**SUMMARY**

A concurrency bug in Eclipse 3.1 Platform Debug.

**DETAILS**

Some details can also be found at: [https://bugs.eclipse.org/bugs/show\_bug.cgi?id=82931](https://bugs.eclipse.org/bugs/show_bug.cgi?id=57467)

This bug is due to a data race.

MemoryBlockTreeViewPane - the tree viewer is disposed in the middle of the memory blocks removed event. Check if the viewer is disposed before proccessing an event.

|  |  |
| --- | --- |
| Thread1 (MemoryBlocksTreeViewPane.java) | Thread2 (MemoryBlocksTreeViewPane.java) |
| public void dispose()  {  fMemoryBlocks.clear();  fParent.getViewSite().getPage().removeSelectionListener(this);  fContentProvider.dispose();  fAddMemoryBlockAction.dispose();  } | public void memoryBlocksRemoved(final IMemoryBlock[] memory) {  DebugUIPlugin.getDefault().getWorkbench().getDisplay().syncExec(new Runnable() {  public void run() {  for (int i=0; i<memory.length; i++)  {  fMemoryBlocks.remove(memory[i]);  }  fTreeViewer.refresh();  IMemoryBlock[] memoryBlocks = DebugPlugin.getDefault().getMemoryBlockManager().getMemoryBlocks(fDebugTarget);  if (memoryBlocks != null && memoryBlocks.length > 0) {  fTreeViewer.setSelection(new StructuredSelection(memoryBlocks[0]));  .. |
| 0-lock | 0-lock |