1. 服务端没启动，客户端连接服务端，

java.net.ConnectException: Connection refused: no further information

2. 服务端已启动，客户端连接服务端超时

> 注: 客户端和服务端三次握手后，就会是已连接状态，即使服务端还没有accept。（已验证）

```text

默认30秒 `private static final int DEFAULT\_CONNECT\_TIMEOUT = 30000;`

客户端 连接超时100毫秒

bootstrap.option(ChannelOption.CONNECT\_TIMEOUT\_MILLIS, 100)

bootstrap.connect("www.gnu.org", 80)

异常

16:34:26.189 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x24bb2319] REGISTERED

16:34:26.190 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x24bb2319] CONNECT: www.gnu.org/209.51.188.116:80

16:34:26.291 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x24bb2319] CLOSE

16:34:26.293 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x24bb2319] UNREGISTERED

Exception in thread "main" io.netty.channel.ConnectTimeoutException: connection timed out: www.gnu.org/209.51.188.116:80

客户端 连接超时1000毫秒

bootstrap.option(ChannelOption.CONNECT\_TIMEOUT\_MILLIS, 1000)

连接成功

16:33:43.278 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xafc9fb6c] REGISTERED

16:33:43.278 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xafc9fb6c] CONNECT: www.gnu.org/209.51.188.116:80

16:33:43.512 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xafc9fb6c, L:/192.168.1.102:65387 - R:www.gnu.org/209.51.188.116:80] ACTIVE

```

3. 服务端，accept队列已满，连接拒绝

```text

服务端

.option(ChannelOption.SO\_BACKLOG, 2)

服务端根据 SelectionKey.OP\_ACCEPT find usages，找到该代码，进行断点，默认服务端没有accept客户端

debug server

```java

if ((readyOps & (SelectionKey.OP\_READ | SelectionKey.OP\_ACCEPT)) != 0 || readyOps == 0) {

unsafe.read();

}

客户端 设置 允许多个实例

依次非debug模式 运行第一个

16:46:32.912 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xebb258eb] REGISTERED

16:46:32.912 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xebb258eb] CONNECT: localhost/127.0.0.1:8080

16:46:32.917 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xebb258eb, L:/127.0.0.1:50553 - R:localhost/127.0.0.1:8080] ACTIVE

第二个

16:48:17.032 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x7a538be6] REGISTERED

16:48:17.032 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x7a538be6] CONNECT: localhost/127.0.0.1:8080

16:48:17.036 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x7a538be6, L:/127.0.0.1:50691 - R:localhost/127.0.0.1:8080] ACTIVE

第三个 Connection refused 连接拒绝 （服务端机器内核维护的一个accept queue，已满，内核会认为应用忙不过来，所以连接拒绝）

16:50:53.501 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xcadf1b3e] REGISTERED

16:50:53.502 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xcadf1b3e] CONNECT: localhost/127.0.0.1:8080

16:50:55.506 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xcadf1b3e] CLOSE

16:50:55.507 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xcadf1b3e] UNREGISTERED

Exception in thread "main" io.netty.channel.AbstractChannel$AnnotatedConnectException: Connection refused: no further information: localhost/127.0.0.1:8080

Caused by: java.net.ConnectException: Connection refused: no further information

4. 服务端已启动，客户端发送请求，长时间未得到服务端响应，调用超时

io.netty.handler.timeout.ReadTimeoutException: null

客户端 设置5秒读取超时

ch.pipeline().addLast(new ReadTimeoutHandler(5000, TimeUnit.MILLISECONDS));

17:14:50.605 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x29411827] REGISTERED

17:14:50.605 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x29411827] CONNECT: localhost/127.0.0.1:8080

17:14:50.609 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x29411827, L:/127.0.0.1:60672 - R:localhost/127.0.0.1:8080] ACTIVE

---- 5秒后 ----

17:14:55.614 [nioEventLoopGroup-2-1] WARN i.n.channel.DefaultChannelPipeline - An exceptionCaught() event was fired, and it reached at the tail of the pipeline. It usually means the last handler in the pipeline did not handle the exception.

io.netty.handler.timeout.ReadTimeoutException: null

17:14:55.616 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x29411827, L:/127.0.0.1:60672 - R:localhost/127.0.0.1:8080] CLOSE

17:14:55.617 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x29411827, L:/127.0.0.1:60672 ! R:localhost/127.0.0.1:8080] INACTIVE

17:14:55.617 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x29411827, L:/127.0.0.1:60672 ! R:localhost/127.0.0.1:8080] UNREGISTERED

5. 服务端进程打开文件数过多

linux

$ rz 上传 netty-server.jar

文件打开数 临时改为 18 (服务端启动后需要18个，刚好剩2个)

