jar
62.	./lib/grpc-protobuf-lite-1.28.0.jar
63.	./lib/grpc-services-1.28.0.jar
64.	./lib/grpc-spring-boot-starter-3.5.3.jar
65.	./lib/grpc-stub-1.28.0.jar
66.	./lib/gson-2.8.6.jar
67.	./lib/guava-28.0-jre.jar
68.	./lib/h2-1.4.200.jar
69.	./lib/hamcrest-2.2.jar
70.	./lib/hibernate-commons-annotations-5.1.0.Final.jar
71.	./lib/hibernate-core-5.4.16.Final.jar
72.	./lib/hibernate-types-52-2.1.1.jar
73.	./lib/hibernate-validator-6.1.5.Final.jar
74.	./lib/http-client-spi-2.13.8.jar
75.	./lib/istack-commons-runtime-3.0.8.jar
76.	./lib/j2objc-annotations-1.3.jar
77.	./lib/jackson-annotations-2.11.0.jar
78.	./lib/jackson-core-2.11.0.jar

79.	./lib/jackson-databind-2.11.0.jar
80.	./lib/jackson-datatype-jdk8-2.11.0.jar
81.	./lib/jackson-datatype-jsr310-2.11.0.jar
82.	./lib/jackson-module-parameter-names-2.11.0.jar
83.	./lib/jakarta.activation-api-1.2.2.jar
84.	./lib/jakarta.annotation-api-1.3.5.jar
85.	./lib/jakarta.validation-api-2.0.2.jar
86.	./lib/jakarta.xml.bind-api-2.3.3.jar
87.	./lib/jandex-2.1.3.Final.jar
88.	./lib/janino-3.0.7.jar
89.	./lib/javassist-3.24.0-GA.jar
90.	./lib/javax.activation-api-1.2.0.jar
91.	./lib/javax.annotation-api-1.3.2.jar
92.	./lib/javax.persistence-api-2.2.jar
93.	./lib/jaxb-api-2.3.1.jar
94.	./lib/jaxb-runtime-2.3.2.jar
95.	./lib/jboss-logging-3.3.2.Final.jar
96.	./lib/jboss-transaction-api_1.2_spec-1.1.1.Final.jar
97.	./lib/jna-5.6.0.jar
98.	./lib/jna-platform-5.6.0.jar
99.	./lib/json-path-2.4.0.jar
100.	./lib/json-smart-2.3.jar
101.	./lib/jsonassert-1.5.0.jar
102.	./lib/jsr305-3.0.2.jar
103.	./lib/jul-to-slf4j-1.7.30.jar
104.	./lib/junit-jupiter-5.6.2.jar
105.	./lib/kafka-clients-1.0.0.jar
106.	./lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
107.	./lib/log4j-api-2.13.2.jar
108.	./lib/log4j-to-slf4j-2.13.2.jar
109.	./lib/logback-classic-1.2.3.jar
110.	./lib/logback-core-1.2.3.jar
111.	./lib/logback-kafka-appender-0.2.0-RC2.jar
112.	./lib/logstash-logback-encoder-5.3.jar
113.	./lib/lombok-1.18.12.jar
114.	./lib/lz4-java-1.4.jar
115.	./lib/micrometer-core-1.5.4.jar
116.	./lib/micrometer-registry-prometheus-1.5.4.jar

