| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/RMISocketFactory.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/java/rmi/server/RMIServerSocketFactory.html)   [**NEXT CLASS**](http://docs.google.com/java/rmi/server/ServerCloneException.html) | [**FRAMES**](http://docs.google.com/index.html?java/rmi/server/RMISocketFactory.html)    [**NO FRAMES**](http://docs.google.com/RMISocketFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **java.rmi.server**

Class RMISocketFactory

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.rmi.server.RMISocketFactory**

**All Implemented Interfaces:** [RMIClientSocketFactory](http://docs.google.com/java/rmi/server/RMIClientSocketFactory.html), [RMIServerSocketFactory](http://docs.google.com/java/rmi/server/RMIServerSocketFactory.html)

public abstract class **RMISocketFactory**extends [Object](http://docs.google.com/java/lang/Object.html)implements [RMIClientSocketFactory](http://docs.google.com/java/rmi/server/RMIClientSocketFactory.html), [RMIServerSocketFactory](http://docs.google.com/java/rmi/server/RMIServerSocketFactory.html)

An RMISocketFactory instance is used by the RMI runtime in order to obtain client and server sockets for RMI calls. An application may use the setSocketFactory method to request that the RMI runtime use its socket factory instance instead of the default implementation.

The default socket factory implementation used goes through a three-tiered approach to creating client sockets. First, a direct socket connection to the remote VM is attempted. If that fails (due to a firewall), the runtime uses HTTP with the explicit port number of the server. If the firewall does not allow this type of communication, then HTTP to a cgi-bin script on the server is used to POST the RMI call.

**Since:** JDK1.1

| **Constructor Summary** | |
| --- | --- |
| [**RMISocketFactory**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#RMISocketFactory())()            Constructs an RMISocketFactory. |

| **Method Summary** | |
| --- | --- |
| abstract  [ServerSocket](http://docs.google.com/java/net/ServerSocket.html) | [**createServerSocket**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#createServerSocket(int))(int port)            Create a server socket on the specified port (port 0 indicates an anonymous port). |
| abstract  [Socket](http://docs.google.com/java/net/Socket.html) | [**createSocket**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#createSocket(java.lang.String,%20int))([String](http://docs.google.com/java/lang/String.html) host, int port)            Creates a client socket connected to the specified host and port. |
| static [RMISocketFactory](http://docs.google.com/java/rmi/server/RMISocketFactory.html) | [**getDefaultSocketFactory**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#getDefaultSocketFactory())()            Returns a reference to the default socket factory used by this RMI implementation. |
| static [RMIFailureHandler](http://docs.google.com/java/rmi/server/RMIFailureHandler.html) | [**getFailureHandler**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#getFailureHandler())()            Returns the handler for socket creation failure set by the setFailureHandler method. |
| static [RMISocketFactory](http://docs.google.com/java/rmi/server/RMISocketFactory.html) | [**getSocketFactory**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#getSocketFactory())()            Returns the socket factory set by the setSocketFactory method. |
| static void | [**setFailureHandler**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#setFailureHandler(java.rmi.server.RMIFailureHandler))([RMIFailureHandler](http://docs.google.com/java/rmi/server/RMIFailureHandler.html) fh)            Sets the failure handler to be called by the RMI runtime if server socket creation fails. |
| static void | [**setSocketFactory**](http://docs.google.com/java/rmi/server/RMISocketFactory.html#setSocketFactory(java.rmi.server.RMISocketFactory))([RMISocketFactory](http://docs.google.com/java/rmi/server/RMISocketFactory.html) fac)            Set the global socket factory from which RMI gets sockets (if the remote object is not associated with a specific client and/or server socket factory). |

| **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** |
| --- |

### RMISocketFactory

public **RMISocketFactory**()

Constructs an RMISocketFactory.

**Since:** JDK1.1

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

### createSocket

public abstract [Socket](http://docs.google.com/java/net/Socket.html) **createSocket**([String](http://docs.google.com/java/lang/String.html) host,  
 int port)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Creates a client socket connected to the specified host and port.

