| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/security/Key.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/security//class-useKey.html)    [**NO FRAMES**](http://docs.google.com/Key.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Interface**

**java.security.Key**

| Packages that use [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| [**java.security**](#3znysh7) | Provides the classes and interfaces for the security framework. |
| [**java.security.interfaces**](#2et92p0) | Provides interfaces for generating RSA (Rivest, Shamir and Adleman AsymmetricCipher algorithm) keys as defined in the RSA Laboratory Technical Note PKCS#1, and DSA (Digital Signature Algorithm) keys as defined in NIST's FIPS-186. |
| [**javax.crypto**](#tyjcwt) | Provides the classes and interfaces for cryptographic operations. |
| [**javax.crypto.interfaces**](#3dy6vkm) | Provides interfaces for Diffie-Hellman keys as defined in RSA Laboratories' PKCS #3. |
| [**javax.crypto.spec**](#1t3h5sf) | Provides classes and interfaces for key specifications and algorithm parameter specifications. |
| [**javax.security.auth.kerberos**](#4d34og8) | This package contains utility classes related to the Kerberos network authentication protocol. |
| [**javax.xml.crypto**](#2s8eyo1) | Common classes for XML cryptography. |
| [**javax.xml.crypto.dsig.dom**](#17dp8vu) | DOM-specific classes for the [javax.xml.crypto.dsig](http://docs.google.com/javax/xml/crypto/dsig/package-summary.html) package. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [java.security](http://docs.google.com/java/security/package-summary.html) | |
| --- | --- |

| Subinterfaces of [Key](http://docs.google.com/java/security/Key.html) in [java.security](http://docs.google.com/java/security/package-summary.html) | |
| --- | --- |
| interface | [**PrivateKey**](http://docs.google.com/java/security/PrivateKey.html)            A private key. |
| interface | [**PublicKey**](http://docs.google.com/java/security/PublicKey.html)            A public key. |

| Methods in [java.security](http://docs.google.com/java/security/package-summary.html) that return [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| abstract  [Key](http://docs.google.com/java/security/Key.html) | **KeyStoreSpi.**[**engineGetKey**](http://docs.google.com/java/security/KeyStoreSpi.html#engineGetKey(java.lang.String,%20char%5B%5D))([String](http://docs.google.com/java/lang/String.html) alias, char[] password)            Returns the key associated with the given alias, using the given password to recover it. |
| protected abstract  [Key](http://docs.google.com/java/security/Key.html) | **KeyFactorySpi.**[**engineTranslateKey**](http://docs.google.com/java/security/KeyFactorySpi.html#engineTranslateKey(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Translates a key object, whose provider may be unknown or potentially untrusted, into a corresponding key object of this key factory. |
| [Key](http://docs.google.com/java/security/Key.html) | **KeyStore.**[**getKey**](http://docs.google.com/java/security/KeyStore.html#getKey(java.lang.String,%20char%5B%5D))([String](http://docs.google.com/java/lang/String.html) alias, char[] password)            Returns the key associated with the given alias, using the given password to recover it. |
| [Key](http://docs.google.com/java/security/Key.html) | **KeyFactory.**[**translateKey**](http://docs.google.com/java/security/KeyFactory.html#translateKey(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Translates a key object, whose provider may be unknown or potentially untrusted, into a corresponding key object of this key factory. |

