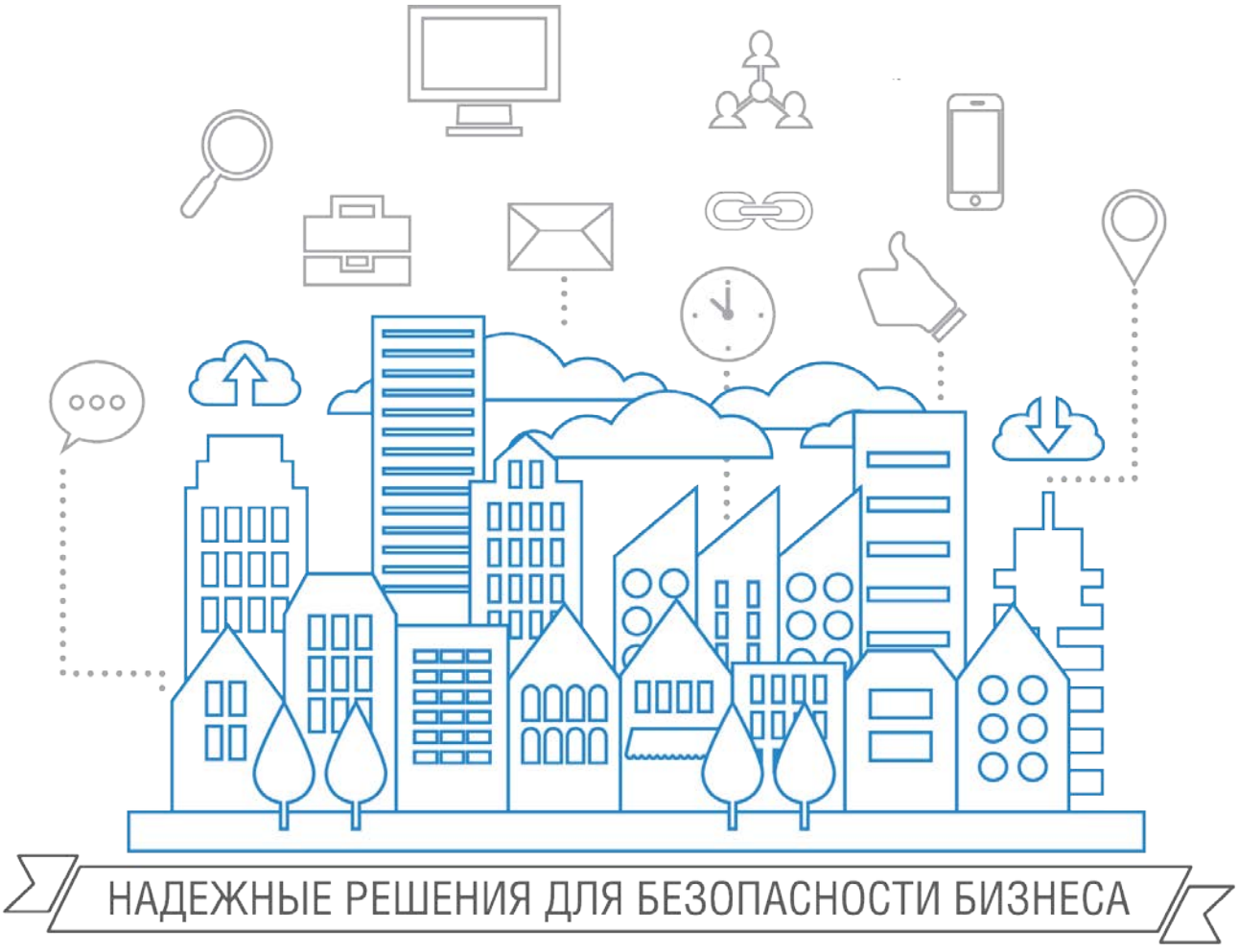
ГАЗИНФОРМСЕРВИС

198096, г. Санкт-Петербург, ул. Кронштадтская, д.10, лит. А, тел.: (812) 677-20-50, факс: (812) 677-20-51
Почтовый адрес: 198096, г. Санкт-Петербург, а/я 59, e-mail: resp@gaz-is.ru, www.gaz-is.ru
р/с 40702810800000001703 Ф-л Банка ГПБ (АО) в г. Санкт-Петербурге БИК 044030827,
к/с 30101810200000000827, ОКПО 72410666, ОГРН 1047833006099, ИНН/КПП 7838017968/783450001