$ ulimit -n 18

$ ulimit -a 验证

$ java -cp netty-server.jar org.example.exception.Server

$ jps

1686 Jps

1657 Server

$ cat /proc/1657/limits

Limit Soft Limit Hard Limit Units

Max cpu time unlimited unlimited seconds

Max file size unlimited unlimited bytes

Max data size unlimited unlimited bytes

Max stack size 8388608 unlimited bytes

Max core file size 0 unlimited bytes

Max resident set unlimited unlimited bytes

Max processes 3795 3795 processes

Max open files 18 18 files

Max locked memory 65536 65536 bytes

Max address space unlimited unlimited bytes

Max file locks unlimited unlimited locks

Max pending signals 3795 3795 signals

Max msgqueue size 819200 819200 bytes

Max nice priority 0 0

Max realtime priority 0 0

Max realtime timeout unlimited unlimited us

$ ll /proc/1657/fd

total 0

lrwx------. 1 root root 64 Apr 12 18:50 0 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:50 1 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:50 10 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:50 11 -> pipe:[32869]

l-wx------. 1 root root 64 Apr 12 18:50 12 -> pipe:[32869]

lrwx------. 1 root root 64 Apr 12 18:50 13 -> anon\_inode:[eventpoll]

lrwx------. 1 root root 64 Apr 12 18:50 14 -> socket:[32873]

lrwx------. 1 root root 64 Apr 12 18:50 15 -> socket:[32884]

lrwx------. 1 root root 64 Apr 12 18:50 2 -> /dev/pts/0

lr-x------. 1 root root 64 Apr 12 18:50 3 -> /rocketmq/jdk1.8.0\_351/jre/lib/rt.jar

lr-x------. 1 root root 64 Apr 12 18:50 4 -> /root/netty-server.jar

lr-x------. 1 root root 64 Apr 12 18:50 5 -> pipe:[32867]

l-wx------. 1 root root 64 Apr 12 18:50 6 -> pipe:[32867]

lrwx------. 1 root root 64 Apr 12 18:50 7 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:50 8 -> pipe:[32868]

l-wx------. 1 root root 64 Apr 12 18:50 9 -> pipe:[32868]

$ ll /proc/1657/fd | wc -l

17

( -l, --lines print the newline counts 数多少行)

(total 0 减掉 -1)

最大索引为15，此时共打开16个文件，还剩下2个

IDEA 客户端连接服务端

bootstrap.connect("192.168.1.206", 8080) 去掉ReadTimeoutHandler

1. 第一个客户端

服务端日志

18:52:20.772 [nioEventLoopGroup-3-1] DEBUG i.n.handler.logging.LoggingHandler -- [id: 0x2643646e, L:/192.168.1.206:8080 - R:/192.168.1.102:56603] REGISTERED

18:52:20.772 [nioEventLoopGroup-3-1] DEBUG i.n.handler.logging.LoggingHandler -- [id: 0x2643646e, L:/192.168.1.206:8080 - R:/192.168.1.102:56603] ACTIVE

多了一个 16 -> socket:[33272]

[root@centos /root]# ll /proc/1657/fd

total 0

lrwx------. 1 root root 64 Apr 12 18:50 0 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:50 1 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:50 10 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:50 11 -> pipe:[32869]

l-wx------. 1 root root 64 Apr 12 18:50 12 -> pipe:[32869]

lrwx------. 1 root root 64 Apr 12 18:50 13 -> anon\_inode:[eventpoll]

lrwx------. 1 root root 64 Apr 12 18:50 14 -> socket:[32873]

lrwx------. 1 root root 64 Apr 12 18:50 15 -> socket:[32884]

lrwx------. 1 root root 64 Apr 12 18:52 16 -> socket:[33272]

lrwx------. 1 root root 64 Apr 12 18:50 2 -> /dev/pts/0

lr-x------. 1 root root 64 Apr 12 18:50 3 -> /rocketmq/jdk1.8.0\_351/jre/lib/rt.jar

lr-x------. 1 root root 64 Apr 12 18:50 4 -> /root/netty-server.jar

lr-x------. 1 root root 64 Apr 12 18:50 5 -> pipe:[32867]

l-wx------. 1 root root 64 Apr 12 18:50 6 -> pipe:[32867]

lrwx------. 1 root root 64 Apr 12 18:50 7 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:50 8 -> pipe:[32868]

l-wx------. 1 root root 64 Apr 12 18:50 9 -> pipe:[32868]

2. 启动第二个客户端

出现太多打开文件异常，当貌似还是连上了

java.io.IOException: Too many open files

多了 17 -> socket:[33914]