| Methods in [java.security](http://docs.google.com/java/security/package-summary.html) with parameters of type [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| protected abstract   | <T extends [KeySpec](http://docs.google.com/java/security/spec/KeySpec.html)>  T | | --- | | **KeyFactorySpi.**[**engineGetKeySpec**](http://docs.google.com/java/security/KeyFactorySpi.html#engineGetKeySpec(java.security.Key,%20java.lang.Class))([Key](http://docs.google.com/java/security/Key.html) key, [Class](http://docs.google.com/java/lang/Class.html)<T> keySpec)            Returns a specification (key material) of the given key object. |
| abstract  void | **KeyStoreSpi.**[**engineSetKeyEntry**](http://docs.google.com/java/security/KeyStoreSpi.html#engineSetKeyEntry(java.lang.String,%20java.security.Key,%20char%5B%5D,%20java.security.cert.Certificate%5B%5D))([String](http://docs.google.com/java/lang/String.html) alias, [Key](http://docs.google.com/java/security/Key.html) key, char[] password, [Certificate](http://docs.google.com/java/security/cert/Certificate.html)[] chain)            Assigns the given key to the given alias, protecting it with the given password. |
| protected abstract  [Key](http://docs.google.com/java/security/Key.html) | **KeyFactorySpi.**[**engineTranslateKey**](http://docs.google.com/java/security/KeyFactorySpi.html#engineTranslateKey(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Translates a key object, whose provider may be unknown or potentially untrusted, into a corresponding key object of this key factory. |
| | <T extends [KeySpec](http://docs.google.com/java/security/spec/KeySpec.html)>  T | | --- | | **KeyFactory.**[**getKeySpec**](http://docs.google.com/java/security/KeyFactory.html#getKeySpec(java.security.Key,%20java.lang.Class))([Key](http://docs.google.com/java/security/Key.html) key, [Class](http://docs.google.com/java/lang/Class.html)<T> keySpec)            Returns a specification (key material) of the given key object. |
| void | **KeyStore.**[**setKeyEntry**](http://docs.google.com/java/security/KeyStore.html#setKeyEntry(java.lang.String,%20java.security.Key,%20char%5B%5D,%20java.security.cert.Certificate%5B%5D))([String](http://docs.google.com/java/lang/String.html) alias, [Key](http://docs.google.com/java/security/Key.html) key, char[] password, [Certificate](http://docs.google.com/java/security/cert/Certificate.html)[] chain)            Assigns the given key to the given alias, protecting it with the given password. |
| [Key](http://docs.google.com/java/security/Key.html) | **KeyFactory.**[**translateKey**](http://docs.google.com/java/security/KeyFactory.html#translateKey(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Translates a key object, whose provider may be unknown or potentially untrusted, into a corresponding key object of this key factory. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [java.security.interfaces](http://docs.google.com/java/security/interfaces/package-summary.html) | |
| --- | --- |

| Subinterfaces of [Key](http://docs.google.com/java/security/Key.html) in [java.security.interfaces](http://docs.google.com/java/security/interfaces/package-summary.html) | |
| --- | --- |
| interface | [**DSAPrivateKey**](http://docs.google.com/java/security/interfaces/DSAPrivateKey.html)            The standard interface to a DSA private key. |
| interface | [**DSAPublicKey**](http://docs.google.com/java/security/interfaces/DSAPublicKey.html)            The interface to a DSA public key. |
| interface | [**ECPrivateKey**](http://docs.google.com/java/security/interfaces/ECPrivateKey.html)            The interface to an elliptic curve (EC) private key. |
| interface | [**ECPublicKey**](http://docs.google.com/java/security/interfaces/ECPublicKey.html)            The interface to an elliptic curve (EC) public key. |
| interface | [**RSAMultiPrimePrivateCrtKey**](http://docs.google.com/java/security/interfaces/RSAMultiPrimePrivateCrtKey.html)            The interface to an RSA multi-prime private key, as defined in the PKCS#1 v2.1, using the *Chinese Remainder Theorem* (CRT) information values. |
| interface | [**RSAPrivateCrtKey**](http://docs.google.com/java/security/interfaces/RSAPrivateCrtKey.html)            The interface to an RSA private key, as defined in the PKCS#1 standard, using the *Chinese Remainder Theorem* (CRT) information values. |
| interface | [**RSAPrivateKey**](http://docs.google.com/java/security/interfaces/RSAPrivateKey.html)            The interface to an RSA private key. |
| interface | [**RSAPublicKey**](http://docs.google.com/java/security/interfaces/RSAPublicKey.html)            The interface to an RSA public key. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [javax.crypto](http://docs.google.com/javax/crypto/package-summary.html) | |
| --- | --- |

| Subinterfaces of [Key](http://docs.google.com/java/security/Key.html) in [javax.crypto](http://docs.google.com/javax/crypto/package-summary.html) | |
| --- | --- |
| interface | [**SecretKey**](http://docs.google.com/javax/crypto/SecretKey.html)            A secret (symmetric) key. |