**Specified by:**[createSocket](http://docs.google.com/java/rmi/server/RMIClientSocketFactory.html#createSocket(java.lang.String,%20int)) in interface [RMIClientSocketFactory](http://docs.google.com/java/rmi/server/RMIClientSocketFactory.html) **Parameters:**host - the host nameport - the port number **Returns:**a socket connected to the specified host and port. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs during socket creation**Since:** JDK1.1

### createServerSocket

public abstract [ServerSocket](http://docs.google.com/java/net/ServerSocket.html) **createServerSocket**(int port)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Create a server socket on the specified port (port 0 indicates an anonymous port).

**Specified by:**[createServerSocket](http://docs.google.com/java/rmi/server/RMIServerSocketFactory.html#createServerSocket(int)) in interface [RMIServerSocketFactory](http://docs.google.com/java/rmi/server/RMIServerSocketFactory.html) **Parameters:**port - the port number **Returns:**the server socket on the specified port **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs during server socket creation**Since:** JDK1.1

### setSocketFactory

public static void **setSocketFactory**([RMISocketFactory](http://docs.google.com/java/rmi/server/RMISocketFactory.html) fac)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Set the global socket factory from which RMI gets sockets (if the remote object is not associated with a specific client and/or server socket factory). The RMI socket factory can only be set once. Note: The RMISocketFactory may only be set if the current security manager allows setting a socket factory; if disallowed, a SecurityException will be thrown.

**Parameters:**fac - the socket factory **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if the RMI socket factory is already set [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and its checkSetFactory method doesn't allow the operation.**Since:** JDK1.1 **See Also:**[getSocketFactory()](http://docs.google.com/java/rmi/server/RMISocketFactory.html#getSocketFactory()), [SecurityManager.checkSetFactory()](http://docs.google.com/java/lang/SecurityManager.html#checkSetFactory())

### getSocketFactory

public static [RMISocketFactory](http://docs.google.com/java/rmi/server/RMISocketFactory.html) **getSocketFactory**()

Returns the socket factory set by the setSocketFactory method. Returns null if no socket factory has been set.

**Returns:**the socket factory**Since:** JDK1.1 **See Also:**[setSocketFactory(RMISocketFactory)](http://docs.google.com/java/rmi/server/RMISocketFactory.html#setSocketFactory(java.rmi.server.RMISocketFactory))

### getDefaultSocketFactory

public static [RMISocketFactory](http://docs.google.com/java/rmi/server/RMISocketFactory.html) **getDefaultSocketFactory**()

Returns a reference to the default socket factory used by this RMI implementation. This will be the factory used by the RMI runtime when getSocketFactory returns null.

**Returns:**the default RMI socket factory**Since:** JDK1.1

### setFailureHandler

public static void **setFailureHandler**([RMIFailureHandler](http://docs.google.com/java/rmi/server/RMIFailureHandler.html) fh)

Sets the failure handler to be called by the RMI runtime if server socket creation fails. By default, if no failure handler is installed and server socket creation fails, the RMI runtime does attempt to recreate the server socket.

If there is a security manager, this method first calls the security manager's checkSetFactory method to ensure the operation is allowed. This could result in a SecurityException.

**Parameters:**fh - the failure handler **Throws:** [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and its checkSetFactory method doesn't allow the operation.**Since:** JDK1.1 **See Also:**[getFailureHandler()](http://docs.google.com/java/rmi/server/RMISocketFactory.html#getFailureHandler()), [RMIFailureHandler.failure(Exception)](http://docs.google.com/java/rmi/server/RMIFailureHandler.html#failure(java.lang.Exception))

### getFailureHandler

public static [RMIFailureHandler](http://docs.google.com/java/rmi/server/RMIFailureHandler.html) **getFailureHandler**()

Returns the handler for socket creation failure set by the setFailureHandler method.

**Returns:**the failure handler**Since:** JDK1.1 **See Also:**[setFailureHandler(RMIFailureHandler)](http://docs.google.com/java/rmi/server/RMISocketFactory.html#setFailureHandler(java.rmi.server.RMIFailureHandler))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/RMISocketFactory.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/java/rmi/server/RMIServerSocketFactory.html)   [**NEXT CLASS**](http://docs.google.com/java/rmi/server/ServerCloneException.html) | [**FRAMES**](http://docs.google.com/index.html?java/rmi/server/RMISocketFactory.html)    [**NO FRAMES**](http://docs.google.com/RMISocketFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).