

# TRUSTED API 0.1

Модуль trusted-crypto  
для nodejs

СОСТАВ ПРОДУКТА-----	3
ПОРЯДОК УСТАНОВКИ-----	3
ОПРЕДЕЛЕНИЯ-----	4
CMS -----	5
КЛАСС SIGNEDDATA-----	5
КЛАСС SIGNER -----	8
КЛАСС SIGNERCOLLECTION -----	9
КЛАСС SIGNERATRIBUTECOLLECTION -----	10
PKI-----	11
ОПРЕДЕЛЕНИЯ-----	11
КЛАСС ALGORITHM-----	11
КЛАСС ATTRIBUTE -----	13
КЛАСС ATTRIBUTEVALUECOLLECTION-----	14
КЛАСС CERTIFICATE-----	15
КЛАСС CERTIFICATIONREQUEST -----	18
КЛАСС CERTIFICATIONREQUESTINFO-----	19
КЛАСС CERTIFICATIONCOLLECTION -----	20
КЛАСС CERTSTORE-----	21
КЛАСС CHAIN -----	23
КЛАСС CIPHER -----	24
КЛАСС CRL -----	25
КЛАСС CRLCOLLECTION-----	28
КЛАСС CSR -----	29
КЛАСС KEY -----	30
КЛАСС OID -----	32
КЛАСС PKCS12-----	33
КЛАСС PROVIDERSYSTEM -----	35
КЛАСС REVOCATION-----	36
PKISTORE -----	37
КЛАСС CASHJSON-----	37
КЛАСС FILTER -----	38
КЛАСС PKIITEM-----	39
КЛАСС PKISTORE -----	40

## **СОСТАВ ПРОДУКТА**

Продукт trusted-crypto представляет собой внешний модуль, который работает с приложением Nodejs и предоставляет расширенные возможности по работе с цифровыми сертификатами, подписью, шифрованием данных, работе с хранилищами ключей и сертификатов.

## **ПОРЯДОК УСТАНОВКИ**

Для сборки модуля необходимо сделать предварительную установку внешних модулей для Nodejs. Установка выполняется запуском последовательности консольных команд:

```
> npm install -g typescript  
> npm install -g tsd  
> npm install -g mocha
```

Сборка модуля при условии установки MS Visual Studio 2013 на Windows, g++ и библиотек разработки на Ubuntu и XCode на MacOS, выполняется с помощью консольной команды:

```
> npm install
```

Запуск тестов для проверки основных функций модуля trusted-crypto выполняется командой:

```
> npm test
```

## ОПРЕДЕЛЕНИЯ

```
CryptoMethod {  
    SYMMETRIC = 0,  
    ASYMMETRIC = 1,  
}
```

```
DataFormat {  
    DER = 0,  
    PEM = 1,  
}
```

```
PublicExponent {  
    RSA_3 = 0,  
    RSA_F4 = 1,  
}
```

CMS

КЛАСС SIGNEDDATA

INTERFACES -----

НАЗВАНИЕ	ПАРАМЕТРЫ	ОПИСАНИЕ
ISIGNEDDATACONTENT	type: SignedDataContentType data: string   Buffer	

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
CONTENT	ISignedDataContent	Возвращает или задает содержимое подписи
POLICIES	Array<string>	Возвращает или задает политики подписи

METHODS -----

МЕТОД	ОПИСАНИЕ
ISDETACHED()	Возвращает true если подпись открепленная
CERTIFICATES()	Возвращает коллекцию сертификатов
SIGNERS()	Возвращает коллекцию подписчиков
LOAD(STRING, DATAFORMAT)	Выполняет чтение подписи из файла
IMPORT(BUFFER, DATAFORMAT)	Выполняет чтение подписи из памяти
SAVE(STRING, DATAFORMAT)	Выполняет сохранение подписи в файл
EXPORT(DATAFORMAT)	Выполняет сохранение подписи в память
CREATESIGNER(CERTIFICATE, KEY, STRING)	Создает нового подписчика
VERIFY(CERTIFICATECOLLECTION)	Выполняет проверку подписи

isDetached() : boolean

Возвращает true если подпись открепленная

certificates(index: number): Certificate

Возвращает сертификат по индексу

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

certificates(): CertificateCollection

Возвращает коллекцию сертификатов CertificateCollection

signers(index: number): Signer

Возвращает подписчика по индексу

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

signers(): SignerCollection

Возвращает коллекцию подписчиков

----- ПРИМЕР -----

```

var trusted = require("trusted-crypto");
var cms = trusted.cms.SignedData.load("sigdoc.sig", trusted.DataFormat.PEM);
console.log(cms.isDetached()); // true
var signers = cms.signers();
for (var i = 0; i < signers.length; i++){
    var signer = signers.items(i);
    console.log("Signer digest name:", signer.digestAlgorithm.name); // Signer digest name: sha1
}

```

load(filename: string, format?: DataFormat): void

*Чтение подписи из файла*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

static load(filename: string, format?: DataFormat): SignedData

*Чтение подписи из памяти*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

import(buffer: Buffer, format?: DataFormat): void

*Чтение подписи из памяти*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

static import(buffer: Buffer, format?: DataFormat): SignedData

*Чтение подписи из памяти*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

export(format?: DataFormat): Buffer

*Сохранение подписи в память*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

save(filename: string, format?: DataFormat): void

*Сохранение подписи в файл*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу

