**SUMMARY**

A concurrency bug in Eclipse 3.5 Platform Debug.

**DETAILS**

Some details can also be found at: [https://bugs.eclipse.org/bugs/show\_bug.cgi?id=261837](https://bugs.eclipse.org/bugs/show_bug.cgi?id=57467)

This bug is due to a data race.

|  |  |
| --- | --- |
| Thread1 () | Thread2 () |
| while (!process.isTerminated()) {} | process.terminate(); |
| public boolean isTerminated() {  return fTerminated;  } | public void terminate() throws DebugException {  if (!isTerminated()) {  if (fStreamsProxy instanceof StreamsProxy) {  ((StreamsProxy)fStreamsProxy).kill();  }  Process process = getSystemProcess();  if (process != null) {  process.destroy();  }  int attempts = 0;  while (attempts < MAX\_WAIT\_FOR\_DEATH\_ATTEMPTS) {  try {  process = getSystemProcess();  if (process != null) {  fExitValue = process.exitValue(); // throws exception if process not exited  }  return;  } catch (IllegalThreadStateException ie) {  }  try {  Thread.sleep(TIME\_TO\_WAIT\_FOR\_THREAD\_DEATH);  } catch (InterruptedException e) {  }  attempts++;  }  // clean-up  if (fMonitor != null) {  fMonitor.killThread();  fMonitor = null;  }  IStatus status = new Status(IStatus.ERROR, DebugPlugin.getUniqueIdentifier(), DebugException.TARGET\_REQUEST\_FAILED, DebugCoreMessages.RuntimeProcess\_terminate\_failed, null);  throw new DebugException(status);  }  } |
| 0-lock | 0-lock |