[HDFS-4404](https://www.google.com/url?q=https://issues.apache.org/jira/browse/HDFS-4404&sa=D&source=editors&ust=1751359155543751&usg=AOvVaw0e6JDg_1sqvAkiytsUAwok)

Create file failure when the machine of first attempted NameNode is down

**(1)  Log information**

**(1.1) Roles in this case**

 DFSClient (client-side)           NameNode (server-side)

**(1.2) Symptoms**

Description in the bug report:

test Environment: NN1,NN2,DN1,DN2,DN3

machine1:NN1,DN1                       machine2:NN2,DN2                       machine3:DN3

machine1 is down.

 (from the comments: “If you bring down a namenode and pretty much all clients that create files fail. That to me means file system appears to be down and not available”)

The logs are from DFSClient.

2013-01-12 09:51:21,248 DEBUG ipc.Client (Client.java:setupIOstreams(562)) - Connecting to /160.161.0.155:8020

The ipc client on DFSClient side tries to connect NameNode(NN)1.

2013-01-12 09:51:38,442 DEBUG ipc.Client (Client.java:close(932)) - closing ipc connection to vm2/160.161.0.155:8020: 10000 millis timeout while waiting for channel to be ready for connect. ch : java.nio.channels.SocketChannel[connection-pending remote=/160.161.0.155:8020]

java.net.SocketTimeoutException: 10000 millis timeout while waiting for channel to be ready for connect. ch : java.nio.channels.SocketChannel[connection-pending remote=/160.161.0.155:8020]

at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:213)

at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:524)

at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:489)

at org.apache.hadoop.ipc.Client$Connection.setupConnection(Client.java:474)

at org.apache.hadoop.ipc.Client$Connection.setupIOstreams(Client.java:568)

at org.apache.hadoop.ipc.Client$Connection.access$2000(Client.java:217)

at org.apache.hadoop.ipc.Client.getConnection(Client.java:1286)

at org.apache.hadoop.ipc.Client.call(Client.java:1156)

at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:184)

at $Proxy9.create(Unknown Source)

at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.create(ClientNamenodeProtocolTranslatorPB.java:187)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)

at java.lang.reflect.Method.invoke(Method.java:597)

at org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:165)

at org.apache.hadoop.io.retry.**RetryInvocationHandler**.invoke(RetryInvocationHandler.java:84)

at $Proxy10.create(Unknown Source)

at org.apache.hadoop.hdfs.DFSOutputStream.<init>(DFSOutputStream.java:1261)

at org.apache.hadoop.hdfs.DFSOutputStream.newStreamForCreate(DFSOutputStream.java:1280)

at org.apache.hadoop.hdfs.DFSClient.create(DFSClient.java:1128)

at org.apache.hadoop.hdfs.DFSClient.create(DFSClient.java:1086)

at org.apache.hadoop.hdfs.DistributedFileSystem.create(DistributedFileSystem.java:232)

at org.apache.hadoop.hdfs.DistributedFileSystem.create(DistributedFileSystem.java:75)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:806)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:787)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:715)

at test.TestLease.main(TestLease.java:45)

Since machine1 is down(NN1 is running on it), the ipc connection times out and waits to connect again. From the call-stack, we know that this ipc communication is for creating file (FileSystem.create()).

2013-01-12 09:51:38,443 DEBUG ipc.Client (Client.java:close(940)) - IPC Client (31594013) connection to /160.161.0.155:8020 from hdfs/hadoop@HADOOP.COM: closed

IPC client still fails to connect NN1.

2013-01-12 09:52:47,834 WARN retry.RetryInvocationHandler (RetryInvocationHandler.java:invoke(95)) - Exception while invoking class org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.create. Not retrying because the invoked method is not idempotent, and unable to determine whether it was invoked

java.net.**SocketTimeoutException**: Call From szxy1x001833091/172.0.0.13 to vm2:8020 failed on socket timeout exception: java.net.SocketTimeoutException: 10000 millis timeout while waiting for channel to be ready for connect. ch : java.nio.channels.SocketChannel[connection-pending remote=/160.161.0.155:8020]; For more details see:[http://wiki.apache.org/hadoop/SocketTimeout](https://www.google.com/url?q=http://wiki.apache.org/hadoop/SocketTimeout&sa=D&source=editors&ust=1751359155554347&usg=AOvVaw3gbhLSj61nnRKsHHOkF7ca)

at org.apache.hadoop.net.NetUtils.wrapException(NetUtils.java:743)

at org.apache.hadoop.ipc.Client.call(Client.java:1180)

at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:184)

at $Proxy9.create(Unknown Source)

at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.create(ClientNamenodeProtocolTranslatorPB.java:187)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)

