Java- JRT11

File	Modified
ZIP Archive jrt11-0.5-javadoc.zip	Aug 16, 2013 by

Java- JRT11 .

JRT11 , Java.

JRT11 Java 2 Runtime Environment 1.6 , Oracle Open JDK.

JRT11 28147-89, 34.10-2001, 34.11-94 RFC 4357 Java Cryptography Architecture (JCA). :

- , 34.10-2001;
- , 28147-89;
- -;
- ;
- 28147-89;
- / 34.10-2001;
- 34.11-94;
- 28147-89;
- ;
- RSA, 2048 .

JRT11 JCA .

JRT11 JCA

JCA Java, API , - , , (/, /), . JCA Java-, .

, JCA , , , , . .

, javax.crypto, java.security.

java.security API.

.

JCA , – , . , - (, 34.10-2001), .

JRT11 JCA .

, ,

<jre>\lib\ext\,

<jre> – JRE JDK.

JRT11 ru.rutoken.jrt11.JRT11Provider, Provider / . "JRT11".

Security:

addProvider(Provider provider)-	
insertProviderAt(Provider provider, int position)	
removeProvider(String name)	

1.

```
Security.addProvider(new ru.rutoken.jrt11.JRT11Provider());
```

<jre>\lib\security\java.security

security.provider.*n* = *masterClassName*,

n – (1 –), *masterClassName* – .

2.1.

```
security.provider.7=ru.rutoken.jrt11.JRT11Provider
```

, :

2.2. c *nix

```
security.provider.7=ru.rutoken.jrt11.JRT11Provider /jrt11/jrt11.properties
```

Windows :

2.3. c Windows

```
security.provider.7=ru.rutoken.jrt11.JRT11Provider C:\\jrt11\\jrt11.properties
```

. .

java.security

JCA

, / , , , getInstance(). , . getInstance() . . getInstance(String algorithm, String provider) getInstance(String algorithm, Provider provider) . .

3.1

```
MessageDigest digest = MessageDigest.getInstance("rtGOST3411", "JRT11");
```

AlgorithmParameterSpec

. , AlgorithmParameterSpec (, KeyPairGenerator.initialize(java.security.spec.AlgorithmParameterSpec) Signature.setParameter(java.security.spec.AlgorithmParameterSpec)). , AlgorithmParameterSpec, .

:

- Config - PKCS#11;
- Context - PKCS#11;
- Token - ;
- Session - ;
- ParamPin - PIN- ;
- ParamSet - .

PKCS#11 .

, , (MessageDigest) (SecureRandom) , PIN-, . .

. "/" OID.

3.2

```
KeyPairGenerator generator = KeyPairGenerator.getInstance("rtGOST3410/1.2.643.2.2.35.1", "JRT11");
```

JRT11 28147-89 JCA KeyGenerator.

28147-89 KeyGenerator getInstance() "rt11GOST28147".

generateKey().

KeyGenerator JCA ( ).

4. 28147-89

```
KeyGenerator generator = KeyGenerator.getInstance("rt11GOST28147", "JRT11");
SecretKey secret = generator.generateKey();
```

JRT11 34.10-2001 JCA KeyPairGenerator.

34.10-2001 KeyPairGenerator getInstance() "rtGOST3410".

generateKeyPair().

KeyPairGenerator JCA( ).

5.1. 34-10.2001

```
KeyGenerator generator = KeyGenerator.getInstance("rtGOST3410", "JRT11");
KeyPair pair = generator.generateKeyPair();
```

JRT11 KeyPairGenerator() , . initialize() ParamAlias "alias".

5.2. 34-10.2001

```
KeyPairGenerator generator = KeyPairGenerator.getInstance("rtGOST3410", "JRT11");
generator.initialize(new ru.rutoken.security.spec.ParamAlias("alias"));
KeyPair pair = generator.generateKeyPair();
```

.

RSA "rtRSA". KeyPairGenerator.initialize(int).

JRT11 - JCA KeyAgreement.

28147-89 KeyAgreement getInstance() "rtGOST3410".

. , , IvParameterSpec, . doPhase(), . generateSecret() "rt11GOST28147".

KeyAgreement JCA( ).

6. -

```
KeyAgreement agreement = KeyAgreement.getInstance("rtGOST3410", "JRT11");
agreement.init(privateKey, new IvParameterSpec(ivBytes));
agreement.doPhase(publicKey, true);
SecretKey secret = agreement.generateSecret("rt11GOST28147");
```

JRT11 28147-89 JCA Cipher.

28147-89 Cipher getInstance() "rt11GOST28147".

init(). update()(/) / doFinal(). update() .

Cipher JCA( ).