18:53:23.225 [nioEventLoopGroup-2-1] WARN i.n.channel.DefaultChannelPipeline -- An exceptionCaught() event was fired, and it reached at the tail of the pipeline. It usually means the last handler in the pipeline did not handle the exception.

java.io.IOException: Too many open files

at sun.nio.ch.ServerSocketChannelImpl.accept0(Native Method)

at sun.nio.ch.ServerSocketChannelImpl.accept(ServerSocketChannelImpl.java:424)

at sun.nio.ch.ServerSocketChannelImpl.accept(ServerSocketChannelImpl.java:252)

at io.netty.util.internal.SocketUtils$5.run(SocketUtils.java:119)

at io.netty.util.internal.SocketUtils$5.run(SocketUtils.java:116)

at java.security.AccessController.doPrivileged(Native Method)

at io.netty.util.internal.SocketUtils.accept(SocketUtils.java:116)

at io.netty.channel.socket.nio.NioServerSocketChannel.doReadMessages(NioServerSocketChannel.java:154)

at io.netty.channel.nio.AbstractNioMessageChannel$NioMessageUnsafe.read(AbstractNioMessageChannel.java:79)

at io.netty.channel.nio.NioEventLoop.processSelectedKey(NioEventLoop.java:788)

at io.netty.channel.nio.NioEventLoop.processSelectedKeysOptimized(NioEventLoop.java:724)

at io.netty.channel.nio.NioEventLoop.processSelectedKeys(NioEventLoop.java:650)

at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:562)

at io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)

at io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)

at io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)

at java.lang.Thread.run(Thread.java:750)

18:53:23.229 [nioEventLoopGroup-3-2] DEBUG i.n.handler.logging.LoggingHandler -- [id: 0xdd8abe2a, L:/192.168.1.206:8080 - R:/192.168.1.102:56839] REGISTERED

18:53:23.229 [nioEventLoopGroup-3-2] DEBUG i.n.handler.logging.LoggingHandler -- [id: 0xdd8abe2a, L:/192.168.1.206:8080 - R:/192.168.1.102:56839] ACTIVE

$ ll /proc/1749/fd

total 0

lrwx------. 1 root root 64 Apr 12 18:58 0 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:58 1 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:58 10 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:58 11 -> pipe:[33820]

l-wx------. 1 root root 64 Apr 12 18:58 12 -> pipe:[33820]

lrwx------. 1 root root 64 Apr 12 18:58 13 -> anon\_inode:[eventpoll]

lrwx------. 1 root root 64 Apr 12 18:58 14 -> socket:[33824]

lrwx------. 1 root root 64 Apr 12 18:58 15 -> socket:[33835]

lrwx------. 1 root root 64 Apr 12 18:58 16 -> socket:[33870]

lrwx------. 1 root root 64 Apr 12 18:58 17 -> socket:[33914]

lrwx------. 1 root root 64 Apr 12 18:58 2 -> /dev/pts/0

lr-x------. 1 root root 64 Apr 12 18:58 3 -> /rocketmq/jdk1.8.0\_351/jre/lib/rt.jar

lr-x------. 1 root root 64 Apr 12 18:58 4 -> /root/netty-server.jar

lr-x------. 1 root root 64 Apr 12 18:58 5 -> pipe:[33818]

l-wx------. 1 root root 64 Apr 12 18:58 6 -> pipe:[33818]

lrwx------. 1 root root 64 Apr 12 18:58 7 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:58 8 -> pipe:[33819]

l-wx------. 1 root root 64 Apr 12 18:58 9 -> pipe:[33819]

3. 启动第三个客户端，客户端一切正常，服务端不正常

客户端日志

18:58:49.597 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x53104b66] REGISTERED

18:58:49.597 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x53104b66] CONNECT: /192.168.1.206:8080

18:58:49.601 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0x53104b66, L:/192.168.1.102:57378 - R:/192.168.1.206:8080] ACTIVE

服务端日志 约每秒发一条警告 太多打开文件

$ ll /proc/1749/fd

没有增加额外的打开文件描述符，还是18个

$ ll /proc/1749/fd

total 0

lrwx------. 1 root root 64 Apr 12 18:58 0 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:58 1 -> /dev/pts/0

lrwx------. 1 root root 64 Apr 12 18:58 10 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:58 11 -> pipe:[33820]

l-wx------. 1 root root 64 Apr 12 18:58 12 -> pipe:[33820]

lrwx------. 1 root root 64 Apr 12 18:58 13 -> anon\_inode:[eventpoll]

lrwx------. 1 root root 64 Apr 12 18:58 14 -> socket:[33824]

lrwx------. 1 root root 64 Apr 12 18:58 15 -> socket:[33835]

lrwx------. 1 root root 64 Apr 12 18:58 16 -> socket:[33870]