117.	./lib/mockito-junit-jupiter-3.3.3.jar
118.	./lib/netty-buffer-4.1.45.Final.jar
119.	./lib/netty-codec-4.1.45.Final.jar
120.	./lib/netty-codec-http-4.1.45.Final.jar
121.	./lib/netty-codec-http2-4.1.45.Final.jar
122.	./lib/netty-codec-socks-4.1.45.Final.jar
123.	./lib/netty-common-4.1.45.Final.jar
124.	./lib/netty-handler-4.1.45.Final.jar
125.	./lib/netty-handler-proxy-4.1.45.Final.jar
126.	./lib/netty-resolver-4.1.45.Final.jar
127.	./lib/netty-tcnative-boringssl-static-2.0.25.Final.jar
128.	./lib/netty-transport-4.1.45.Final.jar
129.	./lib/netty-transport-native-epoll-4.1.49.Final-linux-x86_64.jar
130.	./lib/netty-transport-native-unix-common-4.1.49.Final.jar
131.	./lib/nio-multipart-parser-1.1.0.jar
132.	./lib/nio-stream-storage-1.1.3.jar
133.	./lib/objenesis-2.6.jar
134.	./lib/opentest4j-1.2.0.jar
135.	./lib/perfmark-api-0.19.0.jar
136.	./lib/profiles-2.13.8.jar
137.	./lib/proto-google-common-protos-1.17.0.jar
138.	./lib/protobuf-java-3.11.3.jar
139.	./lib/protobuf-java-util-3.11.3.jar
140.	./lib/protocol-core-2.13.8.jar
141.	./lib/r2dbc-h2-0.8.3.RELEASE.jar
142.	./lib/r2dbc-pool-0.8.3.RELEASE.jar
143.	./lib/r2dbc-postgresql-0.8.2.RELEASE.jar
144.	./lib/r2dbc-spi-0.8.0.RELEASE.jar
145.	./lib/reactive-streams-1.0.3.jar
146.	./lib/reactor-core-3.3.5.RELEASE.jar
147.	./lib/reactor-netty-0.9.7.RELEASE.jar
148.	./lib/reactor-pool-0.1.3.RELEASE.jar
149.	./lib/regions-2.13.8.jar
150.	./lib/s3-2.13.8.jar
151.	./lib/saslprep-1.1.jar
152.	./lib/sdk-core-2.13.8.jar
153.	./lib/simpleclient-0.8.1.jar
154.	./lib/simpleclient_common-0.8.1.jar

155.	./lib/slf4j-api-1.7.25.jar
156.	./lib/snakeyaml-1.26.jar
157.	./lib/snappy-java-1.1.4.jar
158.	./lib/spring-aop-5.2.6.RELEASE.jar
159.	./lib/spring-beans-5.2.6.RELEASE.jar
160.	./lib/spring-boot-2.3.0.RELEASE.jar
161.	./lib/spring-boot-actuator-2.3.0.RELEASE.jar
162.	./lib/spring-boot-actuator-autoconfigure-2.3.0.RELEASE.jar
163.	./lib/spring-boot-autoconfigure-2.3.0.RELEASE.jar
164.	./lib/spring-boot-configuration-processor-2.3.0.RELEASE.jar
165.	./lib/spring-boot-dependencies-2.3.0.RELEASE.pom
166.	./lib/spring-boot-starter-2.3.0.RELEASE.jar
167.	./lib/spring-boot-starter-actuator-2.3.0.RELEASE.jar
168.	./lib/spring-boot-starter-aop-2.3.0.RELEASE.jar
169.	./lib/spring-boot-starter-json-2.3.0.RELEASE.jar
170.	./lib/spring-boot-starter-logging-2.3.0.RELEASE.jar
171.	./lib/spring-boot-starter-reactor-netty-2.3.0.RELEASE.jar
172.	./lib/spring-boot-starter-web-2.3.0.RELEASE.jar
173.	./lib/spring-boot-starter-webflux-2.3.0.RELEASE.jar
174.	./lib/spring-boot-test-2.3.0.RELEASE.jar
175.	./lib/spring-context-5.2.6.RELEASE.jar
176.	./lib/spring-core-5.2.6.RELEASE.jar
177.	./lib/spring-data-commons-2.3.0.RELEASE.jar
178.	./lib/spring-data-r2dbc-1.1.0.RELEASE.jar
179.	./lib/spring-data-relational-2.0.0.RELEASE.jar
180.	./lib/spring-expression-5.2.6.RELEASE.jar
181.	./lib/spring-jcl-5.2.6.RELEASE.jar
182.	./lib/spring-jdbc-5.2.6.RELEASE.jar
183.	./lib/spring-orm-5.2.6.RELEASE.jar
184.	./lib/spring-tx-5.2.6.RELEASE.jar
185.	./lib/spring-web-5.2.6.RELEASE.jar
186.	./lib/spring-webflux-5.2.6.RELEASE.jar
187.	./lib/spring-webmvc-5.2.6.RELEASE.jar
188.	./lib/stax-ex-1.8.1.jar
189.	./lib/stax2-api-3.1.4.jar
190.	./lib/stringprep-1.1.jar
191.	./lib/txw2-2.3.2.jar
192.	./lib/url-connection-client-2.13.8.jar

193.	./lib/utills-2.13.8.jar
194.	./lib/validation-api-2.0.1.Final.jar
195.	./lib/woodstox-core-5.0.3.jar
196.	./lib/xmlsec-2.1.5.jar
197.	./lib/xmlunit-core-2.7.0.jar
198.	./verify-signature-service.jar

Мост интеграции kafka с сервисом подписания (create-signature-bridge)