7.1. 28147-89

```
cipher = Cipher.getInstance("rt11GOST28147", "JRT11");
cipher.init(Cipher.ENCRYPT_MODE, secret, new IvParameterSpec(iv));
byte[] result = cipher.doFinal(data);
```

7.2. 28147-89

```
cipher = Cipher.getInstance("rt11GOST28147", "JRT11");
cipher.init(Cipher.DECRYPT_MODE, secret, new IvParameterSpec(iv));
byte[] result = cipher.doFinal(data);
```

28147-89 Cipher getInstance() "rt11GOST28147".

init(). wrap(), - unwrap().

Cipher JCA ( ).

8.1. 28147-89

```
cipher = Cipher.getInstance("rt11GOST28147", "JRT11");
cipher.init(Cipher.WRAP_MODE, secret);
byte[] result = cipher.wrap(wrappedKey);
```

8.2. 28147-89

```
cipher = Cipher.getInstance("rt11GOST28147", "JRT11");
cipher.init(Cipher.UNWRAP_MODE, secret);
byte[] result = cipher.unwrap(unwrappedKey);
```

JRT11 34.11-94 JCA MessageDigest.

34.11-94 getInstance() , –"rtGOST3411". update(). - digest().

MessageDigest JCA ( ).

9. - 34.11-94

```
MessageDigest messageDigest = MessageDigest.getInstance("rtGOST3411", "JRT11");
messageDigest.update("rtGOST3411".getBytes());
byte[] digestValue = messageDigest.digest();
```

JRT11 34.10-2001 JCA Signature.

/ 34.10-94 getInstance() , –"rtGOST3411withrtGOST3410". initSign() / initVerify(). / update(). sign(), – verify().

Signature JCA ( ).

10.1. 34.10-2001

```
Signature signature = Signature.getInstance("rtGOST3411withrtGOST3410", "JRT11");
signature.initSign(privateKey);
signature.update(TEXT);
byte[] signatureValue = signature.sign();
```

10.2. 34.10-2001

```
Signature signature = Signature.getInstance("rtGOST3411withrtGOST3410", "JRT11");
signature.initVerify(publicKey);
signature.update(TEXT);
boolean result = signature.verify(signatureValue));
```

RSA "MD5withRSA".

JRT11 28147-89 JCA Mac.

28147-89 Mac getInstance() "rt11GOST28147".

init(). update()( ) doFinal(). update() .

Mac JCA( ).

11. 28147-89

```
Mac mac = Mac.getInstance("rt11GOST28147", "JRT11");
mac.init(secret);
mac.update(dataBytes);
byte[] macValue = mac.doFinal();
```

JRT11 JCA SecureRandom.

SecureRandom getInstance() "rtRandom". nextBytes(), .

SecureRandom JCA.

12.

```
byte[] bytes = new byte[8];
SecureRandom random = SecureRandom.getInstance("rtRandom", "JRT11");
random.nextBytes(bytes);
```

JRT11 JCA KeyStore. , KeyPairGenerator .

KeyStore getInstance() "rtStore". KeyStore.load(InputStream, char[]) PIN- , InputStream .

KeyStore JCA.

13.1.

```
KeyStore keyStore = KeyStore.getInstance("rtStore", "JRT11");
keyStore.load(null, "12345678".toCharArray());
PrivateKey privateKey = (PrivateKey)keyStore.getKey("alias", null);
```

KeyStore , "rtStore" , : KeyStore.Entry KeyStore.getEntry(String, ProtectionParameter), .

13.2.

```
KeyStore.Entry entry = keyStore.getEntry("alias", null);
PublicKey publicKey = ((ru.rutoken.security.KeyContainer)entry).getPublicKey();
```

JRT11 JCA

, JCA :

- ;
- ;
- , ;
- ;
- Java- .

JRT11 , JCA .

JRT11

JCA, SecureRandom, MessageDigest, KeyPairGenerator, Signature java.security, JRT11 SecureRandom, MessageDigest, KeyPairGenerator, Signature ru.rutoken.security. JCA , : , , .

,

14. JRT11

```
MessageDigest digest = MessageDigest.getInstance(algorithm);
digest.update(data);
byte[] result = digest.digest();
```

java.security.MessageDigest, ru.rutoken.security.MessageDigest.

import. ru.rutoken.security JavaDoc.

JCA/JCE javax.crypto. , JRT11 ru.rutoken.crypto.

(, , ,) . , , .

Java , . JRT11 .

JRT11 ru.rutoken.security.Cleaner. , , Cleaner.clean(Object), finally .

15.

```
public byte[] sign(byte[] data, PrivateKey key) throws Exception
{
        Signature signature = null;
    try
        {
            signature = Signature.getInstance(JRT11Provider.GOST_SIGNATURE_ALGORITHM, JRT11Provider.
PROVIDER_NAME);
        signature.initSign(key);
        signature.update(data);
        return signature.sign();
     }
        finally
        {
        Cleaner.clean(signature);
    }
}
```

,

- ,
- , ,
- , ,
- PIN- ,
- , .