lrwx------. 1 root root 64 Apr 12 18:58 17 -> socket:[33914]

lrwx------. 1 root root 64 Apr 12 18:58 2 -> /dev/pts/0

lr-x------. 1 root root 64 Apr 12 18:58 3 -> /rocketmq/jdk1.8.0\_351/jre/lib/rt.jar

lr-x------. 1 root root 64 Apr 12 18:58 4 -> /root/netty-server.jar

lr-x------. 1 root root 64 Apr 12 18:58 5 -> pipe:[33818]

l-wx------. 1 root root 64 Apr 12 18:58 6 -> pipe:[33818]

lrwx------. 1 root root 64 Apr 12 18:58 7 -> anon\_inode:[eventpoll]

lr-x------. 1 root root 64 Apr 12 18:58 8 -> pipe:[33819]

l-wx------. 1 root root 64 Apr 12 18:58 9 -> pipe:[33819]

19:00:37.843 [nioEventLoopGroup-2-1] WARN i.n.channel.DefaultChannelPipeline -- An exceptionCaught() event was fired, and it reached at the tail of the pipeline. It usually means the last handler in the pipeline did not handle the exception.

java.io.IOException: Too many open files

at sun.nio.ch.ServerSocketChannelImpl.accept0(Native Method)

at sun.nio.ch.ServerSocketChannelImpl.accept(ServerSocketChannelImpl.java:424)

at sun.nio.ch.ServerSocketChannelImpl.accept(ServerSocketChannelImpl.java:252)

at io.netty.util.internal.SocketUtils$5.run(SocketUtils.java:119)

at io.netty.util.internal.SocketUtils$5.run(SocketUtils.java:116)

at java.security.AccessController.doPrivileged(Native Method)

at io.netty.util.internal.SocketUtils.accept(SocketUtils.java:116)

at io.netty.channel.socket.nio.NioServerSocketChannel.doReadMessages(NioServerSocketChannel.java:154)

at io.netty.channel.nio.AbstractNioMessageChannel$NioMessageUnsafe.read(AbstractNioMessageChannel.java:79)

at io.netty.channel.nio.NioEventLoop.processSelectedKey(NioEventLoop.java:788)

at io.netty.channel.nio.NioEventLoop.processSelectedKeysOptimized(NioEventLoop.java:724)

at io.netty.channel.nio.NioEventLoop.processSelectedKeys(NioEventLoop.java:650)

at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:562)

at io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)

at io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)

at io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)

at java.lang.Thread.run(Thread.java:750)

4. 启动第四个客户端，并往服务端写数据，客户端一切正常

linux没增加文件描述符 $ ll /proc/1749/fd

依然总共18个

客户端日志

19:03:29.579 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xbcbe2182] REGISTERED

19:03:29.579 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xbcbe2182] CONNECT: /192.168.1.206:8080

19:03:29.583 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xbcbe2182, L:/192.168.1.102:57680 - R:/192.168.1.206:8080] ACTIVE

19:03:29.587 [nioEventLoopGroup-2-1] DEBUG io.netty.util.Recycler - -Dio.netty.recycler.maxCapacityPerThread: 4096

19:03:29.587 [nioEventLoopGroup-2-1] DEBUG io.netty.util.Recycler - -Dio.netty.recycler.ratio: 8

19:03:29.587 [nioEventLoopGroup-2-1] DEBUG io.netty.util.Recycler - -Dio.netty.recycler.chunkSize: 32

19:03:29.587 [nioEventLoopGroup-2-1] DEBUG io.netty.util.Recycler - -Dio.netty.recycler.blocking: false

19:03:29.587 [nioEventLoopGroup-2-1] DEBUG io.netty.util.Recycler - -Dio.netty.recycler.batchFastThreadLocalOnly: true

19:03:29.596 [nioEventLoopGroup-2-1] DEBUG io.netty.buffer.AbstractByteBuf - -Dio.netty.buffer.checkAccessible: true

19:03:29.596 [nioEventLoopGroup-2-1] DEBUG io.netty.buffer.AbstractByteBuf - -Dio.netty.buffer.checkBounds: true

19:03:29.597 [nioEventLoopGroup-2-1] DEBUG i.n.util.ResourceLeakDetectorFactory - Loaded default ResourceLeakDetector: io.netty.util.ResourceLeakDetector@55c601cb

19:03:29.603 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xbcbe2182, L:/192.168.1.102:57680 - R:/192.168.1.206:8080] WRITE: 4

19:03:29.603 [nioEventLoopGroup-2-1] DEBUG i.n.handler.logging.LoggingHandler - [id: 0xbcbe2182, L:/192.168.1.102:57680 - R:/192.168.1.206:8080] FLUSH

```