Таблица 8. Перечень файлов Изделия, подлежащих контролю целостности

№ п/п.	Имя файла
1.	./create-signature-bridge.jar
2.	./lib/FastInfoset-1.2.16.jar
3.	./lib/HdrHistogram-2.1.12.jar
4.	./lib/LatencyUtils-2.0.3.jar
5.	./lib/accessors-smart-1.2.jar
6.	./lib/android-json-0.0.20131108.vaadin1.jar
7.	./lib/animal-sniffer-annotations-1.17.jar
8.	./lib/annotations-4.1.1.4.jar
9.	./lib/apiguardian-api-1.1.0.jar
10.	./lib/asm-5.0.4.jar
11.	./lib/aspectjweaver-1.9.5.jar
12.	./lib/assertj-core-3.16.1.jar
13.	./lib/bcpkix-jdk15on-1.65.jar
14.	./lib/bcprov-jdk15on-1.65.jar
15.	./lib/byte-buddy-1.9.10.jar
16.	./lib/byte-buddy-agent-1.9.10.jar
17.	./lib/checker-qual-2.8.1.jar
18.	./lib/classmate-1.3.4.jar
19.	./lib/commons-codec-1.13.jar
20.	./lib/commons-collections-3.2.2.jar
21.	./lib/commons-compiler-3.0.7.jar
22.	./lib/commons-lang3-3.9.jar
23.	./lib/commons-net-3.6.jar
24.	./lib/cryptoserver-grpc-api-1.0-SNAPSHOT.jar
25.	./lib/cryptoserver-grpc-client-create-signature-service-1.0-SNAPSHOT.jar
26.	./lib/cryptoserver-grpc-client-signature-rule-service-1.0-SNAPSHOT.jar

27.	./lib/cryptoserver-integrity-base-1.0-SNAPSHOT.jar
28.	./lib/cryptoserver-kafka-commons-1.0-SNAPSHOT.jar
29.	./lib/cryptoserver-logging-commons-1.0-SNAPSHOT.jar
30.	./lib/cryptoserver-utilities-1.0-SNAPSHOT.jar
31.	./lib/error_prone_annotations-2.3.4.jar
32.	./lib/failureaccess-1.0.1.jar
33.	./lib/grpc-api-1.28.0.jar
34.	./lib/grpc-context-1.28.0.jar
35.	./lib/grpc-core-1.28.0.jar
36.	./lib/grpc-netty-1.28.0.jar
37.	./lib/grpc-protobuf-1.28.0.jar
38.	./lib/grpc-protobuf-lite-1.28.0.jar
39.	./lib/grpc-services-1.28.0.jar
40.	./lib/grpc-spring-boot-starter-3.5.3.jar
41.	./lib/grpc-stub-1.28.0.jar
42.	./lib/gson-2.8.6.jar
43.	./lib/guava-28.0-jre.jar
44.	./lib/hamcrest-2.2.jar
45.	./lib/hibernate-validator-6.1.5.Final.jar
46.	./lib/istack-commons-runtime-3.0.8.jar
47.	./lib/j2objc-annotations-1.3.jar
48.	./lib/jackson-annotations-2.11.0.jar
49.	./lib/jackson-core-2.11.0.jar
50.	./lib/jackson-databind-2.11.0.jar
51.	./lib/jackson-datatype-jdk8-2.11.0.jar
52.	./lib/jackson-datatype-jsr310-2.11.0.jar
53.	./lib/jackson-module-parameter-names-2.11.0.jar
54.	./lib/jakarta.activation-api-1.2.2.jar
55.	./lib/jakarta.annotation-api-1.3.5.jar
56.	./lib/jakarta.validation-api-2.0.2.jar
57.	./lib/jakarta.xml.bind-api-2.3.3.jar
58.	./lib/janino-3.0.7.jar
59.	./lib/javax.activation-api-1.2.0.jar
60.	./lib/javax.annotation-api-1.3.2.jar
61.	./lib/jaxb-api-2.3.1.jar
62.	./lib/jaxb-runtime-2.3.2.jar
63.	./lib/jboss-logging-3.3.2.Final.jar
64.	./lib/json-path-2.4.0.jar

