

12.2.0.1 and Above JDBC Connections Sometimes Fail With: IO Error: Socket Read Interrupted (Doc ID 2612009.1)

In this Document

[Symptoms](#)

[Changes](#)

[Cause](#)

[Solution](#)

[References](#)

APPLIES TO:

JDBC - Version 12.2.0.0 and later

Information in this document applies to any platform.

SYMPTOMS

When using version 12.2.0.1 or above of the Oracle JDBC driver, the following error is sometimes seen in application error logs:

```
IO Error: Socket read interrupted java.io.IOException: IO Error: Socket read interrupted
at oracle.jdbc.driver.OracleClobReader.needChars(OracleClobReader.java:252)
at oracle.jdbc.driver.OracleClobReader.read(OracleClobReader.java:198)
at java.io.Reader.read(Reader.java:151)
...
```

```
Caused by: java.sql.SQLRecoverableException: IO Error: Socket read interrupted
at oracle.jdbc.driver.T4CConnection.getChars(T4CConnection.java:3961)
at oracle.sql.CLOB.getChars(CLOB.java:448)
at oracle.jdbc.driver.OracleClobReader.needChars(OracleClobReader.java:235)
... 25 more Caused by: java.net.SocketTimeoutException: Socket read interrupted
at oracle.net.nt.TimeoutSocketChannel.read(TimeoutSocketChannel.java:152)
at oracle.net.ns.NIOHeader.readHeaderBuffer(NIOHeader.java:82)
at oracle.net.ns.NIOPacket.readFromSocketChannel(NIOPacket.java:139)
at oracle.net.ns.NIOPacket.readFromSocketChannel(NIOPacket.java:101)
at oracle.net.ns.NIONSDataChannel.readDataFromSocketChannel(NIONSDataChannel.java:80)
at oracle.jdbc.driver.T4CMAREngineNIO.prepareForReading(T4CMAREngineNIO.java:98)
at oracle.jdbc.driver.T4CMAREngineNIO.unmarshalUB1(T4CMAREngineNIO.java:534)
at oracle.jdbc.driver.T4CTTIFun.receive(T4CTTIFun.java:485)
```

The following exception stack may also be reported instead:

```
Caused by: java.io.InterruptedIOException: Operation interrupted
at oracle.net.nt.TimeoutSocketChannel.handleInterrupt(TimeoutSocketChannel.java:311)
at oracle.net.nt.TimeoutSocketChannel.write(TimeoutSocketChannel.java:221)
at oracle.net.ns.NIOPacket.writeToSocketChannel(NIOPacket.java:211)
at oracle.net.ns.NIONSDataChannel.writeDataToSocketChannel(NIONSDataChannel.java:181)
at oracle.net.ns.NIONSDataChannel.writeDataToSocketChannel(NIONSDataChannel.java:132)
at oracle.net.ano.CryptoNIONSDataChannel.writeDataToSocketChannel(Unknown Source)
at oracle.jdbc.driver.T4CMAREngineNIO.prepareForReading(T4CMAREngineNIO.java:96)
at oracle.jdbc.driver.T4CMAREngineNIO.unmarshalUB1(T4CMAREngineNIO.java:534)
at oracle.jdbc.driver.T4CTTIFun.receive(T4CTTIFun.java:485)
at oracle.jdbc.driver.T4CTTIFun.doRPC(T4CTTIFun.java:252)
```

```
at oracle.jdbc.driver.T4C8Oall.doOALL(T4C8Oall.java:612)
at oracle.jdbc.driver.T4CPreparedStatement.doOall8(T4CPreparedStatement.java:226)
at oracle.jdbc.driver.T4CPreparedStatement.fetch(T4CPreparedStatement.java:1023)
```

CHANGES

CAUSE

This is the issue investigated in unpublished / internal Bug 26414928 - SOCKET READ INTERRUPTED AND IO ERROR: OPERATION INTERRUPTED WITH OJDBC8 JARS

This was closed as "Not a Bug".

The 12.2.0.1 and above JDBC driver uses Java NIO calls in blocking mode, which can be impacted by any interrupt() calls being made by the application.

This differs from previous versions of the JDBC driver, which used stream-based I/O API calls that were not affected by calls to interrupt().

Note that this is a deliberate / intentional change beginning in the 12.2.0.1 JDBC driver, rather than a bug.

SOLUTION

Inspect the application code for any interrupt() method calls.

Alternatively, set the following connection property: oracle.jdbc.javaNetNio=false.

Example:

```
java -Doracle.jdbc.javaNetNio=false ...
```

This will direct the JDBC driver to revert to the pre-12.2 behavior of using stream-based I/O API calls no affected by interrupt() method calls.

REFERENCES

Didn't find what you are looking for?