Java-библиотека «BHCrypto» (Demo-версия)

Описание программного интерфейса



Санкт-Петербург, 2017

Аннотация

В документе приводится описание программного интерфейса Java-библиотеки «BHCrypto», с помощью которых вызываются основные функции криптографической платформы «Litoria Crypto Platform».

В разделе «Общие сведения» описано назначение Java-библиотеки.

В разделе «Описание программного интерфейса» приведена классификация функций, а также указаны сигнатуры функций, входные параметры, возвращаемые значения и примеры вызовов функций.

В разделе «Обработка исключительных ситуаций» описан способ обработки исключительных ситуаций.

В конце документа приведен список использованных терминов и сокращений.

Содержание

1 Общие сведения

Java-библиотека «BHCrypto» содержит программный интерфейс, предназначенный для вызова основных функций криптографической платформы «Litoria Crypto Platform».

Основные функции криптографической платформы «Litoria Crypto Platform»:

- установка сертификата в хранилище;
- создание усовершенствованной электронной подписи (УЭП);
- добавление УЭП;
- проверка УЭП;
- шифрование файла;
- расшифровывание файла.

2 Описание программных интерфейсов

При использовании программного интерфейса Java-библиотеки «BHCrypto» и наличии на АРМ криптографической платформы «Litoria Crypto Platform» можно реализовать клиентское программное обеспечение, в котором пользователю предоставляются все основные функции криптографической платформы «Litoria Crypto Platform».

Классификация функций представлена в таблице 2.1.

Таблица 2.1. Классификация функций

Тип функций	Функция	Назначение
Настройка доступа к прокси-серверу	setProxyUserNameAndPassword	Настройка имени пользователя и пароля для доступа к прокси-серверу
Работа с сертификатами	getAllCertFromStore	Получение коллекции сертификатов из хранилища
	installCert	Установка сертификата в хранилище
Работа с УЭП	signFile	Создание УЭП
	isSignatureFileDetached	Определение, является ли файл с УЭП отделенным
	addSignatureToAttachedFile	Добавление УЭП к файлу с присоединенной подписью
	addSignatureToDetachedFile	Добавление УЭП к файлу с отделенной подписью
	verifyAttachedFileSignature	Проверка присоединенной УЭП
	verifyDetachedFileSignature	Проверка отделенной УЭП
Шифрование	encryptFile	Шифрование файла
Расшифровывание	decryptFile	Расшифровывание файла

2.1 Настройка доступа к прокси-серверу

2.1.1 Функция setProxyUserNameAndPassword

Описание:

Настройка имени пользователя и пароля для доступа к прокси-серверу

Сигнатура функции:

public static void setProxyUserNameAndPassword (java.lang.String name, java.lang.String password) throws EDSException

Входные параметры:

name – имя пользователя
password – пароль пользователя

Пример вызова:

BHCrypto.setProxyUserNameAndPassword("domen\\user", "123456");

2.2 Работа с сертификатами

2.2.1 Функция getAllCertFromStore

Описание:

Получение коллекции сертификатов из хранилища

Сигнатура функции:

public static java.security.cert.X509Certificate[] getAllCertFromStore (java.lang.String storeName) throws EDSException

Входные параметры:

StoreName – имя хранилища

Примеры имен хранилищ:

- «MY» – хранилище личных сертификатов
- «CA» – сертификаты удостоверяющего центра
- «ROOT» – корневые сертификаты
- «AddressBook» – сертификаты других пользователей