65.	./lib/json-smart-2.3.jar
66.	./lib/jsonassert-1.5.0.jar
67.	./lib/jsr305-3.0.2.jar
68.	./lib/jul-to-slf4j-1.7.30.jar
69.	./lib/junit-jupiter-5.6.2.jar
70.	./lib/kafka-clients-2.5.1.jar
71.	./lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
72.	./lib/log4j-api-2.13.2.jar
73.	./lib/log4j-to-slf4j-2.13.2.jar
74.	./lib/logback-classic-1.2.3.jar
75.	./lib/logback-core-1.2.3.jar
76.	./lib/logback-kafka-appender-0.2.0-RC2.jar
77.	./lib/logstash-logback-encoder-5.3.jar
78.	./lib/lombok-1.18.12.jar
79.	./lib/lz4-java-1.7.1.jar
80.	./lib/micrometer-core-1.5.4.jar
81.	./lib/mockito-junit-jupiter-3.3.3.jar
82.	./lib/netty-buffer-4.1.45.Final.jar
83.	./lib/netty-codec-4.1.45.Final.jar
84.	./lib/netty-codec-http-4.1.45.Final.jar
85.	./lib/netty-codec-http2-4.1.45.Final.jar
86.	./lib/netty-codec-socks-4.1.45.Final.jar
87.	./lib/netty-common-4.1.45.Final.jar
88.	./lib/netty-handler-4.1.45.Final.jar
89.	./lib/netty-handler-proxy-4.1.45.Final.jar
90.	./lib/netty-resolver-4.1.45.Final.jar
91.	./lib/netty-tcnative-boringssl-static-2.0.25.Final.jar
92.	./lib/netty-transport-4.1.45.Final.jar
93.	./lib/objenesis-2.6.jar
94.	./lib/opentest4j-1.2.0.jar
95.	./lib/perfmark-api-0.19.0.jar
96.	./lib/proto-google-common-protos-1.17.0.jar
97.	./lib/protobuf-java-3.11.3.jar
98.	./lib/protobuf-java-util-3.11.3.jar
99.	./lib/slf4j-api-1.7.25.jar
100.	./lib/snakeyaml-1.26.jar
101.	./lib/snappy-java-1.1.7.3.jar
102.	./lib/spring-aop-5.2.6.RELEASE.jar

103.	./lib/spring-beans-5.2.6.RELEASE.jar
104.	./lib/spring-boot-2.3.0.RELEASE.jar
105.	./lib/spring-boot-actuator-2.3.0.RELEASE.jar
106.	./lib/spring-boot-actuator-autoconfigure-2.3.0.RELEASE.jar
107.	./lib/spring-boot-autoconfigure-2.3.0.RELEASE.jar
108.	./lib/spring-boot-configuration-processor-2.3.0.RELEASE.jar
109.	./lib/spring-boot-starter-2.3.0.RELEASE.jar
110.	./lib/spring-boot-starter-actuator-2.3.0.RELEASE.jar
111.	./lib/spring-boot-starter-aop-2.3.0.RELEASE.jar
112.	./lib/spring-boot-starter-json-2.3.0.RELEASE.jar
113.	./lib/spring-boot-starter-logging-2.3.0.RELEASE.jar
114.	./lib/spring-boot-starter-web-2.3.0.RELEASE.jar
115.	./lib/spring-boot-test-2.3.0.RELEASE.jar
116.	./lib/spring-context-5.2.6.RELEASE.jar
117.	./lib/spring-core-5.2.6.RELEASE.jar
118.	./lib/spring-expression-5.2.6.RELEASE.jar
119.	./lib/spring-jcl-5.2.6.RELEASE.jar
120.	./lib/spring-kafka-2.5.5.RELEASE.jar
121.	./lib/spring-messaging-5.2.8.RELEASE.jar
122.	./lib/spring-retry-1.2.5.RELEASE.jar
123.	./lib/spring-tx-5.2.8.RELEASE.jar
124.	./lib/spring-web-5.2.6.RELEASE.jar
125.	./lib/spring-webmvc-5.2.6.RELEASE.jar
126.	./lib/stax-ex-1.8.1.jar
127.	./lib/stax2-api-3.1.4.jar
128.	./lib/txw2-2.3.2.jar
129.	./lib/woodstox-core-5.0.3.jar
130.	./lib/xmlsec-2.1.5.jar
131.	./lib/xmlunit-core-2.7.0.jar
132.	./lib/zstd-jni-1.4.4-7.jar

Мост интеграции kafka с сервисом проверки подписи (verify-signature-bridge)

Таблица 9. Перечень файлов Изделия, подлежащих контролю целостности

№ п/п.	Имя файла
1.	./lib/FastInfoset-1.2.16.jar
2.	./lib/HdrHistogram-2.1.12.jar

