Клиентская JAVA-библиотека для сервера ОЭП

11485466.72.21.12.112

Инструкция по установке и эксплуатации

Содержание

1 Введение

Настоящий документ содержит руководство по установке и эксплуатации программного изделия «Клиентская JAVA-библиотека для сервера ОЭП». Руководство включает в себя справочную информацию по работе с библиотекой «Клиентская JAVA-библиотека для сервера ОЭП».

2 Назначение и условия применения

2.1 Назначение системы

«Клиентская JAVA-библиотека для сервера ОЭП» представляет собой библиотеку JAVA, которая предназначена для предоставления удобного мультиплатформенного программного интерфейса к программному изделию «Сервер ОЭП» в составе ПАК ФПСУ-IP.

В программном изделии «Клиентская JAVA-библиотека для сервера ОЭП» реализовано выполнение с помощью программного изделия «Сервер ОЭП» следующих основных функций:

- Создание и установка мастер-ключа.
- Создание ключа ОЭП и его шифрование с помощью мастер-ключа.
- Создание облачной электронной подписи в соответствии с требованиями ГОСТ Р 34.10-2001 и ГОСТ Р 34.10-2012.

2.2 Условия применения системы

«Клиентская JAVA-библиотека для сервера ОЭП» должна работать под управлением ОС, поддерживающих среду JVM версий 1.6, 1.7 и 1.8.

Для работы программного изделия «Клиентская JAVA-библиотека для сервера ОЭП» необходим сетевой доступ к ПАК ФПСУ-IP, на котором установлено программное изделие «Сервер ОЭП».

3 Установка программного изделия «Клиентская JAVA-библиотека для сервера ОЭП»

Для того чтобы установить программное изделие «Клиентская JAVA-библиотека для сервера ОЭП», следует скопировать содержимое дистрибутива «Клиентская JAVA-библиотека для сервера ОЭП» на жёсткий диск компьютера.

4 Удаление программного изделия «Клиентская JAVA-библиотека для сервера ОЭП»

Для того чтобы удалить программное изделие «Клиентская JAVA-библиотека для сервера ОЭП», необходимо удалить с жёсткого диска компьютера ранее установленные файлы дистрибутива «Клиентская JAVA-библиотека для сервера ОЭП».

5 Описание библиотеки

5.1 Пакеты библиотеки «Клиентская JAVA-библиотека для сервера ОЭП»

В состав библиотеки «Клиентская JAVA-библиотека для сервера ОЭП» входят два пакета:

- ru.infocrypt.crypto.enums
- ru.infocrypt.fpsu

5.2 Классы пакета ru.infocrypt.crypto.enums

В состав пакета ru.infocrypt.crypto.enums входит один класс – перечисление PublicKeyParamSet.

5.2.1 Перечисление PublicKeyParamSet

```
java.lang.Object
  java.lang.Enum<PublicKeyParamSet>
      ru.infocrypt.crypto.enums.PublicKeyParamSet
-----------------------------------------------------------------------
      public enum PublicKeyParamSet
      extends java.lang.Enum<PublicKeyParamSet>
```

Описание

Перечисление значений параметров открытого ключа по ГОСТ Р 34.10-2001 и ГОСТ Р 34.10-2012.

Константы

Константа	Описание
A	public static final PublicKeyParamSet A
A_2001_TK26_256	public static final PublicKeyParamSet A_2001_TK26_256
A_TK26_256	public static final PublicKeyParamSet A_TK26_256
A_TK26_512	public static final PublicKeyParamSet A_TK26_512
B	public static final PublicKeyParamSet B
B_2001_TK26_256	public static final PublicKeyParamSet B_2001_TK26_256
B_TK26_256	public static final PublicKeyParamSet B_TK26_256
B_TK26_512	public static final PublicKeyParamSet B_TK26_512
C	public static final PublicKeyParamSet C
C_2001_TK26_256	public static final PublicKeyParamSet C_2001_TK26_256
C_TK26_256	public static final PublicKeyParamSet C_TK26_256
C_TK26_512	public static final PublicKeyParamSet C_TK26_512
D_TK26_256	public static final PublicKeyParamSet D_TK26_256
RSA	public static final PublicKeyParamSet RSA
X_2001_TK26_256	public static final PublicKeyParamSet X_2001_TK26_256
XchA	public static final PublicKeyParamSet XchA
XchB	public static final PublicKeyParamSet XchB
Y_2001_TK26_256	public static final PublicKeyParamSet Y_2001_TK26_256