<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER
---------------	------------	--

createSigner(cert: Certificate, key: Key, digestName: string): Signer

*Создание нового подписчика*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>CERT</b>	Certificate	Сертификат подписчика
<b>KEY</b>	Key	Формат данных. Опционально. По умолчанию DER
<b>DIGESTNAME</b>	string	Имя хэш алгоритма

verify(certs?: CertificateCollection): boolean

*Проверка подписи*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>CERTS</b>	CertificateCollection	Коллекция дополнительных сертификатов

sign(): void

*Создание подписи*

КЛАСС SIGNER

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
CERTIFICATE	Certificate	Задаёт или возвращает сертификат подписчика
DIGESTALGORITHM	Algorithm	Возвращает хэш алгоритм проверки содержимого

METHODS -----

МЕТОД	ОПИСАНИЕ
SIGNEDATTRIBUTES(SIGNERATTRIBUTECOLLECTION)	Возвращает коллекцию подписанных атрибутов
UNSIGNEDATTRIBUTES(SIGNERATTRIBUTECOLLECTION)	Возвращает коллекцию неподписанных атрибутов

signedAttributes(): SignerAttributeCollection

*возвращает коллекцию подписанных атрибутов*

signedAttributes(index: number): Attribute

*возвращает атрибут из коллекции по заданному индексу*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

unsignedAttributes(): SignerAttributeCollection

*возвращает коллекцию неподписанных подписанных атрибутов*

unsignedAttributes(index: number): Attribute

*возвращает атрибут из коллекции по заданному индексу*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции



КЛАСС SIGNERCOLLECTION

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
LENGTH	number	Возвращает размер коллекции

METHODS -----

МЕТОД	ОПИСАНИЕ
ITEMS	Возвращает элемент из коллекции по заданному индексу

items(index: number): Signer

*Возвращает элемент из коллекции по заданному индексу*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

КЛАСС SIGNERATRIBUTECOLLECTION

Представление коллекции атрибутов подписчика

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
LENGTH	number	Возвращает размер коллекции

METHODS -----

МЕТОД	ОПИСАНИЕ
ITEMS	Возвращает элемент из коллекции по заданному индексу
PUSH	Добавляет новый элемент в коллекцию
REMOVEAT	Удаляет элемент из коллекции по заданному индексу

push(attr: Attribute): void

добавляет новый элемент в коллекцию

ПАРАМЕТР	ТИП	ОПИСАНИЕ
ATTR	Attribute	Новый элемент коллекции

removeAt(index: number): void

удаляет элемент из коллекции по заданному индексу

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

items(index: number): Attribute

возвращает элемент коллекции по заданному индексу

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

## PKI

### ОПРЕДЕЛЕНИЯ

```
KeyUsageFlags {  
    DigitalSignature = 128,  
    NonRepudiation = 64,  
    KeyEncipherment = 32,  
    DataEncipherment = 16,  
    KeyAgreement = 8,  
    KeyCertSign = 4,  
    CrlSign = 2,  
    EncipherOnly = 1,  
    DecipherOnly = 32768,  
}
```

### КЛАСС ALGORITHM

*Представление X509\_ALGOR*

### CONSTRUCTORS

КОНСТРУКТОР	ОПИСАНИЕ
<b>CONSTRUCTOR()</b>	Конструктор по умолчанию
<b>CONSTRUCTOR(HANDLE: NATIVE.PKI.ALGORITHM)</b>	Конструктор с указанием алгоритма – через экземпляр объекта
<b>CONSTRUCTOR(NAME: STRING)</b>	Конструктор с указанием строкового названия алгоритма

### PROPERTIES

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
<b>NAME</b>	string	Возвращает название алгоритма
<b>TYPEID</b>	Oid	Возвращает идентификатор алгоритма

### METHODS

МЕТОД	ОПИСАНИЕ
<b>DUPLICATE</b>	Возвращает копию алгоритма
<b>ISDIGEST</b>	Возвращает true если алгоритм предназначен для вычисления хэш

constructor()

*Конструктор для создания экземпляра объекта по умолчанию.*

constructor(handle: native.PKI.Algorithm)

*Конструктор для создания экземпляра объекта на основе существующего экземпляра (Конструктор копированием.)*

constructor(name: string)

*Конструктор для создания экземпляра объекта по указанному имени алгоритма.*

duplicate(): Algorithm

*Возвращает копию алгоритма*

isDigest(): boolean

*Возвращает true если алгоритм предназначен для вычисления хэша.*

КЛАСС ATTRIBUTE

*Представление X509\_ATTR*

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR(HANDLE: NATIVE.PKI.ATTRIBUTE)	Конструктор с указанием атрибута – через экземпляр объекта

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
ASNTYPE	number	Задаёт ASN1 тип атрибута
TYPEID	Oid	Задаёт идентификатор атрибута