3.	./lib/LatencyUtils-2.0.3.jar
4.	./lib/accessors-smart-1.2.jar
5.	./lib/android-json-0.0.20131108.vaadin1.jar
6.	./lib/animal-sniffer-annotations-1.17.jar
7.	./lib/annotations-4.1.1.4.jar
8.	./lib/apiguardian-api-1.1.0.jar
9.	./lib/asm-5.0.4.jar
10.	./lib/aspectjweaver-1.9.5.jar
11.	./lib/assertj-core-3.16.1.jar
12.	./lib/bcpkix-jdk15on-1.65.jar
13.	./lib/bcprov-jdk15on-1.65.jar
14.	./lib/byte-buddy-1.9.10.jar
15.	./lib/byte-buddy-agent-1.9.10.jar
16.	./lib/checker-qual-2.8.1.jar
17.	./lib/classmate-1.3.4.jar
18.	./lib/commons-codec-1.13.jar
19.	./lib/commons-collections-3.2.2.jar
20.	./lib/commons-compiler-3.0.7.jar
21.	./lib/commons-lang3-3.9.jar
22.	./lib/commons-net-3.6.jar
23.	./lib/cryptoserver-grpc-api-1.0-SNAPSHOT.jar
24.	./lib/cryptoserver-grpc-client-signature-rule-service-1.0-SNAPSHOT.jar
25.	./lib/cryptoserver-grpc-client-verify-signature-service-1.0-SNAPSHOT.jar
26.	./lib/cryptoserver-integrity-base-1.0-SNAPSHOT.jar
27.	./lib/cryptoserver-kafka-commons-1.0-SNAPSHOT.jar
28.	./lib/cryptoserver-logging-commons-1.0-SNAPSHOT.jar
29.	./lib/cryptoserver-utilities-1.0-SNAPSHOT.jar
30.	./lib/error_prone_annotations-2.3.4.jar
31.	./lib/failureaccess-1.0.1.jar
32.	./lib/grpc-api-1.28.0.jar
33.	./lib/grpc-context-1.28.0.jar
34.	./lib/grpc-core-1.28.0.jar
35.	./lib/grpc-netty-1.28.0.jar
36.	./lib/grpc-protobuf-1.28.0.jar
37.	./lib/grpc-protobuf-lite-1.28.0.jar
38.	./lib/grpc-services-1.28.0.jar
39.	./lib/grpc-spring-boot-starter-3.5.3.jar
40.	./lib/grpc-stub-1.28.0.jar

41.	./lib/gson-2.8.6.jar
42.	./lib/guava-28.0-jre.jar
43.	./lib/hamcrest-2.2.jar
44.	./lib/hibernate-validator-6.1.5.Final.jar
45.	./lib/istack-commons-runtime-3.0.8.jar
46.	./lib/j2objc-annotations-1.3.jar
47.	./lib/jackson-annotations-2.11.0.jar
48.	./lib/jackson-core-2.11.0.jar
49.	./lib/jackson-databind-2.11.0.jar
50.	./lib/jackson-datatype-jdk8-2.11.0.jar
51.	./lib/jackson-datatype-jsr310-2.11.0.jar
52.	./lib/jackson-module-parameter-names-2.11.0.jar
53.	./lib/jakarta.activation-api-1.2.2.jar
54.	./lib/jakarta.annotation-api-1.3.5.jar
55.	./lib/jakarta.validation-api-2.0.2.jar
56.	./lib/jakarta.xml.bind-api-2.3.3.jar
57.	./lib/janino-3.0.7.jar
58.	./lib/javax.activation-api-1.2.0.jar
59.	./lib/javax.annotation-api-1.3.2.jar
60.	./lib/jaxb-api-2.3.1.jar
61.	./lib/jaxb-runtime-2.3.2.jar
62.	./lib/jboss-logging-3.3.2.Final.jar
63.	./lib/json-path-2.4.0.jar
64.	./lib/json-smart-2.3.jar
65.	./lib/jsonassert-1.5.0.jar
66.	./lib/jsr305-3.0.2.jar
67.	./lib/jul-to-slf4j-1.7.30.jar
68.	./lib/junit-jupiter-5.6.2.jar
69.	./lib/kafka-clients-2.5.1.jar
70.	./lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
71.	./lib/log4j-api-2.13.2.jar
72.	./lib/log4j-to-slf4j-2.13.2.jar
73.	./lib/logback-classic-1.2.3.jar
74.	./lib/logback-core-1.2.3.jar
75.	./lib/logback-kafka-appender-0.2.0-RC2.jar
76.	./lib/logstash-logback-encoder-5.3.jar
77.	./lib/lombok-1.18.12.jar
78.	./lib/lz4-java-1.7.1.jar

