



链滴

# SpringBoot 集成 Netty 写的 TCP 服务器遇到的问题

作者: [Ever](#)

原文链接: <https://ld246.com/article/1594958527568>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

问题是这样子的，我使用SpringBoot写了个服务，然后启动了https，端口监听的是8080，然后我用etty写了个TCP服务器，端口监听9000，然后集成到SpringBoot中，所以打包成jar包之后启动会启两个服务，一个8080的http服务，一个9000的netty tcp服务。

在本地通过以下代码可以进行客户端连接到服务器

```
public final class EchoClient {  
    static final String HOST = System.getProperty("host", "localhost");  
    static final int PORT = Integer.parseInt(System.getProperty("port", "9000"));  
    static final int SIZE = Integer.parseInt(System.getProperty("size", "256"));  
  
    public static void main(String[] args) throws Exception {  
        EventLoopGroup group = new NioEventLoopGroup();  
        try {  
            Bootstrap b = new Bootstrap();  
            b.group(group)  
                .channel(NioSocketChannel.class)  
                .remoteAddress(new InetSocketAddress(HOST, PORT))  
                .option(ChannelOption.TCP_NODELAY, true)  
                .handler(new ChannelInitializer<SocketChannel>() {  
                    @Override  
                    public void initChannel(SocketChannel ch) throws Exception {  
                        ChannelPipeline p = ch.pipeline();  
                        //p.addLast(new LoggingHandler(LogLevel.INFO));  
                        p.addLast(new EchoClientHandler());  
                    }  
                });  
            ChannelFuture channelFuture = b.connect().sync();  
            channelFuture.channel().closeFuture().sync();  
        } finally {  
            // Shut down the event loop to terminate all threads.  
            group.shutdownGracefully();  
        }  
    }  
  
    public class EchoClientHandler extends ChannelInboundHandlerAdapter {  
        private static final Logger log = Logger.getLogger(EchoClientHandler.class.getName());  
  
        private final ByteBuf firstMessage;  
  
        /**  
         * Creates a client-side handler.  
         */  
        public EchoClientHandler() {  
            firstMessage = Unpooled.buffer(EchoClient.SIZE);  
            firstMessage.writeBytes("YXV0aEluZm9AYWE6MTIxMTIxMjEy".getBytes(StandardCharsets.  
TF_8));  
        }  
  
        @Override  
        public void channelActive(ChannelHandlerContext ctx) throws Exception {  
            log.log(Level.INFO, "正在连接服务端");  
            if (ctx.channel().isOpen()) {  
        }
```

```

        log.info("已开辟连接链路");
        log.info("正在发送消息");
        ChannelFuture channelFuture = ctx.writeAndFlush(firstMessage);
        if (channelFuture.isDone()) {
            log.info("消息发送完毕");
        }
    }
    super.channelActive(ctx);
}

@Override
public void channelReadComplete(ChannelHandlerContext ctx) {
    ctx.flush();
}

@Override
public void exceptionCaught(ChannelHandlerContext ctx, Throwable cause) {
    // Close the connection when an exception is raised.
    cause.printStackTrace();
    ctx.close();
}
}

```

但是我把localhost修改成阿里云服务器的ip地址就不行

```

Exception in thread "main" io.netty.channel.AbstractChannel$AnnotatedConnectException: Connection refused: no further information: /ip地址:9000
Caused by: java.net.ConnectException: Connection refused: no further information
at sun.nio.ch.SocketChannelImpl.checkConnect(Native Method)
at sun.nio.ch.SocketChannelImpl.finishConnect(SocketChannelImpl.java:714)
at io.netty.channel.socket.nio.NioSocketChannel.doFinishConnect(NioSocketChannel.java:330)
at io.netty.channel.nio.AbstractNioChannel$AbstractNioUnsafe.finishConnect(AbstractNioChannel.java:334)
at io.netty.channel.nio.NioEventLoop.processSelectedKey(NioEventLoop.java:702)
at io.netty.channel.nio.NioEventLoop.processSelectedKeysOptimized(NioEventLoop.java:650)
at io.netty.channel.nio.NioEventLoop.processSelectedKeys(NioEventLoop.java:576)
at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:493)
at io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:99)
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:748)

```

目前知道的信息是：

1. 使用Windows Server 2008 RC2
2. 已经关闭了防火墙
3. 已经在阿里云配置了安全组规则，入站出站规则都配置了
4. SpringBoot服务肯定是已经启动了的，在服务器上使用客户端连接没有问题
5. SpringBoot启用了https，tcp没有
6. 通过telnet ip 9000 不通

不知道是什么原因，还请各位帮忙。