| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.html) | [**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 PACKAGE**](http://docs.google.com/java/rmi/server/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/java/security/acl/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?java/security/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

## Package java.security

Provides the classes and interfaces for the security framework.

**See:**

[**Description**](#3znysh7)

| **Interface Summary** | |
| --- | --- |
| [**Certificate**](http://docs.google.com/java/security/Certificate.html) | **Deprecated.** *A new certificate handling package is created in the Java platform.* |
| [**DomainCombiner**](http://docs.google.com/java/security/DomainCombiner.html) | A DomainCombiner provides a means to dynamically update the ProtectionDomains associated with the current AccessControlContext. |
| [**Guard**](http://docs.google.com/java/security/Guard.html) | This interface represents a guard, which is an object that is used to protect access to another object. |
| [**Key**](http://docs.google.com/java/security/Key.html) | The Key interface is the top-level interface for all keys. |
| [**KeyStore.Entry**](http://docs.google.com/java/security/KeyStore.Entry.html) | A marker interface for KeyStore entry types. |
| [**KeyStore.LoadStoreParameter**](http://docs.google.com/java/security/KeyStore.LoadStoreParameter.html) | A marker interface for KeyStore [load](http://docs.google.com/java/security/KeyStore.html#load(java.security.KeyStore.LoadStoreParameter)) and [store](http://docs.google.com/java/security/KeyStore.html#store(java.security.KeyStore.LoadStoreParameter)) parameters. |
| [**KeyStore.ProtectionParameter**](http://docs.google.com/java/security/KeyStore.ProtectionParameter.html) | A marker interface for keystore protection parameters. |
| [**Policy.Parameters**](http://docs.google.com/java/security/Policy.Parameters.html) | This represents a marker interface for Policy parameters. |
| [**Principal**](http://docs.google.com/java/security/Principal.html) | This interface represents the abstract notion of a principal, which can be used to represent any entity, such as an individual, a corporation, and a login id. |
| [**PrivateKey**](http://docs.google.com/java/security/PrivateKey.html) | A private key. |
| [**PrivilegedAction<T>**](http://docs.google.com/java/security/PrivilegedAction.html) | A computation to be performed with privileges enabled. |
| [**PrivilegedExceptionAction<T>**](http://docs.google.com/java/security/PrivilegedExceptionAction.html) | A computation to be performed with privileges enabled, that throws one or more checked exceptions. |
| [**PublicKey**](http://docs.google.com/java/security/PublicKey.html) | A public key. |