79.	./lib/micrometer-core-1.5.4.jar
80.	./lib/mockito-junit-jupiter-3.3.3.jar
81.	./lib/netty-buffer-4.1.45.Final.jar
82.	./lib/netty-codec-4.1.45.Final.jar
83.	./lib/netty-codec-http-4.1.45.Final.jar
84.	./lib/netty-codec-http2-4.1.45.Final.jar
85.	./lib/netty-codec-socks-4.1.45.Final.jar
86.	./lib/netty-common-4.1.45.Final.jar
87.	./lib/netty-handler-4.1.45.Final.jar
88.	./lib/netty-handler-proxy-4.1.45.Final.jar
89.	./lib/netty-resolver-4.1.45.Final.jar
90.	./lib/netty-tcnative-boringssl-static-2.0.25.Final.jar
91.	./lib/netty-transport-4.1.45.Final.jar
92.	./lib/objenesis-2.6.jar
93.	./lib/opentest4j-1.2.0.jar
94.	./lib/perfmark-api-0.19.0.jar
95.	./lib/proto-google-common-protos-1.17.0.jar
96.	./lib/protobuf-java-3.11.3.jar
97.	./lib/protobuf-java-util-3.11.3.jar
98.	./lib/slf4j-api-1.7.25.jar
99.	./lib/snakeyaml-1.26.jar
100.	./lib/snappy-java-1.1.7.3.jar
101.	./lib/spring-aop-5.2.6.RELEASE.jar
102.	./lib/spring-beans-5.2.6.RELEASE.jar
103.	./lib/spring-boot-2.3.0.RELEASE.jar
104.	./lib/spring-boot-actuator-2.3.0.RELEASE.jar
105.	./lib/spring-boot-actuator-autoconfigure-2.3.0.RELEASE.jar
106.	./lib/spring-boot-autoconfigure-2.3.0.RELEASE.jar
107.	./lib/spring-boot-configuration-processor-2.3.0.RELEASE.jar
108.	./lib/spring-boot-starter-2.3.0.RELEASE.jar
109.	./lib/spring-boot-starter-actuator-2.3.0.RELEASE.jar
110.	./lib/spring-boot-starter-aop-2.3.0.RELEASE.jar
111.	./lib/spring-boot-starter-json-2.3.0.RELEASE.jar
112.	./lib/spring-boot-starter-logging-2.3.0.RELEASE.jar
113.	./lib/spring-boot-starter-web-2.3.0.RELEASE.jar
114.	./lib/spring-boot-test-2.3.0.RELEASE.jar
115.	./lib/spring-context-5.2.6.RELEASE.jar
116.	./lib/spring-core-5.2.6.RELEASE.jar

117.	./lib/spring-expression-5.2.6.RELEASE.jar
118.	./lib/spring-jcl-5.2.6.RELEASE.jar
119.	./lib/spring-kafka-2.5.5.RELEASE.jar
120.	./lib/spring-messaging-5.2.8.RELEASE.jar
121.	./lib/spring-retry-1.2.5.RELEASE.jar
122.	./lib/spring-tx-5.2.8.RELEASE.jar
123.	./lib/spring-web-5.2.6.RELEASE.jar
124.	./lib/spring-webmvc-5.2.6.RELEASE.jar
125.	./lib/stax-ex-1.8.1.jar
126.	./lib/stax2-api-3.1.4.jar
127.	./lib/txw2-2.3.2.jar
128.	./lib/woodstox-core-5.0.3.jar
129.	./lib/xmlsec-2.1.5.jar
130.	./lib/xmlunit-core-2.7.0.jar
131.	./lib/zstd-jni-1.4.4-7.jar
132.	./verify-signature-bridge.jar

Backend-модуль сервиса управления (control-module-backend)

Таблица 10. Перечень файлов Изделия, подлежащих контролю целостности

№ п/п.	Имя файла
1.	./control-module-backend.jar
2.	./lib/FastInfoset-1.2.16.jar
3.	./lib/HdrHistogram-2.1.12.jar
4.	./lib/LatencyUtils-2.0.3.jar
5.	./lib/accessors-smart-1.2.jar
6.	./lib/android-json-0.0.20131108.vaadin1.jar
7.	./lib/animal-sniffer-annotations-1.17.jar
8.	./lib/annotations-13.0.jar
9.	./lib/annotations-4.1.1.4.jar
10.	./lib/apiguardian-api-1.1.0.jar
11.	./lib/asm-5.0.4.jar
12.	./lib/aspectjweaver-1.9.5.jar
13.	./lib/assertj-core-3.16.1.jar
14.	./lib/bcpkix-jdk15on-1.65.jar
15.	./lib/bcprov-jdk15on-1.65.jar
16.	./lib/byte-buddy-1.9.10.jar