Методы

Модификатор и тип	Метод и описание
ru.infocrypt.crypto.enums.DigestParamSet	getDigestParamSet() Получить значение перечисления

Модификатор и тип	Метод и описание
byte	letter() Значение перечисления типа byte
static PublicKeyParamSet	lookup(byte value) Определение параметров по «Бикриптовским» буквам
static PublicKeyParamSet	lookup(int value) Поиск значения перечисления по числовому значению
java.lang.String	toString() Текстовое описание
int	value() Числовое значение
static PublicKeyParamSet	valueOf(java.lang.String name) Возвращает константу перечисления данного типа с указанным именем
static PublicKeyParamSet[]	values() Возвращает массив констант в порядке, в котором они были указаны в перечислении данного типа

Метод getMaskedKey

```
public ru.infocrypt.crypto.enums.DigestParamSet getDigestParamSet()
```

Метод letter

```
public byte letter()
```

Метод lookup

```
public static PublicKeyParamSet lookup(byte value)
```

Описание:
Определение параметров по «Бикриптовским» буквам

Параметры:
`value - буква`

Возвращаемое значение:
`экземпляр PublicKeyParamSet`

Метод lookup

```
public static PublicKeyParamSet lookup(int value)
```

Метод toString

```
public java.lang.String toString()
```

Переопределение:
toString в классе java.lang.Enum<PublicKeyParamSet>

Метод value

```
public int value()
```

Метод valueOf

```
public static PublicKeyParamSet valueOf(java.lang.String name)
```

Описание:
Возвращает константу перечисления данного типа с указанным именем. Строка должна точно соответствовать идентификатору константы, указанному в перечислении данного типа. (Лишние пробелы недопустимы.)

Параметры:
name – возвращаемое имя константы перечисления.

Возвращаемое значение:

константа перечисления данного типа с указанным именем

Исключения:
java.lang.IllegalArgumentException – если перечисление данного типа не содержит константу с указанным именем

java.lang.NullPointerException – если аргумент равен null

Метод values

```
public static PublicKeyParamSet[] values()
```

Описание:
Возвращает массив констант в порядке, в котором они были указаны в перечислении данного типа. Данный метод позволяет перебрать константы следующим образом:

```
for (PublicKeyParamSet c : PublicKeyParamSet.values())
    System.out.println(c);
```

Возвращаемое значение:

Массив констант в порядке, в котором они были указаны в перечислении данного типа.

5.3 Классы пакета ru.infocrypt.fpsu

В состав пакета ru.infocrypt.fpsu входят классы:

- CloudSign,
- CSKeyPair,
- MaskedKey.

5.3.1 Класс CloudSign

```
java.lang.Object

        ru.infocrypt.fpsu.CloudSign
-------------------------------------------------------------------------

      public class CloudSign

      extends java.lang.Object
```

Описание

Облачная подпись.

Конструкторы

CloudSign (java.lang.String addr, int port, int timeout) – инициализация соединения с сервером облачной подписи.

```
public CloudSign(java.lang.String addr,
                 int port,
                 int timeout)
          throws ru.infocrypt.fpsu.exception.FpsuException
```

Параметры:

```
      addr - адрес сервера

      port - порт

      timeout - таймаут сессии с сервером
```

Исключения:

```
      ru.infocrypt.fpsu.exception.FpsuException - возможные исключения
```

CloudSign (java.lang.String addr, int port, int timeout, ru.infocrypt.fpsu.enums.ProtocolType protocolType, ru.infocrypt.fpsu.TLSContext tlsContext) – инициализация соединения с сервером облачной подписи с указанием протокола связи.

```
public CloudSign(java.lang.String addr,
                 int port,
                 int timeout,
                 ru.infocrypt.fpsu.enums.ProtocolType protocolType,
                 ru.infocrypt.fpsu.TLSContext tlsContext)
          throws ru.infocrypt.fpsu.exception.FpsuException
```