Возвращаемое значение:

Коллекция сертификатов

Пример вызова:

```
X509Certificate[] certs= BHCrypto.getAllCertFromStore("AddressBook");
```

2.2.2 Функция installCert

Описание:

Установка сертификата в хранилище

Сигнатура функции:

public static void installCert(java.security.cert.X509Certificate certForInstall, java.lang.String storeName) throws EDSException

Входные параметры:

certForInstall – устанавливаемый сертификат

storeName – имя хранилища

Примеры имен хранилищ:

- «MY» – хранилище личных сертификатов
- «CA» – сертификаты удостоверяющего центра
- «ROOT» – корневые сертификаты
- «AddressBook» – сертификаты других пользователей

Пример вызова:

```
BufferedInputStream file_stream = new BufferedInputStream(new FileInputStream("C:\\CertFile.cer"));
CertificateFactory cf = CertificateFactory.getInstance("X.509");
X509Certificate cert = (X509Certificate)cf.generateCertificate(file_stream);
BHCrypto.installCert(cert, "AddressBook");
```

2.3 Работа с УЭП

2.3.1 Функция signFile

Описание:

Создание УЭП

Сигнатура функции:

public static void SignData(java.io.File inputFile, java.io.File outputFile, SigningOptions options) throws EDSException

Входные параметры:

inputFile – путь к подписываемому файлу

outputFile – путь к файлу, куда будет помещена подпись

options – параметры подписи

Пример вызова:

```
X509Certificate[] myCerts = BHCrypto.getAllCertFromStore("MY");
SigningOptions options = new SigningOptions
        (myCerts [0]
        ,false//создать присоединенную подпись
        ,"http://tsp.gaz-is.ru/tsp/tsp.srf"
        ,"Комментарий");
BHCrypto.signFile(
        new File("C:\\test.txt")
        ,new File("C:\\test.txt.p7s")
        ,options);
```

2.3.2 Функция isSignatureFileDetached

Описание:

Определение, является ли файл с УЭП отделенным

Сигнатура функции:

public static boolean isSignatureFileDetached (java.io.File signature_file) throws EDSException

Входные параметры:

signature_file – путь к файлу

Возвращаемое значение:

true – если файл с УЭП отделенный

Пример вызова:

```
boolean detached = BHCrypto.isSignatureFileDetached(new File("C:\\test.txt.p7s"));
```

2.3.3 Функция addSignatureToAttachedFile

Описание:

Добавление УЭП к файлу с присоединенной подписью

Сигнатура функции:

public static void addSignatureToAttachedFile (java.io.File inputFile, java.io.File outputFile, AddingSignatureOptions options) throws EDSException

Входные параметры:

inputFile – путь к файлу с подписью
outputFile – путь к файлу, куда будут помещены результирующие подписи
options – параметры подписи

Пример вызова:

```
X509Certificate[] myCerts = BHCrypto.getAllCertFromStore("MY");
AddingSignatureOptions options = new AddingSignatureOptions(
        myCerts [0]
        ,"http://tsp.gaz-is.ru/tsp/tsp.srf"
        ,"Комментарий");
BHCrypto.addSignatureToAttachedFile(
        new File("C:\\test.txt.p7s")
        ,new File("C:\\test_out.txt.p7s")
        ,options);
```

2.3.4 Функция addSignatureToDetachedFile

Описание:

Добавление УЭП к файлу с отделенной подписью

Сигнатура функции:

public static void addSignatureToDetachedFile (java.io.File signatureFile, java.io.File dataFile, java.io.File outputFile, AddingSignatureOptions options) throws EDSException

Входные параметры:

signatureFile – путь к файлу с отделенной подписью
dataFile – путь к файлу с данными
outputFile – путь к файлу, куда будут помещены результирующие подписи
options – параметры подписи

