
Subject: Re: I keep getting the same "cannot open exe for writing" error...

Posted by [peterh](#) on Sat, 08 Dec 2018 18:27:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

New Code:

```
void Pdb::Stop()
{
    DWORD retval;
    LOG("-----");
    LOG("terminated flag");
    DUMP(terminated);
    if(!terminated) {
        terminated = true;
        SaveTree();
        String fn = ConfigFile("TreeTypes.txt");
        FileOut out(fn);
        for(int i = 0; i < treetype.GetCount(); i++)
            out << treetype.GetKey(i) << "\r\n" << treetype[i] << "\r\n";
        StringStream ss;
        Store(callback(this, &Pdb::SerializeSession), ss);
        WorkspaceConfigData("pdb-debugger") = ss;
        if(hProcess == INVALID_HANDLE_VALUE)
            { LOG("hProcess Handle is invalid handle value");};

        if(hProcess != INVALID_HANDLE_VALUE) {
            LOG("DebugActiveProcessStop");
            for(int i=0; i<10; i++){
                if (retval = DebugActiveProcessStop(processid)) break;
                DUMPHEX(retval);
                Sleep(100);
            }
            DUMPHEX(retval);

            retval = TerminateProcess(hProcess, 0);
            LOG("TerminateProcess");
            DUMPHEX(retval);
            retval = WaitForSingleObject(hProcess,10000);
            LOG("WaitForSingleObject");
            DUMPHEX(retval);
#ifdef NEVEREVER
            retval = MsgWaitForMultipleObjects(
                1, //DWORD      nCount,
                &hProcess, //const HANDLE *pHandles,
                true, //BOOL      fWaitAll,
                10000, //DWORD      dwMilliseconds,
                0 //DWORD      dwWakeMask
            );
#endif
        }
    }
}
```

```
    LOG("WaitForMultipleObjects");
    DUMPHEX(retval);
#endif
    while(threads.GetCount())
        RemoveThread(threads.GetKey(0)); // To CloseHandle
    UnloadModuleSymbols();
    SymCleanup(hProcess);
    retval = CloseHandle(hProcess);
    LOG("CloseHandle");
    DUMPHEX(retval);
}
StoreToGlobal(*this, CONFIGNAME);
IdeRemoveBottom(*this);
IdeRemoveRight(disas);
}
}
```