| Methods in [javax.crypto](http://docs.google.com/javax/crypto/package-summary.html) that return [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| [Key](http://docs.google.com/java/security/Key.html) | **KeyAgreement.**[**doPhase**](http://docs.google.com/javax/crypto/KeyAgreement.html#doPhase(java.security.Key,%20boolean))([Key](http://docs.google.com/java/security/Key.html) key, boolean lastPhase)            Executes the next phase of this key agreement with the given key that was received from one of the other parties involved in this key agreement. |
| protected abstract  [Key](http://docs.google.com/java/security/Key.html) | **KeyAgreementSpi.**[**engineDoPhase**](http://docs.google.com/javax/crypto/KeyAgreementSpi.html#engineDoPhase(java.security.Key,%20boolean))([Key](http://docs.google.com/java/security/Key.html) key, boolean lastPhase)            Executes the next phase of this key agreement with the given key that was received from one of the other parties involved in this key agreement. |
| protected  [Key](http://docs.google.com/java/security/Key.html) | **CipherSpi.**[**engineUnwrap**](http://docs.google.com/javax/crypto/CipherSpi.html#engineUnwrap(byte%5B%5D,%20java.lang.String,%20int))(byte[] wrappedKey, [String](http://docs.google.com/java/lang/String.html) wrappedKeyAlgorithm, int wrappedKeyType)            Unwrap a previously wrapped key. |
| [Key](http://docs.google.com/java/security/Key.html) | **Cipher.**[**unwrap**](http://docs.google.com/javax/crypto/Cipher.html#unwrap(byte%5B%5D,%20java.lang.String,%20int))(byte[] wrappedKey, [String](http://docs.google.com/java/lang/String.html) wrappedKeyAlgorithm, int wrappedKeyType)            Unwrap a previously wrapped key. |