at java.lang.reflect.Method.invoke(Method.java:597)

at org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:165)

at org.apache.hadoop.io.retry.**RetryInvocationHandler.invoke**(RetryInvocationHandler.java:84)

at $Proxy10.create(Unknown Source)

at org.apache.hadoop.hdfs.DFSOutputStream.<init>(DFSOutputStream.java:1261)

at org.apache.hadoop.hdfs.DFSOutputStream.newStreamForCreate(DFSOutputStream.java:1280)

at org.apache.hadoop.hdfs.DFSClient.create(DFSClient.java:1128)

at org.apache.hadoop.hdfs.DFSClient.create(DFSClient.java:1086)

at org.apache.hadoop.hdfs.DistributedFileSystem.create(DistributedFileSystem.java:232)

at org.apache.hadoop.hdfs.DistributedFileSystem.create(DistributedFileSystem.java:75)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:806)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:787)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:715)

at test.TestLease.main(TestLease.java:45)

Caused by: java.net.**SocketTimeoutException**: 10000 millis timeout while waiting for channel to be ready for connect. ch : java.nio.channels.SocketChannel[connection-pending remote=/160.161.0.155:8020]

at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:213)

at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:524)

at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:489)

at org.apache.hadoop.ipc.Client$Connection.setupConnection(Client.java:474)

at org.apache.hadoop.ipc.Client$Connection.setupIOstreams(Client.java:568)

at org.apache.hadoop.ipc.Client$Connection.access$2000(Client.java:217)

at org.apache.hadoop.ipc.Client.getConnection(Client.java:1286)

at org.apache.hadoop.ipc.Client.call(Client.java:1156)

... 20 more

java.net.SocketTimeoutException: Call From szxy1x001833091/172.0.0.13 to vm2:8020 failed on socket timeout exception: java.net.**SocketTimeoutException**: 10000 millis timeout while waiting for channel to be ready for connect. ch : java.nio.channels.SocketChannel[connection-pending remote=/160.161.0.155:8020]; For more details see:[http://wiki.apache.org/hadoop/SocketTimeout](https://www.google.com/url?q=http://wiki.apache.org/hadoop/SocketTimeout&sa=D&source=editors&ust=1751359155561251&usg=AOvVaw3fhDCnizJ2TvXbi7Oj65IV)

at org.apache.hadoop.net.NetUtils.wrapException(NetUtils.java:743)

at org.apache.hadoop.ipc.Client.call(Client.java:1180)

at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:184)

at $Proxy9.create(Unknown Source)

at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.create(ClientNamenodeProtocolTranslatorPB.java:187)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)

at java.lang.reflect.Method.invoke(Method.java:597)

at org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:165)

at org.apache.hadoop.io.retry.RetryInvocationHandler.invoke(RetryInvocationHandler.java:84)

at $Proxy10.create(Unknown Source)

at org.apache.hadoop.hdfs.DFSOutputStream.<init>(DFSOutputStream.java:1261)

at org.apache.hadoop.hdfs.DFSOutputStream.newStreamForCreate(DFSOutputStream.java:1280)

at org.apache.hadoop.hdfs.DFSClient.create(DFSClient.java:1128)

at org.apache.hadoop.hdfs.DFSClient.create(DFSClient.java:1086)

at org.apache.hadoop.hdfs.DistributedFileSystem.create(DistributedFileSystem.java:232)

at org.apache.hadoop.hdfs.DistributedFileSystem.create(DistributedFileSystem.java:75)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:806)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:787)

at org.apache.hadoop.fs.FileSystem.create(FileSystem.java:715)

at test.TestLease.main(TestLease.java:45)

Caused by: java.net.SocketTimeoutException: 10000 millis timeout while waiting for channel to be ready for connect. ch : java.nio.channels.SocketChannel[connection-pending remote=/160.161.0.155:8020]

at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:213)

at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:524)

at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:489)

at org.apache.hadoop.ipc.Client$Connection.setupConnection(Client.java:474)

at org.apache.hadoop.ipc.Client$Connection.setupIOstreams(Client.java:568)

at org.apache.hadoop.ipc.Client$Connection.access$2000(Client.java:217)

at org.apache.hadoop.ipc.Client.getConnection(Client.java:1286)

at org.apache.hadoop.ipc.Client.call(Client.java:1156)

... 20 more

The call of create() from DFSClient fails finally due to the socketTimeout. Neither is create() tried again. Since “Not retrying because the invoked method is not idempotent, and unable to determine whether it was invoked”.

2013-01-12 09:54:52,269 DEBUG ipc.Client (Client.java:stop(1021)) - Stopping client

Stop a RPC client connection.  A RPC client is closed only when its reference count becomes zero.