METHODS -----

МЕТОД	ОПИСАНИЕ
DUPLICATE	Возвращает копию алгоритма
EXPORT	Возвращает атрибут в DER кодировке
VALUES	Возвращает коллекцию значений атрибута. Значения представляются в DER формате

duplicate(): Attribute

*Возвращает копию атрибута*

export(): Buffer

*Возвращает атрибут в DER кодировке*

values(): AttributeValueCollection

*Возвращает коллекцию значений атрибута. Значения представляются в DER формате*

values(index: number): Buffer

*возвращает коллекцию значений атрибута. Значения представляются в DER формате*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

КЛАСС ATTRIBUTEVALUECOLLECTION

Представление коллекции X509\_ATTR

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR(HANDLE: NATIVE.PKI.ATTRIBUTEVALUECOLLECTION)	Конструктор с указанием коллекции атрибутов – через экземпляр объекта

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
LENGTH	number	Возвращает количество элементов в коллекции

METHODS -----

МЕТОД	ОПИСАНИЕ
PUSH	Добавляет новый элемент в коллекцию
POP	Удаляет последний элемент из коллекции
REMOVEAT	Удаляет элемент коллекции по заданному индексу
ITEMS	Возвращает элемент из коллекции по заданному индексу

push(val: Buffer): void

добавляет новый элемент в коллекцию

ПАРАМЕТР	ТИП	ОПИСАНИЕ
VAL	Buffer	Новое значение коллекции

pop(): void

удаляет последний элемент из коллекции

removeAt(index: number): void

удаляет элемент коллекции по заданному индексу

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

items(index: number): Buffer

возвращает элемент из коллекции по заданному индексу

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

КЛАСС CERTIFICATE

Представление 'X509' сертификата

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR()	Конструктор по умолчанию
CONSTRUCTOR(HANDLE: NATIVE.PKI.CERTIFICATE)	Конструктор с указанием сертификата – через экземпляр объекта

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
VERSION	number	Возвращает серийный номер сертификата
SERIALNUMBER	string	Возвращает серийный номер сертификата
TYPE	number	Возвращает тип сертификата
KEYUSAGE	number	Возвращает набор флагов KeyUsageFlags, задающих назначение ключа сертификата
ISSUERFRIENDLYNAME	string	Возвращает пользовательское имя издателя сертификата
ISSUERNNAME	string	Возвращает полное имя издателя сертификата
SUBJECTFRIENDLYNAME	string	Возвращает пользовательское имя владельца сертификата
SUBJECTNAME	string	Возвращает полное имя владельца сертификата
NOTBEFORE	Date	Возвращает время с которого сертификат считается действительным
NOTAFTER	Date	Возвращает время до которого сертификат считается действительным
THUMBPRINT	string	Возвращает алгоритм подписи
SIGNATUREALGORITHM	string	Возвращает алгоритм подписи
ORGANIZATIONNAME	string	Возвращает название организации

METHODS -----

МЕТОД	ОПИСАНИЕ
COMPARE	Сравнение сертификатов
EQUALS	Сравнение сертификатов
HASH	Вычисление значения хэша сертификата
DUPLICATE	Создает копию сертификата
LOAD	Чтение сертификата из файла
IMPORT	Чтение сертификата из памяти
EXPORT	Сохранение сертификата в память
SAVE	Сохранение сертификата в файл

compare(cert: Certificate): number

сравнение сертификатов

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CERT	Certificate	Сертификат для сравнения

equals(cert: Certificate): boolean

сравнение сертификатов

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CERT	Certificate	Сертификат для сравнения

hash(algorithm?: string): String

*вычисление значения хэша сертификата*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
ALGORITHM	string	Имя хэш алгоритма. Опционально. По умолчанию sha1

duplicate(): Certificate

*Создает копию сертификата*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
ALGORITHM	string	Имя хэш алгоритма. Опционально. По умолчанию sha1

load(filename: string, format?: DataFormat): void

*чтение сертификата из файла*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

static load(filename: string, format?: DataFormat): Certificate

*чтение сертификата из файла*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

import(buffer: Buffer, format?: DataFormat): void

*чтение сертификата из памяти*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
BUFFER	Buffer	Буфер памяти
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

static import(buffer: Buffer, format?: DataFormat): Certificate

*чтение сертификата из памяти*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
BUFFER	Buffer	Буфер памяти
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

export(format?: DataFormat): Buffer

*сохранение сертификата в память*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER



save(filename: string, format?: DataFormat): void

*сохранение сертификата в файл*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

## КЛАСС CERTIFICATIONREQUEST

Класс для создания ASN.1 структуры *CertificationRequest*, которая представлена как:  
*CertificationRequest ::= SEQUENCE { certificationRequestInfo CertificationRequestInfo, signatureAlgorithm  
AlgorithmIdentifier{{ SignatureAlgorithms }}, signature BIT STRING }*

### CONSTRUCTORS

КОНСТРУКТОР	ОПИСАНИЕ
<b>CONSTRUCTOR()</b>	Конструктор по умолчанию
<b>CONSTRUCTOR(HANDLE: NATIVE.PKI.CERTIFICATIONREQUEST)</b>	Конструктор копированием экземпляра объекта <i>CertificationRequest</i>

