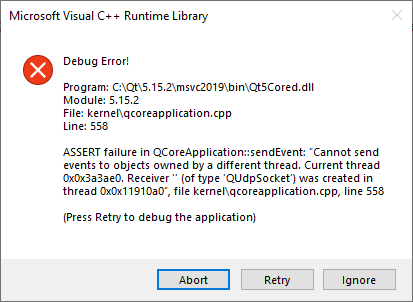
当服务器没有运行时, 如果客户端发送表白消息,就会弹出



main thread: QThread(0x11910a0)

listening thread: QThread(0x3a3ae0)

QUdpSocket对象是在主线程中创建的.

当前线程是监听线程, 监听线程现在在调用sendEvent.

为什么服务器运行时就没调用sendEvent?

难道客户端发现了服务器端没在运行?

我用的是udp协议啊?

我猜测: 当服务器存在时, 会收到表白消息后,会立即给客户端发安抚消息.

监听线程会收到安抚消息,然后对QUdpSocket对象进行某种操作.

导致后边发生sendEvent时, QUdpSocket对象已经处在一种正常状态,或者压根就不会引发sendEvent.

很可能这个sendEvent跟超时有关.

经验证,的确是超时导致.

将udpSocket.waitForReadyRead(5000);设为5秒,对话框就在五秒后弹出.

当服务器存在时,会及时发来安抚消息, waitForReadyRead根本不会超时. 超时后会sendEvent.

可能的解决办法,但我不想用:

让监听线程拥有QUdpSocket对象.

主线程发表白消息所用的函数不会超时.

我想寻求Qt的规范用法.

就是在两个线程间共用一个QUdpSocket对象的规范用法.

Java中我用私钥加密时抛出了异常

Exception in thread "main" java.security.InvalidKeyException: OAEP cannot be used to sign or verify signatures

这行代码导致的:

encryptCipher.init(Cipher.ENCRYPT\_MODE, privateKey, oaepParameterSpecJCE);

难道不能用私钥加密?

但我看这里

<https://mkyong.com/java/java-asymmetric-cryptography-example/>

就是用私钥加密的.