17.	./lib/byte-buddy-agent-1.9.10.jar
18.	./lib/checker-qual-2.8.1.jar
19.	./lib/classmate-1.3.4.jar
20.	./lib/commons-codec-1.13.jar
21.	./lib/commons-collections-3.2.2.jar
22.	./lib/commons-compiler-3.0.7.jar
23.	./lib/commons-lang3-3.9.jar
24.	./lib/commons-net-3.6.jar
25.	./lib/cryptoserver-grpc-api-1.0-SNAPSHOT.jar
26.	./lib/cryptoserver-grpc-support-1.0-SNAPSHOT.jar
27.	./lib/cryptoserver-integrity-base-1.0-SNAPSHOT.jar
28.	./lib/cryptoserver-logging-commons-1.0-SNAPSHOT.jar
29.	./lib/cryptoserver-metrics-commons-1.0-SNAPSHOT.jar
30.	./lib/cryptoserver-utilities-1.0-SNAPSHOT.jar
31.	./lib/de.huxhorn.sulky.ulid-8.2.0.jar
32.	./lib/error_prone_annotations-2.3.4.jar
33.	./lib/execution-model-6.0.1.jar
34.	./lib/failureaccess-1.0.1.jar
35.	./lib/grpc-api-1.28.0.jar
36.	./lib/grpc-context-1.28.0.jar
37.	./lib/grpc-core-1.28.0.jar
38.	./lib/grpc-netty-1.28.0.jar
39.	./lib/grpc-netty-shaded-1.28.0.jar
40.	./lib/grpc-protobuf-1.28.0.jar
41.	./lib/grpc-protobuf-lite-1.28.0.jar
42.	./lib/grpc-services-1.28.0.jar
43.	./lib/grpc-spring-boot-starter-3.5.3.jar
44.	./lib/grpc-stub-1.28.0.jar
45.	./lib/gson-2.8.6.jar
46.	./lib/guava-28.0-jre.jar
47.	./lib/hamcrest-2.2.jar
48.	./lib/hibernate-validator-6.1.5.Final.jar
49.	./lib/istack-commons-runtime-3.0.8.jar
50.	./lib/j2objc-annotations-1.3.jar
51.	./lib/jackson-annotations-2.11.0.jar
52.	./lib/jackson-core-2.11.0.jar
53.	./lib/jackson-databind-2.11.0.jar
54.	./lib/jackson-datatype-jdk8-2.11.0.jar

55.	./lib/jackson-datatype-jsr310-2.11.0.jar
56.	./lib/jackson-module-parameter-names-2.11.0.jar
57.	./lib/jakarta.activation-api-1.2.2.jar
58.	./lib/jakarta.annotation-api-1.3.5.jar
59.	./lib/jakarta.validation-api-2.0.2.jar
60.	./lib/jakarta.xml.bind-api-2.3.3.jar
61.	./lib/janino-3.0.7.jar
62.	./lib/javax.activation-api-1.2.0.jar
63.	./lib/javax.annotation-api-1.3.2.jar
64.	./lib/jaxb-api-2.3.1.jar
65.	./lib/jaxb-runtime-2.3.2.jar
66.	./lib/jboss-logging-3.3.2.Final.jar
67.	./lib/json-path-2.4.0.jar
68.	./lib/json-smart-2.3.jar
69.	./lib/jsonassert-1.5.0.jar
70.	./lib/jsr305-3.0.2.jar
71.	./lib/jul-to-slf4j-1.7.30.jar
72.	./lib/junit-jupiter-5.6.2.jar
73.	./lib/kafka-clients-1.0.0.jar
74.	./lib/kotlin-reflect-1.3.72.jar
75.	./lib/kotlin-stdlib-1.3.72.jar
76.	./lib/kotlin-stdlib-common-1.3.72.jar
77.	./lib/kotlin-stdlib-jdk7-1.3.72.jar
78.	./lib/kotlin-stdlib-jdk8-1.3.72.jar
79.	./lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
80.	./lib/log4j-api-2.13.2.jar
81.	./lib/log4j-to-slf4j-2.13.2.jar
82.	./lib/logback-classic-1.2.3.jar
83.	./lib/logback-core-1.2.3.jar
84.	./lib/logback-kafka-appender-0.2.0-RC2.jar
85.	./lib/logstash-logback-encoder-5.3.jar
86.	./lib/lombok-1.18.12.jar
87.	./lib/lz4-java-1.4.jar
88.	./lib/micrometer-core-1.5.4.jar
89.	./lib/micrometer-registry-prometheus-1.5.4.jar
90.	./lib/mockito-junit-jupiter-3.3.3.jar
91.	./lib/netty-buffer-4.1.45.Final.jar
92.	./lib/netty-codec-4.1.45.Final.jar