### PROPERTIES

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
<b>PEMSTRING</b>	Buffer	Получение запроса на сертификат

### METHODS

МЕТОД	ОПИСАНИЕ
<b>LOAD</b>	Чтение запроса из файла
<b>SIGN</b>	Подпись запроса
<b>VERIFY</b>	Проверка запроса на сертификат

*load(filename: string, format?: DataFormat): void*

*чтение запроса из файла*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>FILENAME</b>	string	Путь к файлу
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

*static load(filename: string, format?: DataFormat): CertificationRequest*

*чтение запроса из файла*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>FILENAME</b>	string	Путь к файлу
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

*sign(key: Key): void*

*подпись запроса*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>KEY</b>	Key	Ключевая пара

*verify(): boolean*

*проверка подписи запроса на сертификат*

КЛАСС CERTIFICATIONREQUESTINFO

Класс для создания ASN.1 структуры CertificationRequestInfo, которая представлена как:  
CertificationRequestInfo ::= SEQUENCE { version INTEGER { v1(0) } (v1,...), subject Name, subjectPKInfo  
SubjectPublicKeyInfo{{ PKInfoAlgorithms }}, attributes [0] Attributes{{ CRIAttributes }} }

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR()	Конструктор по умолчанию
CONSTRUCTOR(HANDLE: NATIVE.PKI.CERTIFICATIONREQUESTINFO)	Конструктор копированием экземпляра объекта CertificationRequestInfo

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
SUBJECT	String	Возвращает заголовок
PUBKEY	Key	Возвращает публичный ключ
VERSION	number	Возвращает номер версии

КЛАСС CERTIFICATIONCOLLECTION

Представление коллекции `X509` сертификатов

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
<b>CONSTRUCTOR()</b>	Конструктор по умолчанию
<b>CONSTRUCTOR(HANDLE: NATIVE.PKI.CERTIFICATECOLLECTION)</b>	Конструктор копированием экземпляра объекта CertificateCollection

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
<b>LENGTH</b>	number	Возвращает размер коллекции

METHODS -----

МЕТОД	ОПИСАНИЕ
<b>ITEMS</b>	Возвращает элемент из коллекции по заданному индексу
<b>PUSH</b>	Добавляет новый элемент в коллекцию
<b>POP</b>	Удаляет последний элемент их коллекции
<b>REMOVEAT</b>	Удаляет элемент из коллекции по заданному индексу

items(index: number): Certificate

Возвращает элемент из коллекции по заданному индексу

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>INDEX</b>	number	Индекс элемента в коллекции

push(cert: Certificate): void

Добавляет новый элемент в коллекцию

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>CERT</b>	Certificate	Элемент для добавления в коллекцию

pop(): void

Удаляет последний элемент их коллекции

removeAt(index: number): void

Удаляет элемент из коллекции по заданному индексу

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>INDEX</b>	number	Индекс элемента в коллекции

## КЛАСС CERTSTORE

Класс для организации стека хранилищ сертификатов и ключей. Организуется кэш (JSON) для формирования описателей объектов хранения.

### CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
<b>CONSTRUCTOR()</b>	Конструктор по умолчанию
<b>CONSTRUCTOR(HANDLE: NATIVE.PKI.CERTSTORE)</b>	Конструктор копированием экземпляра объекта CertStore

### PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
<b>LISTCERTSTORE</b>	string	Возвращает список хранилищ сертификатов

### METHODS -----

МЕТОД	ОПИСАНИЕ
<b>ADDCERTSTORE</b>	Добавляет новое хранилище сертификатов в список хранилищ
<b>REMOVECERTSTORE</b>	Удаляет существующее хранилище сертификатов из списка
<b>CREATECACHE</b>	Создает кэш
<b>ADDCACHESECTION</b>	Создает кэш для указанного хранилища
<b>GETPRVTYPEPRESENT</b>	Проверяет присутствие провайдера в списке

addCertStore(pvdType: string, pvdURI: string): void

*Добавление хранилища в стек*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>PVDTYPE</b>	string	Тип провайдера хранилища
<b>PVDURI</b>	string	Путь к месторасположению хранилища

removeCertStore(pvdType: string): void

*Удаление выбранного хранилища из стека*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>PVDTYPE</b>	string	Тип провайдера хранилища

createCache(cacheURI: string): void

*Создание кэша стека хранилищ (JSON)*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>CACHEURI</b>	string	Путь к файлу кэша

addCacheSection(cacheURI: string, pvdType: string): void

*Добавление в кэш стека хранилищ (JSON) раздела для хранения описания по провайдеру указанного типа*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>CACHEURI</b>	string	Путь к файлу кэша
<b>PVDTYPE</b>	string	Тип провайдера хранилища

getPrvTypePresent(pvdType: string): boolean

*Возврат true если в стеке содержится хранилище указанного типа, и false в противном случае.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
PVDTYPE	string	Тип провайдера хранилища

## КЛАСС CHAIN

Класс для построения цепочки сертификатов

### CONSTRUCTORS

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR()	Конструктор по умолчанию