| Methods in [javax.crypto](http://docs.google.com/javax/crypto/package-summary.html) with parameters of type [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| [Key](http://docs.google.com/java/security/Key.html) | **KeyAgreement.**[**doPhase**](http://docs.google.com/javax/crypto/KeyAgreement.html#doPhase(java.security.Key,%20boolean))([Key](http://docs.google.com/java/security/Key.html) key, boolean lastPhase)            Executes the next phase of this key agreement with the given key that was received from one of the other parties involved in this key agreement. |
| protected abstract  [Key](http://docs.google.com/java/security/Key.html) | **KeyAgreementSpi.**[**engineDoPhase**](http://docs.google.com/javax/crypto/KeyAgreementSpi.html#engineDoPhase(java.security.Key,%20boolean))([Key](http://docs.google.com/java/security/Key.html) key, boolean lastPhase)            Executes the next phase of this key agreement with the given key that was received from one of the other parties involved in this key agreement. |
| protected  int | **CipherSpi.**[**engineGetKeySize**](http://docs.google.com/javax/crypto/CipherSpi.html#engineGetKeySize(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Returns the key size of the given key object in bits. |
| protected abstract  void | **CipherSpi.**[**engineInit**](http://docs.google.com/javax/crypto/CipherSpi.html#engineInit(int,%20java.security.Key,%20java.security.spec.AlgorithmParameterSpec,%20java.security.SecureRandom))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this cipher with a key, a set of algorithm parameters, and a source of randomness. |
| protected abstract  void | **CipherSpi.**[**engineInit**](http://docs.google.com/javax/crypto/CipherSpi.html#engineInit(int,%20java.security.Key,%20java.security.AlgorithmParameters,%20java.security.SecureRandom))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameters](http://docs.google.com/java/security/AlgorithmParameters.html) params, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this cipher with a key, a set of algorithm parameters, and a source of randomness. |
| protected abstract  void | **CipherSpi.**[**engineInit**](http://docs.google.com/javax/crypto/CipherSpi.html#engineInit(int,%20java.security.Key,%20java.security.SecureRandom))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this cipher with a key and a source of randomness. |
| protected abstract  void | **ExemptionMechanismSpi.**[**engineInit**](http://docs.google.com/javax/crypto/ExemptionMechanismSpi.html#engineInit(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Initializes this exemption mechanism with a key. |
| protected abstract  void | **ExemptionMechanismSpi.**[**engineInit**](http://docs.google.com/javax/crypto/ExemptionMechanismSpi.html#engineInit(java.security.Key,%20java.security.AlgorithmParameters))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameters](http://docs.google.com/java/security/AlgorithmParameters.html) params)            Initializes this exemption mechanism with a key and a set of algorithm parameters. |
| protected abstract  void | **MacSpi.**[**engineInit**](http://docs.google.com/javax/crypto/MacSpi.html#engineInit(java.security.Key,%20java.security.spec.AlgorithmParameterSpec))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params)            Initializes the MAC with the given (secret) key and algorithm parameters. |
| protected abstract  void | **ExemptionMechanismSpi.**[**engineInit**](http://docs.google.com/javax/crypto/ExemptionMechanismSpi.html#engineInit(java.security.Key,%20java.security.spec.AlgorithmParameterSpec))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params)            Initializes this exemption mechanism with a key and a set of algorithm parameters. |
| protected abstract  void | **KeyAgreementSpi.**[**engineInit**](http://docs.google.com/javax/crypto/KeyAgreementSpi.html#engineInit(java.security.Key,%20java.security.spec.AlgorithmParameterSpec,%20java.security.SecureRandom))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this key agreement with the given key, set of algorithm parameters, and source of randomness. |
| protected abstract  void | **KeyAgreementSpi.**[**engineInit**](http://docs.google.com/javax/crypto/KeyAgreementSpi.html#engineInit(java.security.Key,%20java.security.SecureRandom))([Key](http://docs.google.com/java/security/Key.html) key, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this key agreement with the given key and source of randomness. |
| protected  byte[] | **CipherSpi.**[**engineWrap**](http://docs.google.com/javax/crypto/CipherSpi.html#engineWrap(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Wrap a key. |
| [PKCS8EncodedKeySpec](http://docs.google.com/java/security/spec/PKCS8EncodedKeySpec.html) | **EncryptedPrivateKeyInfo.**[**getKeySpec**](http://docs.google.com/javax/crypto/EncryptedPrivateKeyInfo.html#getKeySpec(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) decryptKey)            Extract the enclosed PKCS8EncodedKeySpec object from the encrypted data and return it. |
| [PKCS8EncodedKeySpec](http://docs.google.com/java/security/spec/PKCS8EncodedKeySpec.html) | **EncryptedPrivateKeyInfo.**[**getKeySpec**](http://docs.google.com/javax/crypto/EncryptedPrivateKeyInfo.html#getKeySpec(java.security.Key,%20java.security.Provider))([Key](http://docs.google.com/java/security/Key.html) decryptKey, [Provider](http://docs.google.com/java/security/Provider.html) provider)            Extract the enclosed PKCS8EncodedKeySpec object from the encrypted data and return it. |
| [PKCS8EncodedKeySpec](http://docs.google.com/java/security/spec/PKCS8EncodedKeySpec.html) | **EncryptedPrivateKeyInfo.**[**getKeySpec**](http://docs.google.com/javax/crypto/EncryptedPrivateKeyInfo.html#getKeySpec(java.security.Key,%20java.lang.String))([Key](http://docs.google.com/java/security/Key.html) decryptKey, [String](http://docs.google.com/java/lang/String.html) providerName)            Extract the enclosed PKCS8EncodedKeySpec object from the encrypted data and return it. |