93.	./lib/netty-codec-http-4.1.45.Final.jar
94.	./lib/netty-codec-http2-4.1.45.Final.jar
95.	./lib/netty-codec-socks-4.1.45.Final.jar
96.	./lib/netty-common-4.1.45.Final.jar
97.	./lib/netty-handler-4.1.45.Final.jar
98.	./lib/netty-handler-proxy-4.1.45.Final.jar
99.	./lib/netty-resolver-4.1.45.Final.jar
100.	./lib/netty-tcnative-boringssl-static-2.0.25.Final.jar
101.	./lib/netty-transport-4.1.45.Final.jar
102.	./lib/netty-transport-native-epoll-4.1.49.Final-linux-x86_64.jar
103.	./lib/netty-transport-native-unix-common-4.1.49.Final.jar
104.	./lib/nio-multipart-parser-1.1.0.jar
105.	./lib/nio-stream-storage-1.1.3.jar
106.	./lib/objenesis-2.6.jar
107.	./lib/opentest4j-1.2.0.jar
108.	./lib/perfmark-api-0.19.0.jar
109.	./lib/proto-google-common-protos-1.17.0.jar
110.	./lib/protobuf-java-3.11.3.jar
111.	./lib/protobuf-java-util-3.11.3.jar
112.	./lib/reactive-streams-1.0.3.jar
113.	./lib/reactor-core-3.3.5.RELEASE.jar
114.	./lib/reactor-kotlin-extensions-1.0.0.RELEASE.jar
115.	./lib/reactor-netty-0.9.7.RELEASE.jar
116.	./lib/simpleclient-0.8.1.jar
117.	./lib/simpleclient_common-0.8.1.jar
118.	./lib/slf4j-api-1.7.25.jar
119.	./lib/snakeyaml-1.26.jar
120.	./lib/snappy-java-1.1.4.jar
121.	./lib/spring-aop-5.2.6.RELEASE.jar
122.	./lib/spring-beans-5.2.6.RELEASE.jar
123.	./lib/spring-boot-2.3.0.RELEASE.jar
124.	./lib/spring-boot-actuator-2.3.0.RELEASE.jar
125.	./lib/spring-boot-actuator-autoconfigure-2.3.0.RELEASE.jar
126.	./lib/spring-boot-autoconfigure-2.3.0.RELEASE.jar
127.	./lib/spring-boot-configuration-processor-2.3.0.RELEASE.jar
128.	./lib/spring-boot-starter-2.3.0.RELEASE.jar
129.	./lib/spring-boot-starter-actuator-2.3.0.RELEASE.jar
130.	./lib/spring-boot-starter-aop-2.3.0.RELEASE.jar

131.	./lib/spring-boot-starter-json-2.3.0.RELEASE.jar
132.	./lib/spring-boot-starter-logging-2.3.0.RELEASE.jar
133.	./lib/spring-boot-starter-reactor-netty-2.3.0.RELEASE.jar
134.	./lib/spring-boot-starter-web-2.3.0.RELEASE.jar
135.	./lib/spring-boot-starter-webflux-2.3.0.RELEASE.jar
136.	./lib/spring-boot-test-2.3.0.RELEASE.jar
137.	./lib/spring-context-5.2.6.RELEASE.jar
138.	./lib/spring-core-5.2.6.RELEASE.jar
139.	./lib/spring-expression-5.2.6.RELEASE.jar
140.	./lib/spring-jcl-5.2.6.RELEASE.jar
141.	./lib/spring-web-5.2.6.RELEASE.jar
142.	./lib/spring-webflux-5.2.6.RELEASE.jar
143.	./lib/spring-webmvc-5.2.6.RELEASE.jar
144.	./lib/stax-ex-1.8.1.jar
145.	./lib/stax2-api-3.1.4.jar
146.	./lib/txw2-2.3.2.jar
147.	./lib/ulid-6.0.1.jar
148.	./lib/woodstox-core-5.0.3.jar
149.	./lib/xmlsec-2.1.5.jar
150.	./lib/xmlunit-core-2.7.0.jar

СПИСОК ИЗМЕНЕНИЙ

Версия	Дата	Внесенные изменения	Исполнитель
1.0	01.10.2020	Создание документа	Андреев И.Н.