Пример вызова:

```
X509Certificate[] myCerts = BHCrypto.getAllCertFromStore("MY");
AddingSignatureOptions options = new AddingSignatureOptions(
        myCerts [0]
        ,"http://tsp.gaz-is.ru/tsp/tsp.srf"
        ,"Комментарий");
BHCrypto.addSignatureToDetachedFile(
        new File("C:\\test.txt.p7s")
        ,new File("C:\\test.txt")
        ,new File("C:\\test_out.txt.p7s")
        ,options);
```

2.3.5 Функция verifyAttachedFileSignature

Описание:

Проверка присоединенной УЭП

Сигнатура функции:

public static SignatureInfo[] verifyAttachedFileSignature (java.io.File file, VerifyingAttachedSignatureOptions options) throws EDSException

Входные параметры:

file – подписанный файл с присоединенной подписью
options – параметры проверки

Возвращаемое значение:

Результаты проверки

Пример вызова:

```
VerifyingAttachedSignatureOptions options =
VerifyingAttachedSignatureOptions.createWithExtractDataToFile(
        new File("C:\\test.txt"));
SignatureInfo[] infos = BHCrypto.verifyAttachedFileSignature(
        new File("C:\\test.txt.p7s")
        ,options);
```

2.3.6 Функция verifyDetachedFileSignature

Описание:
Проверка отделенной УЭП

Сигнатура функции:
public static SignatureInfo[] verifyDetachedFileSignature (java.io.File detachedSignature, java.io.File dataFile) throws EDSException

Входные параметры:
detachedSignature – файл с отделенной подписью
dataFile – файл с данными для проверки подписи

Возвращаемое значение:
Результаты проверки

Пример вызова:
```
SignatureInfo[] infos = BHCrypto.verifyDetachedFileSignature(
    new File("C:\\test.txt.p7s")
    ,new File("C:\\test.txt"));
```

2.4 Шифрование файла

2.4.1 Функция encryptFile

Описание:
Шифрование файла

Сигнатура функции:
public static void encryptFile (java.io.File inputFile, java.io.File outputFile, java.security.cert.X509Certificate[] certificates) throws EDSException

Входные параметры:
inputFile – путь к файлу, который будет зашифрован
outputFile – путь к файлу, куда будет помещен зашифрованный файл
certificates – сертификаты получателей

Пример вызова:
```
X509Certificate[] certs = BHCrypto.getAllCertFromStore("AddressBook");
BHCrypto.encryptFile(
    new File("C:\\test.txt")
    ,new File("C:\\test.txt.p7m")
    ,certs);
```

2.5 Расшифрование файла

2.5.1 Функция decryptFile

Описание:
Расшифровывание файла

Сигнатура функции:
public static java.security.cert.X509Certificate decryptFile (java.io.File inputFile, java.io.File outputFile) throws EDSException

Входные параметры:

inputFile – путь к зашифрованному файлу

outputFile – путь к файлу, куда будут помещены расшифрованные данные

Возвращаемое значение:

Сертификат, с помощью которого выполнено расшифровывание

Пример вызова:

```
X509Certificate cert = BHCrypto.decryptFile(
        new File("C:\\test.txt.p7m")
        ,new File("C:\\test.txt"));
```

3 Обработка исключительных ситуаций

Функции обработки исключений помогают обрабатывать любые непредвиденные или исключительные ситуации, происходящие при выполнении основных операций. Все исключения, генерируемые классом BHCrypto, являются исключениями типа EDSException. Для обработки исключений используются ключевые слова try и catch. При обработке исключений может быть полезно просматривать цепочку исключений, ставших причиной текущего (метод getCause()).

Пример обработки исключения при выполнении операции создания УЭП:

```
try {
        BHCrypto.signFile(inputFile, outputFile, options);
        }
        catch (EDSException e) {
                        e.printStackTrace();
        }
```

Термины и сокращения

CA	–	Certifying Authority (Сертифицирующая организация)
TSP	–	Time-Stamp Protocol (Протокол штампов времени)
УЭП	–	Усовершенствованная Электронная Подпись