

11-java.io.IOException: Connection reset by peer

jira, confluence系统会在一段时间内出现以下日志信息

```
at org.apache.tomcat.util.net.SocketProcessorBase.run(SocketProcessorBase.java:49)
at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)
at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)
at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
at java.base/java.lang.Thread.run(Unknown Source)
Caused by: org.apache.catalina.connector.ClientAbortException: java.io.IOException: Connection reset by peer
at org.apache.catalina.connector.OutputBuffer.realWriteBytes(OutputBuffer.java:351)
at org.apache.catalina.connector.OutputBuffer.flushByteBuffer(OutputBuffer.java:776)
at org.apache.catalina.connector.OutputBuffer.append(OutputBuffer.java:681)
at org.apache.catalina.connector.OutputBuffer.writeBytes(OutputBuffer.java:386)
at org.apache.catalina.connector.OutputBuffer.write(OutputBuffer.java:364)
at org.apache.catalina.connector.CoyoteOutputStream.write(CoyoteOutputStream.java:96)
at com.atlassian.confluence.web.filter.ResponseOutputStreamFilter$NoopAfterCloseOutputStream.write(ResponseOutputStreamFilter.java:100)
at org.apache.commons.io.IOUtils.copyLarge(IOUtils.java:2315)
at org.apache.commons.io.IOUtils.copy(IOUtils.java:2270)
at org.apache.commons.io.IOUtils.copyLarge(IOUtils.java:2291)
at org.apache.commons.io.IOUtils.copy(IOUtils.java:2246)
at com.atlassian.confluence.image.effects.ImageFilterServlet.copyToResponse(ImageFilterServlet.java:295)
at com.atlassian.confluence.image.effects.ImageFilterServlet.writeTransformToResponse(ImageFilterServlet.java:206)
at com.atlassian.confluence.image.effects.ImageFilterServlet.doGetInTransaction(ImageFilterServlet.java:171)
at com.atlassian.confluence.image.effects.ImageFilterServlet.lambda$doGet$0(ImageFilterServlet.java:148)
.. 487 more
Caused by: java.io.IOException: Connection reset by peer
at java.base/sun.nio.ch.FileDispatcherImpl.write0(Native Method)
at java.base/sun.nio.ch.SocketDispatcher.write(Unknown Source)
at java.base/sun.nio.ch.IOUtil.writeFromNativeBuffer(Unknown Source)
at java.base/sun.nio.ch.IOUtil.write(Unknown Source)
at java.base/sun.nio.ch.IOUtil.write(Unknown Source)
at java.base/sun.nio.ch.SocketChannelImpl.write(Unknown Source)
at org.apache.tomcat.util.net.NioChannel.write(NioChannel.java:138)
at org.apache.tomcat.util.net.NioBlockingSelector.write(NioBlockingSelector.java:101)
at org.apache.tomcat.util.net.NioSelectorPool.write(NioSelectorPool.java:152)
at org.apache.tomcat.util.net.NioEndpoint$NioSocketWrapper.doWrite(NioEndpoint.java:1253)
at org.apache.tomcat.util.net.SocketWrapperBase.doWrite(SocketWrapperBase.java:740)
at org.apache.tomcat.util.net.SocketWrapperBase.writeBlocking(SocketWrapperBase.java:560)
```

常见产生这几种问题的以下几种情况：

- 服务器在接受处理用户请求时，自身的cpu、io、内存、线程等资源都是有最大限制的。当并发请求超过服务器的承载量时，服务器会停掉一些请求。
- 客户端关闭了浏览器，但是服务端还是继续发送数据。
- 浏览器端按了Stop。（ps：有的文档说浏览器弹框阻止也类似stop情况，本人没有亲自测试。可以参考：这篇文章）。上面两种都是客户端造成的，但是因为服务端在进行读写，依然会记录日志的，如果频繁发生，也会造成困扰。可以区分日志进行捕获，少量记录。（一般来说则上面3中情况出现的频率很低）
- 防火墙/nginx影响了。
 - 一种是他们都设置了超时机制，超过了会关闭这个TCP会话。这个可以通过重新定义心跳机制，进行会话维护。
 - 一种是设置防高频策略。