| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CipherInputStream.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 CLASS**](http://docs.google.com/javax/crypto/Cipher.html)   [**NEXT CLASS**](http://docs.google.com/javax/crypto/CipherOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?javax/crypto/CipherInputStream.html)    [**NO FRAMES**](http://docs.google.com/CipherInputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#26in1rg) |

## **javax.crypto**

Class CipherInputStream

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.io.InputStream](http://docs.google.com/java/io/InputStream.html)  
 [java.io.FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html)  
 **javax.crypto.CipherInputStream**

**All Implemented Interfaces:** [Closeable](http://docs.google.com/java/io/Closeable.html)

public class **CipherInputStream**extends [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html)

A CipherInputStream is composed of an InputStream and a Cipher so that read() methods return data that are read in from the underlying InputStream but have been additionally processed by the Cipher. The Cipher must be fully initialized before being used by a CipherInputStream.

For example, if the Cipher is initialized for decryption, the CipherInputStream will attempt to read in data and decrypt them, before returning the decrypted data.

This class adheres strictly to the semantics, especially the failure semantics, of its ancestor classes java.io.FilterInputStream and java.io.InputStream. This class has exactly those methods specified in its ancestor classes, and overrides them all. Moreover, this class catches all exceptions that are not thrown by its ancestor classes. In particular, the skip method skips, and the available method counts only data that have been processed by the encapsulated Cipher.

It is crucial for a programmer using this class not to use methods that are not defined or overriden in this class (such as a new method or constructor that is later added to one of the super classes), because the design and implementation of those methods are unlikely to have considered security impact with regard to CipherInputStream.

**Since:** 1.4 **See Also:**[InputStream](http://docs.google.com/java/io/InputStream.html), [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html), [Cipher](http://docs.google.com/javax/crypto/Cipher.html), [CipherOutputStream](http://docs.google.com/javax/crypto/CipherOutputStream.html)

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class java.io.**[**FilterInputStream**](http://docs.google.com/java/io/FilterInputStream.html) |
| --- |
| [in](http://docs.google.com/java/io/FilterInputStream.html#in) |

| **Constructor Summary** | |
| --- | --- |
| protected | [**CipherInputStream**](http://docs.google.com/javax/crypto/CipherInputStream.html#CipherInputStream(java.io.InputStream))([InputStream](http://docs.google.com/java/io/InputStream.html) is)            Constructs a CipherInputStream from an InputStream without specifying a Cipher. |
|  | [**CipherInputStream**](http://docs.google.com/javax/crypto/CipherInputStream.html#CipherInputStream(java.io.InputStream,%20javax.crypto.Cipher))([InputStream](http://docs.google.com/java/io/InputStream.html) is, [Cipher](http://docs.google.com/javax/crypto/Cipher.html) c)            Constructs a CipherInputStream from an InputStream and a Cipher. |

| **Method Summary** | |
| --- | --- |
| int | [**available**](http://docs.google.com/javax/crypto/CipherInputStream.html#available())()            Returns the number of bytes that can be read from this input stream without blocking. |
| void | [**close**](http://docs.google.com/javax/crypto/CipherInputStream.html#close())()            Closes this input stream and releases any system resources associated with the stream. |
| boolean | [**markSupported**](http://docs.google.com/javax/crypto/CipherInputStream.html#markSupported())()            Tests if this input stream supports the mark and reset methods, which it does not. |
| int | [**read**](http://docs.google.com/javax/crypto/CipherInputStream.html#read())()            Reads the next byte of data from this input stream. |
| int | [**read**](http://docs.google.com/javax/crypto/CipherInputStream.html#read(byte%5B%5D))(byte[] b)            Reads up to b.length bytes of data from this input stream into an array of bytes. |
| int | [**read**](http://docs.google.com/javax/crypto/CipherInputStream.html#read(byte%5B%5D,%20int,%20int))(byte[] b, int off, int len)            Reads up to len bytes of data from this input stream into an array of bytes. |
| long | [**skip**](http://docs.google.com/javax/crypto/CipherInputStream.html#skip(long))(long n)            Skips n bytes of input from the bytes that can be read from this input stream without blocking. |

| **Methods inherited from class java.io.**[**FilterInputStream**](http://docs.google.com/java/io/FilterInputStream.html) |
| --- |
| [mark](http://docs.google.com/java/io/FilterInputStream.html#mark(int)), [reset](http://docs.google.com/java/io/FilterInputStream.html#reset()) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### CipherInputStream

public **CipherInputStream**([InputStream](http://docs.google.com/java/io/InputStream.html) is,  
 [Cipher](http://docs.google.com/javax/crypto/Cipher.html) c)

Constructs a CipherInputStream from an InputStream and a Cipher.