### METHODS

МЕТОД	ОПИСАНИЕ
BUILDCHAIN	Выполняет построение цепочки сертификатов
VERIFYCHAIN	Возвращает статус проверки сертификата относительно цепочки

buildChain(cert: Certificate, certs: CertificateCollection): CertificateCollection

Возвращает коллекцию сертификатов *CertificateCollection*, участвующих при построении цепочки. В качестве параметра функции указывается сертификат *cert*, для которого производится построение цепочки, провайдер хранилища сертификатов (по умолчанию «*pvdSystem*» – системный провайдер). В перегруженной функции указывается сертификат *cert*, для которого производится построение цепочки, коллекция сертификатов *certs* относительно которой происходит построение цепочки.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CERT	Certificate	Сертификат, для которого производится построение цепочки
CERTS	CertificateCollection	Коллекция сертификатов цепочки

verifyChain(chain: CertificateCollection, crls: CrlCollection): boolean

Выполняет проверку относительно цепочки сертификатов. Результат возвращается в виде числового значения – 0 – цепочка не проверена, 1 – цепочка проверена. В случае возврата значения 0 в стеке ошибок *error\_stack* содержится информация о причинах невозможности построения/проверки цепочки сертификатов.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CHAIN	CertificateCollection	Коллекция сертификатов цепочки
CRLS	CrlCollection	Коллекция списков отзыва сертификатов

КЛАСС CIPHER

Представление шифрования данных

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR(CIPHERNAME: STRING)	Конструктор с параметров в виде наименования алгоритма

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
CRYPTOMETHOD	CryptoMethod	Возвращает криптографический метод
RECIPIENTSCERTS	CertificateCollection	Возвращает коллекцию сертификатов получателей
PRIVKEY	Key	Возвращает закрытый ключ
RECIPIENTCERT	Certificate	Возвращает сертификат получателя
PASSWORD	string	Возвращает пароль для шифрования
DIGEST	string	
RIV	Buffer	
IV	string	Возвращает вектор инициализации
RKEY	Buffer	
KEY	string	
RSALT	Buffer	
SALT	string	Возвращает псевдослучайную вставку
ALGORITHM	String	Возвращает алгоритм
MODE	String	
DGST	String	

METHODS -----

МЕТОД	ОПИСАНИЕ
ENCRYPT	Реализация шифрования данных
DECRYPT	Реализация расшифрования данных

encrypt(filenameSource: string, filenameEnc: string, format?: DataFormat): void

Реализация шифрования данных.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAMESOURCE	string	Исходное сообщение
FILENAMEENC	String	Зашифрованное сообщение
FORMAT	DataFormat	Формат представления. По умолчанию - DER

decrypt(filenameEnc: string, filenameDec: string, format?: DataFormat): void

Реализация расшифрования данных.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAMEENC	string	Шифрованное сообщение
FILENAMEDEC	string	Расшифрованное сообщение
FORMAT	DataFormat	Формат представления. По умолчанию - DER



## КЛАСС CRL

Представление X509\_CRL списков отзыва сертификата

### CONSTRUCTORS

КОНСТРУКТОР	ОПИСАНИЕ
<b>CONSTRUCTOR()</b>	Конструктор по умолчанию
<b>CONSTRUCTOR(HANDLE: NATIVE.PKI.CRL)</b>	Конструктор копированием экземпляра объекта CRL

### PROPERTIES

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
<b>ENCODED</b>	Buffer	Возвращает ASN.1 объект CRL в DER кодировке
<b>SIGNATURE</b>	Buffer	Возвращает значение подписи
<b>VERSION</b>	number	Возвращает версию
<b>ISSUENAME</b>	string	Возвращает имя издателя
<b>ISSUERFRIENDLYNAME</b>	string	Возвращает пользовательское имя издателя сертификата
<b>LASTUPDATE</b>	Date	Возвращает дату последнего обновления
<b>NEXTUPDATE</b>	Date	Возвращает дату следующего обновления
<b>THUMBPRINT</b>	string	Возвращает отпечаток (SHA1)
<b>SIGNALGNAME</b>	string	Возвращает имя алгоритма подписи CRL
<b>SIGNALGSHORTNAME</b>	string	Возвращает короткое имя алгоритма подписи CRL
<b>SIGNALGOID</b>	string	Возвращает OID алгоритма подписи CRL

### METHODS

МЕТОД	ОПИСАНИЕ
<b>GETREVOKEDCERTIFICATECERT</b>	возвращает список отозванных сертификатов с применением фильтра (отбор осуществляется сравнением с указанным сертификатов в качестве параметра)
<b>GETREVOKEDCERTIFICATESERIAL</b>	возвращает список отозванных сертификатов с применением фильтра (отбор осуществляется сравнением указанного серийного номера сертификата).
<b>LOAD</b>	чтение структуры из файла
<b>IMPORT</b>	чтение структуры из памяти
<b>EXPORT</b>	сохранение структуры в память
<b>SAVE</b>	сохранение структуры в файл
<b>COMPARE</b>	равнение crl
<b>EQUALS</b>	сравнение
<b>HASH</b>	возвращает хэш структуры по заданному алгоритму
<b>DUPLICATE</b>	создает копию элемента

`getRevokedCertificateCert(ser: Certificate): native.PKI.RevokedCertificate`

Функция возвращает список отозванных сертификатов с применением фильтра (отбор осуществляется сравнением с указанным сертификатов в качестве параметра).

