

# SocketException

## SocketException

Low level socket exceptions. Some diagnostics may be provided.

### Host is Down

Example:

```
java.io.IOException: Failed on local exception: java.net.SocketException: Host is down; Host Details : local
host is: "client1.example.org/192.168.1.86"; destination host is: "hdfs.example.org":8020;
    at org.apache.hadoop.net.NetUtils.wrapException(NetUtils.java:772)
    at org.apache.hadoop.ipc.Client.call(Client.java:1472)
    at org.apache.hadoop.ipc.Client.call(Client.java:1399)
    at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:232)
    at com.sun.proxy.$Proxy22.getFileInfo(Unknown Source)
    at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.getFileInfo
(ClientNamenodeProtocolTranslatorPB.java:752)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:606)
    at org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:187)
    at org.apache.hadoop.io.retry.RetryInvocationHandler.invoke(RetryInvocationHandler.java:102)
    at com.sun.proxy.$Proxy23.getFileInfo(Unknown Source)
    at org.apache.hadoop.hdfs.DFSClient.getFileInfo(DFSClient.java:1988)
    at org.apache.hadoop.hdfs.DistributedFileSystem$18.doCall(DistributedFileSystem.java:1118)
    at org.apache.hadoop.hdfs.DistributedFileSystem$18.doCall(DistributedFileSystem.java:1114)
    at org.apache.hadoop.fs.FileSystemLinkResolver.resolve(FileSystemLinkResolver.java:81)
    at org.apache.hadoop.hdfs.DistributedFileSystem.getFileStatus(DistributedFileSystem.java:1114)
```

What does this stack trace tell us?

1. It's a "local exception": it happened on the local host
2. "the Host is down": the network layer on the local host believes that the destination server is not there.
3. the destination host is "hdfs.example.org":8020 . This is the host to look for
4. The exception is triggered by an HDFS call. (see `org.apache.hadoop.hdfs` at the bottom of the stack trace).

That information is enough to hint to us that an HDFS operation is failing as the HDFS server "hdfs.example.org" is down.

It's not guaranteed to be the cause, as there could be other reasons, including configuration ones

1. The URI to HDFS, as set in `core-site.xml` could be wrong; the client trying to talk to the wrong host—one that is down.
2. The IP address of the host, as set in DNS or `/etc/hosts` is wrong. The client is trying to talk to a machine at the wrong IP address, a

machine that the network stack thinks is down.

### Connection Reset

The connection was reset at the TCP layer

There is good coverage of this issue on <http://stackoverflow.com/questions/62929/java-net-socketexception-connection-reset>

Remember: These are [your network configuration problems](#) . Only you can fix them.

### Permission denied

This can arise if the service is configured to listen on a port numbered less than 1024, but is not running as a user with the appropriate permissions.

```

2016-03-22 15:26:18,905 ERROR org.apache.hadoop.hdfs.server.datanode.DataNode: Exception in secureMain
java.net.SocketException: Permission denied
    at sun.nio.ch.Net.bind0(Native Method)
    at sun.nio.ch.Net.bind(Net.java:433)
    at sun.nio.ch.Net.bind(Net.java:425)
    at sun.nio.ch.ServerSocketChannelImpl.bind(ServerSocketChannelImpl.java:223)
    at sun.nio.ch.ServerSocketAdaptor.bind(ServerSocketAdaptor.java:74)
    at io.netty.channel.socket.nio.NioServerSocketChannel.doBind(NioServerSocketChannel.java:125)
    at io.netty.channel.AbstractChannel$AbstractUnsafe.bind(AbstractChannel.java:522)
    at io.netty.channel.DefaultChannelPipeline$HeadContext.bind(DefaultChannelPipeline.java:1196)
    at io.netty.channel.ChannelHandlerInvokerUtil.invokeBindNow(ChannelHandlerInvokerUtil.java:108)
    at io.netty.channel.DefaultChannelHandlerInvoker.invokeBind(DefaultChannelHandlerInvoker.java:214)
    at io.netty.channel.AbstractChannelHandlerContext.bind(AbstractChannelHandlerContext.java:208)
    at io.netty.channel.DefaultChannelPipeline.bind(DefaultChannelPipeline.java:1003)
    at io.netty.channel.AbstractChannel.bind(AbstractChannel.java:216)
    at io.netty.bootstrap.AbstractBootstrap$2.run(AbstractBootstrap.java:357)
    at io.netty.util.concurrent.SingleThreadEventExecutor.runAllTasks(SingleThreadEventExecutor.java:322)
    at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:356)
    at io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:703)
    at io.netty.util.concurrent.DefaultThreadFactory$DefaultRunnableDecorator.run(DefaultThreadFactory.java:
137)
        at java.lang.Thread.run(Thread.java:745)
2016-03-22 15:26:18,907 INFO org.apache.hadoop.util.ExitUtil: Exiting with status 1
2016-03-22 15:26:18,908 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: SHUTDOWN_MSG:
/*****

```

Fixes: either run the service (here, the Datanode) as a user with permissions, or change the service configuration to use a higher numbered port.