| [Object](http://docs.google.com/java/lang/Object.html) | **SealedObject.**[**getObject**](http://docs.google.com/javax/crypto/SealedObject.html#getObject(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Retrieves the original (encapsulated) object. |
| [Object](http://docs.google.com/java/lang/Object.html) | **SealedObject.**[**getObject**](http://docs.google.com/javax/crypto/SealedObject.html#getObject(java.security.Key,%20java.lang.String))([Key](http://docs.google.com/java/security/Key.html) key, [String](http://docs.google.com/java/lang/String.html) provider)            Retrieves the original (encapsulated) object. |
| void | **Cipher.**[**init**](http://docs.google.com/javax/crypto/Cipher.html#init(int,%20java.security.Key))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key)            Initializes this cipher with a key. |
| void | **Cipher.**[**init**](http://docs.google.com/javax/crypto/Cipher.html#init(int,%20java.security.Key,%20java.security.AlgorithmParameters))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameters](http://docs.google.com/java/security/AlgorithmParameters.html) params)            Initializes this cipher with a key and a set of algorithm parameters. |
| void | **Cipher.**[**init**](http://docs.google.com/javax/crypto/Cipher.html#init(int,%20java.security.Key,%20java.security.spec.AlgorithmParameterSpec))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params)            Initializes this cipher with a key and a set of algorithm parameters. |
| void | **Cipher.**[**init**](http://docs.google.com/javax/crypto/Cipher.html#init(int,%20java.security.Key,%20java.security.spec.AlgorithmParameterSpec,%20java.security.SecureRandom))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this cipher with a key, a set of algorithm parameters, and a source of randomness. |
| void | **Cipher.**[**init**](http://docs.google.com/javax/crypto/Cipher.html#init(int,%20java.security.Key,%20java.security.AlgorithmParameters,%20java.security.SecureRandom))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameters](http://docs.google.com/java/security/AlgorithmParameters.html) params, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this cipher with a key, a set of algorithm parameters, and a source of randomness. |
| void | **Cipher.**[**init**](http://docs.google.com/javax/crypto/Cipher.html#init(int,%20java.security.Key,%20java.security.SecureRandom))(int opmode, [Key](http://docs.google.com/java/security/Key.html) key, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this cipher with a key and a source of randomness. |
| void | **ExemptionMechanism.**[**init**](http://docs.google.com/javax/crypto/ExemptionMechanism.html#init(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Initializes this exemption mechanism with a key. |
| void | **KeyAgreement.**[**init**](http://docs.google.com/javax/crypto/KeyAgreement.html#init(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Initializes this key agreement with the given key, which is required to contain all the algorithm parameters required for this key agreement. |
| void | **Mac.**[**init**](http://docs.google.com/javax/crypto/Mac.html#init(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Initializes this Mac object with the given key. |
| void | **ExemptionMechanism.**[**init**](http://docs.google.com/javax/crypto/ExemptionMechanism.html#init(java.security.Key,%20java.security.AlgorithmParameters))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameters](http://docs.google.com/java/security/AlgorithmParameters.html) params)            Initializes this exemption mechanism with a key and a set of algorithm parameters. |
| void | **ExemptionMechanism.**[**init**](http://docs.google.com/javax/crypto/ExemptionMechanism.html#init(java.security.Key,%20java.security.spec.AlgorithmParameterSpec))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params)            Initializes this exemption mechanism with a key and a set of algorithm parameters. |
| void | **KeyAgreement.**[**init**](http://docs.google.com/javax/crypto/KeyAgreement.html#init(java.security.Key,%20java.security.spec.AlgorithmParameterSpec))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params)            Initializes this key agreement with the given key and set of algorithm parameters. |
| void | **Mac.**[**init**](http://docs.google.com/javax/crypto/Mac.html#init(java.security.Key,%20java.security.spec.AlgorithmParameterSpec))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params)            Initializes this Mac object with the given key and algorithm parameters. |
| void | **KeyAgreement.**[**init**](http://docs.google.com/javax/crypto/KeyAgreement.html#init(java.security.Key,%20java.security.spec.AlgorithmParameterSpec,%20java.security.SecureRandom))([Key](http://docs.google.com/java/security/Key.html) key, [AlgorithmParameterSpec](http://docs.google.com/java/security/spec/AlgorithmParameterSpec.html) params, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this key agreement with the given key, set of algorithm parameters, and source of randomness. |
| void | **KeyAgreement.**[**init**](http://docs.google.com/javax/crypto/KeyAgreement.html#init(java.security.Key,%20java.security.SecureRandom))([Key](http://docs.google.com/java/security/Key.html) key, [SecureRandom](http://docs.google.com/java/security/SecureRandom.html) random)            Initializes this key agreement with the given key and source of randomness. |
| boolean | **ExemptionMechanism.**[**isCryptoAllowed**](http://docs.google.com/javax/crypto/ExemptionMechanism.html#isCryptoAllowed(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Returns whether the result blob has been generated successfully by this exemption mechanism. |
| byte[] | **Cipher.**[**wrap**](http://docs.google.com/javax/crypto/Cipher.html#wrap(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Wrap a key. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [javax.crypto.interfaces](http://docs.google.com/javax/crypto/interfaces/package-summary.html) | |
| --- | --- |