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>CER</b>	Certificate	Сертификат для организации выборки

`getRevokedCertificateSerial(serial: string): native.PKI.RevokedCertificate`

Функция возвращает список отозванных сертификатов с применением фильтра (отбор осуществляется сравнением указанного серийного номера сертификата).

ПАРАМЕТР	ТИП	ОПИСАНИЕ
----------	-----	----------

<b>SERIAL</b>	string	Серийный номер сертификата для организации выборки
---------------	--------	--

load(filename: string, format?: DataFormat): void

*чтение структуры из файла*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>FILENAME</b>	string	Путь к файлу
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

static load(filename: string, format?: DataFormat): Crl

*чтение структуры из файла*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>FILENAME</b>	string	Путь к файлу
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

import(buffer: Buffer, format?: DataFormat): void

*чтение структуры из памяти*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>BUFFER</b>	Buffer	Буфер с данными в памяти
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

static import(buffer: Buffer, format?: DataFormat): Crl

*чтение структуры из памяти*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>BUFFER</b>	Buffer	Буфер с данными в памяти
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

export(format?: DataFormat): Buffer

*сохранение структуры в файл*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

save(filename: string, dataFormat?: DataFormat): void

*сохранение структуры в файл*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>FILENAME</b>	String	Путь к файлу
<b>DATAFORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

compare(crl: Crl): number

*сравнение crl*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>CRL</b>	Crl	Экземпляр объекта Crl

`equals(crl: Crl): boolean`

*сравнение*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CRL	Crl	Экземпляр объекта Crl

`hash(algorithm?: string): String`

*возвращает хэш структуры по заданному алгоритму*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
ALGORITHM	string	Наименование алгоритма

`duplicate(): Crl`

*создает копию элемента*

КЛАСС CRLCollection

Представление коллекции списков отзыва сертификатов

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR()	Конструктор по умолчанию
CONSTRUCTOR(HANDLE: NATIVE.PKI.CRLCollection)	Конструктор копированием экземпляра объекта CrlCollection

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
LENGTH	number	Размер коллекции (число сертификатов)

METHODS -----

МЕТОД	ОПИСАНИЕ
ITEMS	Возвращает элемент коллекции списков отзыва
PUSH	Добавляет элементы в коллекцию
POP	
REMOVEAT	Удаляет элемент из коллекции

items(index: number): Crl

Возвращает элемент коллекции списков отзыва.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

push(crl: Crl): void

Добавляет элементы в коллекцию.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CRL	Crl	Экземпляр объекта списка отзыва

pop(): void

removeAt(index: number): void

Удаляет элемент из коллекции.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
INDEX	number	Индекс элемента в коллекции

КЛАСС CSR

Класс для создания запроса на сертификат.

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR(NAME: STRING, KEY: KEY, DIGEST: STRING)	

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
ENCODED	Buffer	Получение сформированного запроса на сертификат в Hex представлении

METHODS -----

МЕТОД	ОПИСАНИЕ
SAVE	Сохранение запроса на сертификат

save(filename: string, dataFormat?: DataFormat): void

Сохранение запроса на сертификат

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Полный путь к файлу
DATAFORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER

КЛАСС KEY

Представление ключевой пары. В составе класса объявлена структура

KEYPAIR{KeyAlgorithm algorithm; EVP\_PKEY pkey;}

enum KeySize{1024,2048,4096}

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR()	Конструктор по умолчанию
CONSTRUCTOR(HANDLE: NATIVE.PKI.KEY)	Конструктор копированием экземпляра объекта Key

METHODS -----

МЕТОД	ОПИСАНИЕ
GENERATE	Генерация ключевой пары
READPRIVATEKEY	Чтение приватного ключа из файла
WRITEPRIVATEKEY	Запись приватного ключа в файл
READPUBLICKEY	Чтение открытого ключа из файла
WRITEPUBLICKEY	Запись открытого ключа в файл
COMPARE	Сравнение ключей

generate(format: DataFormat, pubExp: PublicExponent, keySize: number, password: string): Key

Генерация ключевой пары

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER
PUBEXP	PublicExponent	
KEYSIZE	number	Размер ключа
PASSWORD	string	Пароль для доступа к зашифрованному контенту

readPrivateKey(filename: string, format: DataFormat, password: string): Key

Чтение приватного ключа из файла

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER
PASSWORD	string	Пароль для доступа к зашифрованному контенту

writePrivateKey(filename: string, format: DataFormat, password: string): any

Запись приватного ключа в файл

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	путь к файлу
FORMAT	DataFormat	Формат данных. Опционально. По умолчанию DER
PASSWORD	string	Пароль для доступа к зашифрованному контенту

readPublicKey(filename: string, format: DataFormat): Key

Чтение открытого ключа из файла

ПАРАМЕТР	ТИП	ОПИСАНИЕ
----------	-----	----------

<b>FILENAME</b>	string	путь к файлу
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

writePublicKey(filename: string, format: DataFormat): any

*Запись открытого ключа в файл*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>FILENAME</b>	string	путь к файлу
<b>FORMAT</b>	DataFormat	Формат данных. Опционально. По умолчанию DER

compare(key: Key): number

*Сравнение ключей*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>KEY</b>	Key	Экземпляр объекта Key