| **Class Summary** | |
| --- | --- |
| [**AccessControlContext**](http://docs.google.com/java/security/AccessControlContext.html) | An AccessControlContext is used to make system resource access decisions based on the context it encapsulates. |
| [**AccessController**](http://docs.google.com/java/security/AccessController.html) | The AccessController class is used for access control operations and decisions. |
| [**AlgorithmParameterGenerator**](http://docs.google.com/java/security/AlgorithmParameterGenerator.html) | The AlgorithmParameterGenerator class is used to generate a set of parameters to be used with a certain algorithm. |
| [**AlgorithmParameterGeneratorSpi**](http://docs.google.com/java/security/AlgorithmParameterGeneratorSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the AlgorithmParameterGenerator class, which is used to generate a set of parameters to be used with a certain algorithm. |
| [**AlgorithmParameters**](http://docs.google.com/java/security/AlgorithmParameters.html) | This class is used as an opaque representation of cryptographic parameters. |
| [**AlgorithmParametersSpi**](http://docs.google.com/java/security/AlgorithmParametersSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the AlgorithmParameters class, which is used to manage algorithm parameters. |
| [**AllPermission**](http://docs.google.com/java/security/AllPermission.html) | The AllPermission is a permission that implies all other permissions. |
| [**AuthProvider**](http://docs.google.com/java/security/AuthProvider.html) | This class defines login and logout methods for a provider. |
| [**BasicPermission**](http://docs.google.com/java/security/BasicPermission.html) | The BasicPermission class extends the Permission class, and can be used as the base class for permissions that want to follow the same naming convention as BasicPermission. |
| [**CodeSigner**](http://docs.google.com/java/security/CodeSigner.html) | This class encapsulates information about a code signer. |
| [**CodeSource**](http://docs.google.com/java/security/CodeSource.html) | This class extends the concept of a codebase to encapsulate not only the location (URL) but also the certificate chains that were used to verify signed code originating from that location. |
| [**DigestInputStream**](http://docs.google.com/java/security/DigestInputStream.html) | A transparent stream that updates the associated message digest using the bits going through the stream. |
| [**DigestOutputStream**](http://docs.google.com/java/security/DigestOutputStream.html) | A transparent stream that updates the associated message digest using the bits going through the stream. |
| [**GuardedObject**](http://docs.google.com/java/security/GuardedObject.html) | A GuardedObject is an object that is used to protect access to another object. |
| [**Identity**](http://docs.google.com/java/security/Identity.html) | **Deprecated.** *This class is no longer used.* |
| [**IdentityScope**](http://docs.google.com/java/security/IdentityScope.html) | **Deprecated.** *This class is no longer used.* |
| [**KeyFactory**](http://docs.google.com/java/security/KeyFactory.html) | Key factories are used to convert *keys* (opaque cryptographic keys of type Key) into *key specifications* (transparent representations of the underlying key material), and vice versa. |
| [**KeyFactorySpi**](http://docs.google.com/java/security/KeyFactorySpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the KeyFactory class. |
| [**KeyPair**](http://docs.google.com/java/security/KeyPair.html) | This class is a simple holder for a key pair (a public key and a private key). |
| [**KeyPairGenerator**](http://docs.google.com/java/security/KeyPairGenerator.html) | The KeyPairGenerator class is used to generate pairs of public and private keys. |
| [**KeyPairGeneratorSpi**](http://docs.google.com/java/security/KeyPairGeneratorSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the KeyPairGenerator class, which is used to generate pairs of public and private keys. |
| [**KeyRep**](http://docs.google.com/java/security/KeyRep.html) | Standardized representation for serialized Key objects. |
| [**KeyStore**](http://docs.google.com/java/security/KeyStore.html) | This class represents a storage facility for cryptographic keys and certificates. |
| [**KeyStore.Builder**](http://docs.google.com/java/security/KeyStore.Builder.html) | A description of a to-be-instantiated KeyStore object. |
| [**KeyStore.CallbackHandlerProtection**](http://docs.google.com/java/security/KeyStore.CallbackHandlerProtection.html) | A ProtectionParameter encapsulating a CallbackHandler. |
| [**KeyStore.PasswordProtection**](http://docs.google.com/java/security/KeyStore.PasswordProtection.html) | A password-based implementation of ProtectionParameter. |