Note: if the specified input stream or cipher is null, a NullPointerException may be thrown later when they are used.

**Parameters:**is - the to-be-processed input streamc - an initialized Cipher object

### CipherInputStream

protected **CipherInputStream**([InputStream](http://docs.google.com/java/io/InputStream.html) is)

Constructs a CipherInputStream from an InputStream without specifying a Cipher. This has the effect of constructing a CipherInputStream using a NullCipher.

Note: if the specified input stream is null, a NullPointerException may be thrown later when it is used.

**Parameters:**is - the to-be-processed input stream

| **Method Detail** |
| --- |

### read

public int **read**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Reads the next byte of data from this input stream. The value byte is returned as an int in the range 0 to 255. If no byte is available because the end of the stream has been reached, the value -1 is returned. This method blocks until input data is available, the end of the stream is detected, or an exception is thrown.

**Overrides:**[read](http://docs.google.com/java/io/FilterInputStream.html#read()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Returns:**the next byte of data, or -1 if the end of the stream is reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**Since:** JCE1.2 **See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in)

### read

public int **read**(byte[] b)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Reads up to b.length bytes of data from this input stream into an array of bytes.

The read method of InputStream calls the read method of three arguments with the arguments b, 0, and b.length.

**Overrides:**[read](http://docs.google.com/java/io/FilterInputStream.html#read(byte%5B%5D)) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Parameters:**b - the buffer into which the data is read. **Returns:**the total number of bytes read into the buffer, or -1 is there is no more data because the end of the stream has been reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**Since:** JCE1.2 **See Also:**[InputStream.read(byte[], int, int)](http://docs.google.com/java/io/InputStream.html#read(byte%5B%5D,%20int,%20int))

### read

public int **read**(byte[] b,  
 int off,  
 int len)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Reads up to len bytes of data from this input stream into an array of bytes. This method blocks until some input is available. If the first argument is null, up to len bytes are read and discarded.

**Overrides:**[read](http://docs.google.com/java/io/FilterInputStream.html#read(byte%5B%5D,%20int,%20int)) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Parameters:**b - the buffer into which the data is read.off - the start offset in the destination array buflen - the maximum number of bytes read. **Returns:**the total number of bytes read into the buffer, or -1 if there is no more data because the end of the stream has been reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**Since:** JCE1.2 **See Also:**[InputStream.read()](http://docs.google.com/java/io/InputStream.html#read())

### skip

public long **skip**(long n)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Skips n bytes of input from the bytes that can be read from this input stream without blocking.

Fewer bytes than requested might be skipped. The actual number of bytes skipped is equal to n or the result of a call to [available](http://docs.google.com/javax/crypto/CipherInputStream.html#available()), whichever is smaller. If n is less than zero, no bytes are skipped.

The actual number of bytes skipped is returned.

**Overrides:**[skip](http://docs.google.com/java/io/FilterInputStream.html#skip(long)) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Parameters:**n - the number of bytes to be skipped. **Returns:**the actual number of bytes skipped. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**Since:** JCE1.2

### available

public int **available**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Returns the number of bytes that can be read from this input stream without blocking. The available method of InputStream returns 0. This method **should** be overridden by subclasses.

**Overrides:**[available](http://docs.google.com/java/io/FilterInputStream.html#available()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Returns:**the number of bytes that can be read from this input stream without blocking. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**Since:** JCE1.2

### close

public void **close**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Closes this input stream and releases any system resources associated with the stream.

The close method of CipherInputStream calls the close method of its underlying input stream.

**Specified by:**[close](http://docs.google.com/java/io/Closeable.html#close()) in interface [Closeable](http://docs.google.com/java/io/Closeable.html)**Overrides:**[close](http://docs.google.com/java/io/FilterInputStream.html#close()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**Since:** JCE1.2 **See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in)

### markSupported

public boolean **markSupported**()

Tests if this input stream supports the mark and reset methods, which it does not.

**Overrides:**[markSupported](http://docs.google.com/java/io/FilterInputStream.html#markSupported()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Returns:**false, since this class does not support the mark and reset methods.**Since:** JCE1.2 **See Also:**[InputStream.mark(int)](http://docs.google.com/java/io/InputStream.html#mark(int)), [InputStream.reset()](http://docs.google.com/java/io/InputStream.html#reset())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CipherInputStream.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 CLASS**](http://docs.google.com/javax/crypto/Cipher.html)   [**NEXT CLASS**](http://docs.google.com/javax/crypto/CipherOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?javax/crypto/CipherInputStream.html)    [**NO FRAMES**](http://docs.google.com/CipherInputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#26in1rg) |

[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).