КЛАСС `OID`

*Представление объектного идентификатора (OID'a)*

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
<code>CONSTRUCTOR(OID: STRING)</code>	Конструктор по умолчанию
<code>CONSTRUCTOR(HANDLE: NATIVE.PKI.OID)</code>	Конструктор копированием экземпляра объекта <code>Key</code>

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
<code>VALUE</code>	<code>string</code>	Возвращает текстовое представление значения OID
<code>LONGNAME</code>	<code>string</code>	Возвращает длинное имя OID
<code>SHORTNAME</code>	<code>string</code>	Возвращает короткое имя OID

----- ПРИМЕР -----

```
var oid = new trusted.Pki.Oid("2.5.4.3");
console.log(oid.longName);           // commonName
console.log(oid.shortName);          // CN
console.log(oid.value);               // 2.5.4.3
```

-----



КЛАСС PKCS12

Представление интерфейса PKCS12

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR()	Конструктор по умолчанию
CONSTRUCTOR(HANDLE: NATIVE.PKI.PKCS12)	Конструктор копированием экземпляра объекта Pkcs12

METHODS -----

МЕТОД	ОПИСАНИЕ
CERTIFICATE	Возвращает сертификат
KEY	Возвращает приватный ключ
CA	Возвращает цепочку сертификатов
LOAD	Чтение pkcs12 из файла
SAVE	Сохранение pkcs12 в файл
CREATE	Создание структуры PKCS12

certificate(password: string): Certificate

Возвращает сертификат.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
PASSWORD	string	Пароль доступа к зашифрованному контенту

key(password: string): Key

Возвращает приватный ключ.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
PASSWORD	string	Пароль доступа к зашифрованному контенту

ca(password: string): CertificateCollection

Возвращает цепочку сертификатов.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
PASSWORD	string	Пароль доступа к зашифрованному контенту

load(filename: string): void

Чтение pkcs12 из файла.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу

static load(filename: string): Pkcs12

Чтение сертификата из файла.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу

save(filename: string): void

*Сохранение pkcs12 в файл.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	string	Путь к файлу

create(cert: Certificate, key: Key, ca: CertificateCollection, password: string, name: string):  
Pkcs12

*Создание структуры PKCS12.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CERT	Certificate	Сертификат
KEY	Key	Приватный ключ
CA	CertificateCollection	Цепочка сертификатов
PASSWORD	String	Пароль доступа к зашифрованному контенту
NAME	string	

КЛАСС PROVIDERSYSTEM

*Представление системного провайдера*

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR(FILENAME: STRING) CONSTRUCTOR(HANDLE: NATIVE.PKI.PROVIDERSYSTEM)	

METHODS -----

МЕТОД	ОПИСАНИЕ
FILLINGCACHE READJSON	

fillingCache(cacheURI: string, pvdURI: string): void

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CACHEURI PVDURI	string string	

readJson(filename: string): string

ПАРАМЕТР	ТИП	ОПИСАНИЕ
FILENAME	String	Путь к файлу

КЛАСС REVOCATION

*Представление Revocation-провайдера*

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR()	Конструктор по умолчанию

METHODS -----

МЕТОД	ОПИСАНИЕ
GETCRLLOCAL GETCRLDISTPOINTS CHECKCRLTIME DOWNLOADCRL	

getCrlLocal(cert: Certificate, store: PkiStore): any

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CERT STORE	Certificate PkiStore	

getCrlDistPoints(cert: Certificate): Array<string>

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CERT	Certificate	

checkCrlTime(crl: Crl): boolean

ПАРАМЕТР	ТИП	ОПИСАНИЕ
CRL	Crl	

downloadCRL(distPoints: Array<string>, pathForSave: string, done: Function): void

ПАРАМЕТР	ТИП	ОПИСАНИЕ
DISTPOINTS PATHFORSAVE DONE	Array<string> String Function	

PKISTORE

КЛАСС CASHJSON

Представление кэша хранилищ объектов PKI

CONSTRUCTORS -----

КОНСТРУКТОР	ОПИСАНИЕ
CONSTRUCTOR(FILENAME: STRING)	Конструктор с параметром пути к файлу JSON

METHODS -----

МЕТОД	ОПИСАНИЕ
EXPORT	Выполняет экспорт из файла кэша
IMPORT	Выполняет импорт в файл кэша

export(): native.PKISTORE.IPkiltem[]

Выполняет экспорт из файла кэша.

import(items: native.PKISTORE.IPkiltem[]): void

Выполняет импорт в файл кэша.