Параметры:

```
      addr - адрес сервера

      port - порт

      timeout - таймаут сессии с сервером
```

protocolType - протокол связи ProtocolType

tlsContext - контекст TLS (в случае, если значение protocolType - TLS или TLS_GOST)

Исключения:
ru.infocrypt.fpsu.exception.FpsuException - возможные исключения

Методы

Модификатор и тип	Метод и описание
CSKeyPair	createKeyPair(PublicKeyParamSet paramSet, java.lang.String keyIdent) Создать облачную ключевую пару
byte[]	createSesKey(char paramSet, byte[] mac, byte[] encryptedKey, byte[] ephemeralPublicKey, byte[] ukm, byte[] maskedKey) Создать сессионный ключ расшифрования CMS для БиКрипт
ru.infocrypt.fpsu.MKeyInfo	getMkeyInfo(int n) Получить данные о загруженном на сервер ключе по его порядковому номеру N
MaskedKey	reMaskSignKey(java.lang.String newId, MaskedKey maskedKey) Перешифровать ключ ЭП на новый ТРК
byte[]	signDigest(MaskedKey maskedKey, byte[] digest) Подписать хеш

Метод createKeyPair

```
public CSKeyPair createKeyPair(PublicKeyParamSet paramSet,
                               java.lang.String keyIdent)
                        throws
ru.infocrypt.fpsu.exception.FpsuException
```

Назначение:
Создание облачной ключевой пары.

Параметры:
paramSet - набор параметров открытого ключа

keyIdent - идентификатор

Возвращаемое значение:
объект CSKeyPair (открытый ключ и закрытый, зашифрованный на мастер ключе)

Исключения:
`ru.infocrypt.fpsu.exception.FpsuException - возможные исключения`

Метод createSesKey

```
public byte[] createSesKey(char paramSet,
                           byte[] mac,
                           byte[] encryptedKey,
                           byte[] ephemeralPublicKey,
                           byte[] ukm,
                           byte[] maskedKey)
                    throws ru.infocrypt.fpsu.exception.FpsuException
```

Назначение:
Создание сессионного ключа расшифрования CMS для БиКрипт.

Параметры:
`paramSet - набор параметров открытого ключа`

`mac - МАК`

`encryptedKey - зашифрованный сессионный ключ`

`ephemeralPublicKey - открытый ключ`

`ukm - УКМ`

`maskedKey - маскированный закрытый ключ`

Возвращаемое значение:
`сессионный ключ для Бикрипта (64 байта)`

Исключения:
`ru.infocrypt.fpsu.exception.FpsuException - возможные исключения`

Метод getMkeyInfo

```
public ru.infocrypt.fpsu.MKeyInfo getMkeyInfo(int n)
                                        throws
ru.infocrypt.fpsu.exception.FpsuException
```

Назначение:
Получение данных о загруженном на сервер ключе по его порядковому номеру N.

Параметры:
`n - порядковый номер ключа`

Возвращаемое значение:
`экземпляр класса MKeyInfo`

Исключения:
`ru.infocrypt.fpsu.exception.FpsuException - возможные исключения`

Метод reMaskSignKey

```
public MaskedKey reMaskSignKey(java.lang.String newId,
                               MaskedKey maskedKey)
                        throws
ru.infocrypt.fpsu.exception.FpsuException
```

Перешифровать ключ ЭП на новый ТРК

Параметры:
newId - id ключа, на который происходит перешифрование

maskedKey - перешифровываемый маскированный ключ

Возвращаемое значение:
перешифрованный маскированный ключ

Исключения:
ru.infocrypt.fpsu.exception.FpsuException - возможные исключения

Метод signDigest

```
public byte[] signDigest(MaskedKey maskedKey,
                         byte[] digest)
                  throws ru.infocrypt.fpsu.exception.FpsuException
```

Назначение:
Подписание значения хеш-функции.