| Subinterfaces of [Key](http://docs.google.com/java/security/Key.html) in [javax.crypto.interfaces](http://docs.google.com/javax/crypto/interfaces/package-summary.html) | |
| --- | --- |
| interface | [**DHPrivateKey**](http://docs.google.com/javax/crypto/interfaces/DHPrivateKey.html)            The interface to a Diffie-Hellman private key. |
| interface | [**DHPublicKey**](http://docs.google.com/javax/crypto/interfaces/DHPublicKey.html)            The interface to a Diffie-Hellman public key. |
| interface | [**PBEKey**](http://docs.google.com/javax/crypto/interfaces/PBEKey.html)            The interface to a PBE key. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [javax.crypto.spec](http://docs.google.com/javax/crypto/spec/package-summary.html) | |
| --- | --- |

| Classes in [javax.crypto.spec](http://docs.google.com/javax/crypto/spec/package-summary.html) that implement [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| class | [**SecretKeySpec**](http://docs.google.com/javax/crypto/spec/SecretKeySpec.html)            This class specifies a secret key in a provider-independent fashion. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [javax.security.auth.kerberos](http://docs.google.com/javax/security/auth/kerberos/package-summary.html) | |
| --- | --- |

| Classes in [javax.security.auth.kerberos](http://docs.google.com/javax/security/auth/kerberos/package-summary.html) that implement [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| class | [**KerberosKey**](http://docs.google.com/javax/security/auth/kerberos/KerberosKey.html)            This class encapsulates a long term secret key for a Kerberos principal. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [javax.xml.crypto](http://docs.google.com/javax/xml/crypto/package-summary.html) | |
| --- | --- |

| Methods in [javax.xml.crypto](http://docs.google.com/javax/xml/crypto/package-summary.html) that return [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| [Key](http://docs.google.com/java/security/Key.html) | **KeySelectorResult.**[**getKey**](http://docs.google.com/javax/xml/crypto/KeySelectorResult.html#getKey())()            Returns the selected key. |

| Methods in [javax.xml.crypto](http://docs.google.com/javax/xml/crypto/package-summary.html) with parameters of type [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| static [KeySelector](http://docs.google.com/javax/xml/crypto/KeySelector.html) | **KeySelector.**[**singletonKeySelector**](http://docs.google.com/javax/xml/crypto/KeySelector.html#singletonKeySelector(java.security.Key))([Key](http://docs.google.com/java/security/Key.html) key)            Returns a KeySelector that always selects the specified key, regardless of the KeyInfo passed to it. |

| Uses of [Key](http://docs.google.com/java/security/Key.html) in [javax.xml.crypto.dsig.dom](http://docs.google.com/javax/xml/crypto/dsig/dom/package-summary.html) | |
| --- | --- |

| Constructors in [javax.xml.crypto.dsig.dom](http://docs.google.com/javax/xml/crypto/dsig/dom/package-summary.html) with parameters of type [Key](http://docs.google.com/java/security/Key.html) | |
| --- | --- |
| [**DOMSignContext**](http://docs.google.com/javax/xml/crypto/dsig/dom/DOMSignContext.html#DOMSignContext(java.security.Key,%20org.w3c.dom.Node))([Key](http://docs.google.com/java/security/Key.html) signingKey, [Node](http://docs.google.com/org/w3c/dom/Node.html) parent)            Creates a DOMSignContext with the specified signing key and parent node. |
| [**DOMSignContext**](http://docs.google.com/javax/xml/crypto/dsig/dom/DOMSignContext.html#DOMSignContext(java.security.Key,%20org.w3c.dom.Node,%20org.w3c.dom.Node))([Key](http://docs.google.com/java/security/Key.html) signingKey, [Node](http://docs.google.com/org/w3c/dom/Node.html) parent, [Node](http://docs.google.com/org/w3c/dom/Node.html) nextSibling)            Creates a DOMSignContext with the specified signing key, parent and next sibling nodes. |
| [**DOMValidateContext**](http://docs.google.com/javax/xml/crypto/dsig/dom/DOMValidateContext.html#DOMValidateContext(java.security.Key,%20org.w3c.dom.Node))([Key](http://docs.google.com/java/security/Key.html) validatingKey, [Node](http://docs.google.com/org/w3c/dom/Node.html) node)            Creates a DOMValidateContext containing the specified key and node. |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/security/Key.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/security//class-useKey.html)    [**NO FRAMES**](http://docs.google.com/Key.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).