JRT11 , RFC 4357 . . , , . .

ru.rutoken.security.spec.gost (JavaDoc).

: (Crypt), (Digest), (Elliptic) (Exchange).

(Exchange) 34.10-2001 , (Exchange), , .

(Elliptic) - .

: , .

, OID RFC 4357. OID ITU-T Rec. X.690 (07/2002), public byte[] getEncodedOID().

:

- ;
- ;
- OID;
- OID;
- ASN.1 .

, JRT11, . .

.

16.1.

```
KeyPairGenerator generator = KeyPairGenerator.getInstance("rtGOST3410", "JRT11");
generator.initialize(EllipticParamFactory.getInstance());
KeyPair pair = generator.generateKeyPair();
```

16.2.

```
Mac mac = Mac.getInstance("rt11GOST28147", "JRT11");
mac.init(key, CryptParamFactory.getInstance("1.2.643.2.2.31.1"));
```

c java- java.security.spec.AlgorithmParameterSpec. , , AlgorithmParameterSpec, (.).

. "/" OID.

16.3.

```
KeyPairGenerator generator = KeyPairGenerator.getInstance("rtGOST3410/1.2.643.2.2.35.1", "JRT11");
```

JCA java.security , ru.rutoken.security.

PKCS#11

PKCS#11 3 :

- PKCS#11 (Library),

- (Serial)
- PIN- (Pin).

Library , PKCS#11, . , JRT11 . , Windows JRT11 C:\Windows\System32\rtPKCS11ECP.dll.

Serial , JRT11. (ID hex). . , JRT11 .

Pin PIN- .

PKCS#11 .

17.

```
Library=C:\Windows\System32\rtPKCS11ECP.dll
Serial=2b356eb7
Pin=12345678
```

ru.rutoken.jrt11.Config (JavaDoc).

Config AlgorithmParameterSpec, , , KeyPairGenerator.initialize(int).

SessionFactory

ru.rutoken.jrt11.SessionFactory PKCS#11, PKCS#11: , , . JCA . JavaDoc.

:

1. SessionFactory;
2. SessionFactory setParameter();
3. .

18. SessionFactory

```
SessionFactory factory = new SessionFactory();
factory.setParameter(new ParamPin(tokenPin));
AlgorithmParameterSpec token = factory.getToken();
```

SessionFactory Config ParamPin.

token JCA .

getSession() .

PKCS#11

PKCS#11.

1. . , , .

2. .

19.1. PKCS#11

```
System.setProperty("ru.rutoken.jrt11.ConfigLibrary", "C:\\Windows\\System32\\rtPKCS11ECP.dll");
System.setProperty("ru.rutoken.jrt11.ConfigSerial", "2b356eb7");
System.setProperty("ru.rutoken.jrt11.ConfigPin", "12345678");
```

. .

3. .

19.2. PKCS#11

```
Config config = new Config("C:\\Windows\\System32\\rtPKCS11ECP.dll", "2b356eb7", "12345678");
Security.addProvider(new JRT11Provider(config));
```

. .

4. .

19.3. PKCS#11

```
String configFileName = "C:\\jrt11\\jrt11.properties";
Security.addProvider(new JRT11Provider(configFileName));
```

5. <jre>\lib\security\java.security:

```
security.provider.7=ru.rutoken.jrt11.JRT11Provider C:\\jrt11\\jrt11.properties
```

6. ,

19.4. PKCS#11

```
Config config = new Config("C:\\Windows\\System32\\rtPKCS11ECP.dll", "2b356eb7", "12345678");
generator.initialize(config);
```

.

19.5. PKCS#11

```
Config config = new Config(null, null, "12345678");
generator.initialize(config);
```

7. PIN- , .

19.6. PIN-

```
generator.initialize(new ru.rutoken.security.spec.ParamPin("12345678"));
```

8. PKCS#11 .

19.6. PKCS#11

```
Config config = new Config("C:\\Windows\\System32\\rtPKCS11ECP.dll", "2b356eb7", "12345678");
sessionFactory.setParameter(config);
generator.initialize(sessionFactory.getToken());
```

.

9. PKCS#11 .

19.6. PKCS#11

```
generator.initialize(sessionFactory.getContext());
```

10. PKCS#11 .

19.6. PKCS#11

```
generator.initialize(sessionFactory.getSession());
```

, , .

Related links

1. JCA: http://docs.oracle.com/javase/6/docs/technotes/guides/security/crypto/CryptoSpec.html
2. Java SE 6 API http://docs.oracle.com/javase/6/docs/api/index.html
3. JDK/JDE http://www.oracle.com/technetwork/java/javase/downloads/index.html
4. maven http://www.apache-maven.ru/index.html

1. JavaDoc jrt11-0.5-javadoc.zip