| [**KeyStore.PrivateKeyEntry**](http://docs.google.com/java/security/KeyStore.PrivateKeyEntry.html) | A KeyStore entry that holds a PrivateKey and corresponding certificate chain. |
| [**KeyStore.SecretKeyEntry**](http://docs.google.com/java/security/KeyStore.SecretKeyEntry.html) | A KeyStore entry that holds a SecretKey. |
| [**KeyStore.TrustedCertificateEntry**](http://docs.google.com/java/security/KeyStore.TrustedCertificateEntry.html) | A KeyStore entry that holds a trusted Certificate. |
| [**KeyStoreSpi**](http://docs.google.com/java/security/KeyStoreSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the KeyStore class. |
| [**MessageDigest**](http://docs.google.com/java/security/MessageDigest.html) | This MessageDigest class provides applications the functionality of a message digest algorithm, such as MD5 or SHA. |
| [**MessageDigestSpi**](http://docs.google.com/java/security/MessageDigestSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the MessageDigest class, which provides the functionality of a message digest algorithm, such as MD5 or SHA. |
| [**Permission**](http://docs.google.com/java/security/Permission.html) | Abstract class for representing access to a system resource. |
| [**PermissionCollection**](http://docs.google.com/java/security/PermissionCollection.html) | Abstract class representing a collection of Permission objects. |
| [**Permissions**](http://docs.google.com/java/security/Permissions.html) | This class represents a heterogeneous collection of Permissions. |
| [**Policy**](http://docs.google.com/java/security/Policy.html) | A Policy object is responsible for determining whether code executing in the Java runtime environment has permission to perform a security-sensitive operation. |
| [**PolicySpi**](http://docs.google.com/java/security/PolicySpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the Policy class. |
| [**ProtectionDomain**](http://docs.google.com/java/security/ProtectionDomain.html) | This ProtectionDomain class encapsulates the characteristics of a domain, which encloses a set of classes whose instances are granted a set of permissions when being executed on behalf of a given set of Principals. |
| [**Provider**](http://docs.google.com/java/security/Provider.html) | This class represents a "provider" for the Java Security API, where a provider implements some or all parts of Java Security. |
| [**Provider.Service**](http://docs.google.com/java/security/Provider.Service.html) | The description of a security service. |
| [**SecureClassLoader**](http://docs.google.com/java/security/SecureClassLoader.html) | This class extends ClassLoader with additional support for defining classes with an associated code source and permissions which are retrieved by the system policy by default. |
| [**SecureRandom**](http://docs.google.com/java/security/SecureRandom.html) | This class provides a cryptographically strong random number generator (RNG). |
| [**SecureRandomSpi**](http://docs.google.com/java/security/SecureRandomSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the SecureRandom class. |
| [**Security**](http://docs.google.com/java/security/Security.html) | This class centralizes all security properties and common security methods. |
| [**SecurityPermission**](http://docs.google.com/java/security/SecurityPermission.html) | This class is for security permissions. |
| [**Signature**](http://docs.google.com/java/security/Signature.html) | This Signature class is used to provide applications the functionality of a digital signature algorithm. |
| [**SignatureSpi**](http://docs.google.com/java/security/SignatureSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the Signature class, which is used to provide the functionality of a digital signature algorithm. |
| [**SignedObject**](http://docs.google.com/java/security/SignedObject.html) | SignedObject is a class for the purpose of creating authentic runtime objects whose integrity cannot be compromised without being detected. |
| [**Signer**](http://docs.google.com/java/security/Signer.html) | **Deprecated.** *This class is no longer used.* |
| [**Timestamp**](http://docs.google.com/java/security/Timestamp.html) | This class encapsulates information about a signed timestamp. |
| [**UnresolvedPermission**](http://docs.google.com/java/security/UnresolvedPermission.html) | The UnresolvedPermission class is used to hold Permissions that were "unresolved" when the Policy was initialized. |
| [**URIParameter**](http://docs.google.com/java/security/URIParameter.html) | A parameter that contains a URI pointing to data intended for a PolicySpi or ConfigurationSpi implementation. |