Параметры:
maskedKey - маскированный закрытый ключ

digest - хеш

Возвращаемое значение:
сырая подпись (64/128 байтов)

Исключения:
ru.infocrypt.fpsu.exception.FpsuException - возможные исключения

5.3.2 Класс CSKeyPair

```
java.lang.Object

         ru.infocrypt.fpsu. CSKeyPair
-----------------------------------------------------------------------

  public class CSKeyPair

      extends java.lang.Object
```

Описание

Ключевая пара облачной подписи.

Конструкторы

CSKeyPair(byte[] serialized) – конструирование объекта на базе сериализованной ключевой информации.

```
public CSKeyPair(byte[] serialized)
          throws ru.infocrypt.fpsu.exception.FpsuException
```

Параметры:
`serialized - сериализованные ключевые данные`

Исключения:
`ru.infocrypt.fpsu.exception.FpsuException - возможные исключения`

Методы

Модификатор и тип	Метод и описание
MaskedKey	getMaskedKey() Маскированный секретный ключ
byte[]	getPublicKey() Публичный ключ
byte[]	getSerializedMaskedKey()

Метод getMaskedKey

```
public MaskedKey getMaskedKey()
```

Назначение:
Получение структуры с маскированным ключом.

Возвращаемое значение:
MaskedKey структура с маскированным ключом.

Метод getPublicKey

```
public byte[] getPublicKey()
```

Назначение:
Получение открытого ключа.

Возвращаемое значение:
Открытый ключ в виде набора байтов.

Метод getSerializedMaskedKey

```
public byte[] getSerializedMaskedKey()
```

Назначение:
Получение сериализованной структуры с маскированным ключом

Возвращаемое значение:
набор байтов

5.3.3 Класс MaskedKey

```
java.lang.Object
        ru.infocrypt.fpsu.MaskedKey
-------------------------------------------------------------------------
```

```
public abstract class MaskedKey
extends java.lang.Object
```

Описание

Структура с маскированным ключом.

Конструкторы

MaskedKey()

```
public MaskedKey()
```

Методы

Модификатор и тип	Метод и описание
abstract java.lang.String	getIf10() Данные инициализационного вектора
abstract byte[]	getKey() Ключ
abstract java.lang.String	getKeyIdent() Идентификатор ключа
abstract PublicKeyParamSet	getPublicKeyParamSet() Набор параметров открытого ключа
abstract byte[]	getSerial() Серийный номер
abstract byte[]	getSerializedKey() Сериализованный объект
abstract byte[]	getVersion() Версия структуры

Метод getIf10

```
public abstract java.lang.String getIf10()
                                    throws
ru.infocrypt.fpsu.exception.FpsuException
```

Назначение:
Получение данных инициализационного вектора.

Возвращаемое значение:
строка

Исключения:
`ru.infocrypt.fpsu.exception.FpsuException - возможные исключения`

Метод getKey

```
public abstract byte[] getKey()
```

Назначение:
Получение ключа.

Возвращаемое значение:
набор байтов

Метод getKeyIdent

```
public abstract java.lang.String getKeyIdent()
                                         throws
ru.infocrypt.fpsu.exception.FpsuException
```

Назначение:
Получение идентификатора ключа.

Возвращаемое значение:
строка

Исключения:
`ru.infocrypt.fpsu.exception.FpsuException - возможные исключения`

Метод getPublicKeyParamSet

```
public abstract PublicKeyParamSet getPublicKeyParamSet()
```

Назначение:
Получение набора параметров открытого ключа

Возвращаемое значение:
`Объект PublicKeyParamSet`

Метод getSerial

```
public abstract byte[] getSerial()
```

Назначение:
Получение серийного номера.

Возвращаемое значение:
набор байтов

Метод getSerializedKey

```
public abstract byte[] getSerializedKey()
```

Назначение:
Получение сериализованного объекта.

Возвращаемое значение:
набор байтов

Метод getVersion

```
public abstract byte[] getVersion()
```

Назначение:
Получение номера версии структуры.

Возвращаемое значение:
набор байтов