ПАРАМЕТР	ТИП	ОПИСАНИЕ
ITEMS	native.PKISTORE.IPkiltem[]	Массив описателей объектов PKI

КЛАСС FILTER

*Представление фильтра для выборки объектов PKI*

CONSTRUCTORS -----

КОНСТРУКТОР		ОПИСАНИЕ
CONSTRUCTOR()		Конструктор по умолчанию

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
TYPES	string	
PROVIDERS	string	
CATEGORYS	string	
HASH	String	
SUBJECTNAME	string	
SUBJECTFRIENDLYNAME	string	
ISSUERNAME	string	
ISSUERFRIENDLYNAME	String	
SERIAL	string	

КЛАСС PKIITEM

Представление описателей PKI объектов

CONSTRUCTORS -----

КОНСТРУКТОР		ОПИСАНИЕ
CONSTRUCTOR()		Конструктор по умолчанию

PROPERTIES -----

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
FORMAT	string	Возвращает формат представления объекта
TYPE	string	Возвращает тип объекта
PROVIDER	string	Возвращает провайдер хранилища в котором размещен объект
CATEGORY	string	Возвращает категорию хранилища (MY, OTHER, TRUST)
URI	string	Возвращает URI к физическому месторасположению объекта
HASH	string	Возвращает хэш объекта
SUBJECTNAME	string	
SUBJECTFRIENDLYNAME	string	
ISSUENAME	string	
ISSUERFRIENDLYNAME	string	
SERIAL	string	Возвращает серийный номер сертификат
NOTBEFORE	String	Возвращает дату начала действия сертификата
NOTAFTER	string	Возвращает дату истечения срока действия сертификата
LASTUPDATE	string	Возвращает дату последнего обновления списка отзыва
NEXTUPDATE	string	Возвращает дату следующего выпуска списка отзыва
KEY	string	
KEYENC	boolean	
ORGANIZATIONNAME	string	
SIGNATUREALGORITHM	string	

## КЛАСС PKIStore

Класс для организации стека хранилищ сертификатов и ключей. Создается кэш (JSON) для формирования описателей объектов хранения.

### CONSTRUCTORS

КОНСТРУКТОР	ОПИСАНИЕ
<b>CONSTRUCTOR(FOLDER: STRING)</b>	Конструктор по умолчанию
<b>CONSTRUCTOR(HANDLE: NATIVE.PKISTORE.PKISTORE)</b>	Конструктор копированием экземпляра объекта PkiStore

### PROPERTIES

НАЗВАНИЕ	ТИП	ОПИСАНИЕ
<b>CASH</b>	CashJson	Возвращает структуру кэша (JSON)

### METHODS

МЕТОД	ОПИСАНИЕ
<b>ADDPROVIDER</b>	Добавляет новый провайдер хранилища
<b>ADDCERT</b>	Добавляет описатель сертификата
<b>ADDCRL</b>	Добавляет описатель списка отзыва
<b>ADDKEY</b>	Добавляет описатель ключа
<b>ADDCSR</b>	Добавляет описатель запроса на сертификат
<b>FIND</b>	Поиск объекта в кэше
<b>FINDKEY</b>	Поиск ключа в кэше
<b>GETITEM</b>	Извлечение объекта по его описателю

`addProvider(provider: native.PKISTORE.Provider): void`

*Добавляет новый провайдер хранилища.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>PROVIDER</b>	native.PKISTORE.Provider	Провайдер хранилища

`addCert(provider: native.PKISTORE.Provider, category: string, cert: Certificate, flags: number): string`

*Добавляет описатель сертификата.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>PROVIDER</b>	native.PKISTORE.Provider	Провайдер хранилища
<b>CATEGORY</b>	string	Категория хранилища (MY, OTHER, TRUST)
<b>CERT</b>	Certificate	Сертификат для которого создается описатель
<b>FLAGS</b>	number	

`addCrl(provider: native.PKISTORE.Provider, category: string, crl: Crl, flags: number): string`

*Добавляет описатель списка отзыва.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
<b>PROVIDER</b>	native.PKISTORE.Provider	Провайдер хранилища
<b>CATEGORY</b>	string	Категория хранилища (MY, OTHER, TRUST)
<b>CRL</b>	Crl	Список отзыва для которого создается описатель
<b>FLAGS</b>	number	



addKey(provider: native.PKISTORE.Provider, key: Key, password: string): string

*Добавляет описатель ключа.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
PROVIDER	native.PKISTORE.Provider	Провайдер хранилища
KEY	Key	Ключ для которого создается описатель
PASSWORD	string	Пароль для доступа к зашифрованному контенту

addCsr(provider: native.PKISTORE.Provider, category: string, csr: CertificationRequest): string

*Добавляет описатель запроса на сертификат.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
PROVIDER	native.PKISTORE.Provider	Провайдер хранилища
CATEGORY	string	Категория хранилища (MY, OTHER, TRUST)
CSR	CertificationRequest	Запрос на сертификат для которого создается описатель

find(ifilter?: native.PKISTORE.IFilter): native.PKISTORE.IPkitem[]

*Поиск объекта в кэше.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
IFILTER	native.PKISTORE.IFilter	

findKey(ifilter: native.PKISTORE.IFilter): native.PKISTORE.IPkitem

*Поиск ключа в кэше.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
IFILTER	native.PKISTORE.IFilter	

getItem(item: native.PKISTORE.IPkitem): any

*Извлечение объекта по его описателю.*

ПАРАМЕТР	ТИП	ОПИСАНИЕ
ITEM	native.PKISTORE.IPkitem	