| **Enum Summary** | |
| --- | --- |
| [**KeyRep.Type**](http://docs.google.com/java/security/KeyRep.Type.html) | Key type. |

| **Exception Summary** | |
| --- | --- |
| [**AccessControlException**](http://docs.google.com/java/security/AccessControlException.html) | This exception is thrown by the AccessController to indicate that a requested access (to a critical system resource such as the file system or the network) is denied. |
| [**DigestException**](http://docs.google.com/java/security/DigestException.html) | This is the generic Message Digest exception. |
| [**GeneralSecurityException**](http://docs.google.com/java/security/GeneralSecurityException.html) | The GeneralSecurityException class is a generic security exception class that provides type safety for all the security-related exception classes that extend from it. |
| [**InvalidAlgorithmParameterException**](http://docs.google.com/java/security/InvalidAlgorithmParameterException.html) | This is the exception for invalid or inappropriate algorithm parameters. |
| [**InvalidKeyException**](http://docs.google.com/java/security/InvalidKeyException.html) | This is the exception for invalid Keys (invalid encoding, wrong length, uninitialized, etc). |
| [**InvalidParameterException**](http://docs.google.com/java/security/InvalidParameterException.html) | This exception, designed for use by the JCA/JCE engine classes, is thrown when an invalid parameter is passed to a method. |
| [**KeyException**](http://docs.google.com/java/security/KeyException.html) | This is the basic key exception. |
| [**KeyManagementException**](http://docs.google.com/java/security/KeyManagementException.html) | This is the general key management exception for all operations dealing with key management. |
| [**KeyStoreException**](http://docs.google.com/java/security/KeyStoreException.html) | This is the generic KeyStore exception. |
| [**NoSuchAlgorithmException**](http://docs.google.com/java/security/NoSuchAlgorithmException.html) | This exception is thrown when a particular cryptographic algorithm is requested but is not available in the environment. |
| [**NoSuchProviderException**](http://docs.google.com/java/security/NoSuchProviderException.html) | This exception is thrown when a particular security provider is requested but is not available in the environment. |
| [**PrivilegedActionException**](http://docs.google.com/java/security/PrivilegedActionException.html) | This exception is thrown by doPrivileged(PrivilegedExceptionAction) and doPrivileged(PrivilegedExceptionAction, AccessControlContext context) to indicate that the action being performed threw a checked exception. |
| [**ProviderException**](http://docs.google.com/java/security/ProviderException.html) | A runtime exception for Provider exceptions (such as misconfiguration errors or unrecoverable internal errors), which may be subclassed by Providers to throw specialized, provider-specific runtime errors. |
| [**SignatureException**](http://docs.google.com/java/security/SignatureException.html) | This is the generic Signature exception. |
| [**UnrecoverableEntryException**](http://docs.google.com/java/security/UnrecoverableEntryException.html) | This exception is thrown if an entry in the keystore cannot be recovered. |
| [**UnrecoverableKeyException**](http://docs.google.com/java/security/UnrecoverableKeyException.html) | This exception is thrown if a key in the keystore cannot be recovered. |

## Package java.security Description

Provides the classes and interfaces for the security framework. This includes classes that implement an easily configurable, fine-grained access control security architecture. This package also supports the generation and storage of cryptographic public key pairs, as well as a number of exportable cryptographic operations including those for message digest and signature generation. Finally, this package provides classes that support signed/guarded objects and secure random number generation. Many of the classes provided in this package (the cryptographic and secure random number generator classes in particular) are provider-based. The class itself defines a programming interface to which applications may write. The implementations themselves may then be written by independent third-party vendors and plugged in seamlessly as needed. Therefore application developers may take advantage of any number of provider-based implementations without having to add or rewrite code.

## Package Specification

* [**Cryptography Architecture**](http://docs.google.com/technotes/guides/security/crypto/CryptoSpec.html)
* PKCS8: Private-Key Information Standard, Version 1.2, November 1993

## Related Documentation

For further documentation, please see:

* [**Security Architecture**](http://docs.google.com/technotes/guides/security/spec/security-spec.doc.html)
* [**How to Implement a Provider for the Java Cryptography Architecture**](http://docs.google.com/technotes/guides/security/crypto/HowToImplAProvider.html)
* [**Default Policy Implementation and Policy File Syntax**](http://docs.google.com/technotes/guides/security/PolicyFiles.html)
* [**Policy Permissions**](http://docs.google.com/technotes/guides/security/permissions.html)
* [**Security Tools Summary**](http://docs.google.com/technotes/guides/security/SecurityToolsSummary.html)
* **keytool** ( [for Solaris/Linux](http://docs.google.com/technotes/tools/solaris/keytool.html)) ( [for Windows](http://docs.google.com/technotes/tools/windows/keytool.html))
* **jarsigner** ( [for Solaris/Linux](http://docs.google.com/technotes/tools/solaris/jarsigner.html)) ( [for Windows](http://docs.google.com/technotes/tools/windows/jarsigner.html))

**Since:** JDK1.1

| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.html) | [**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 PACKAGE**](http://docs.google.com/java/rmi/server/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/java/security/acl/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?java/security/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.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).