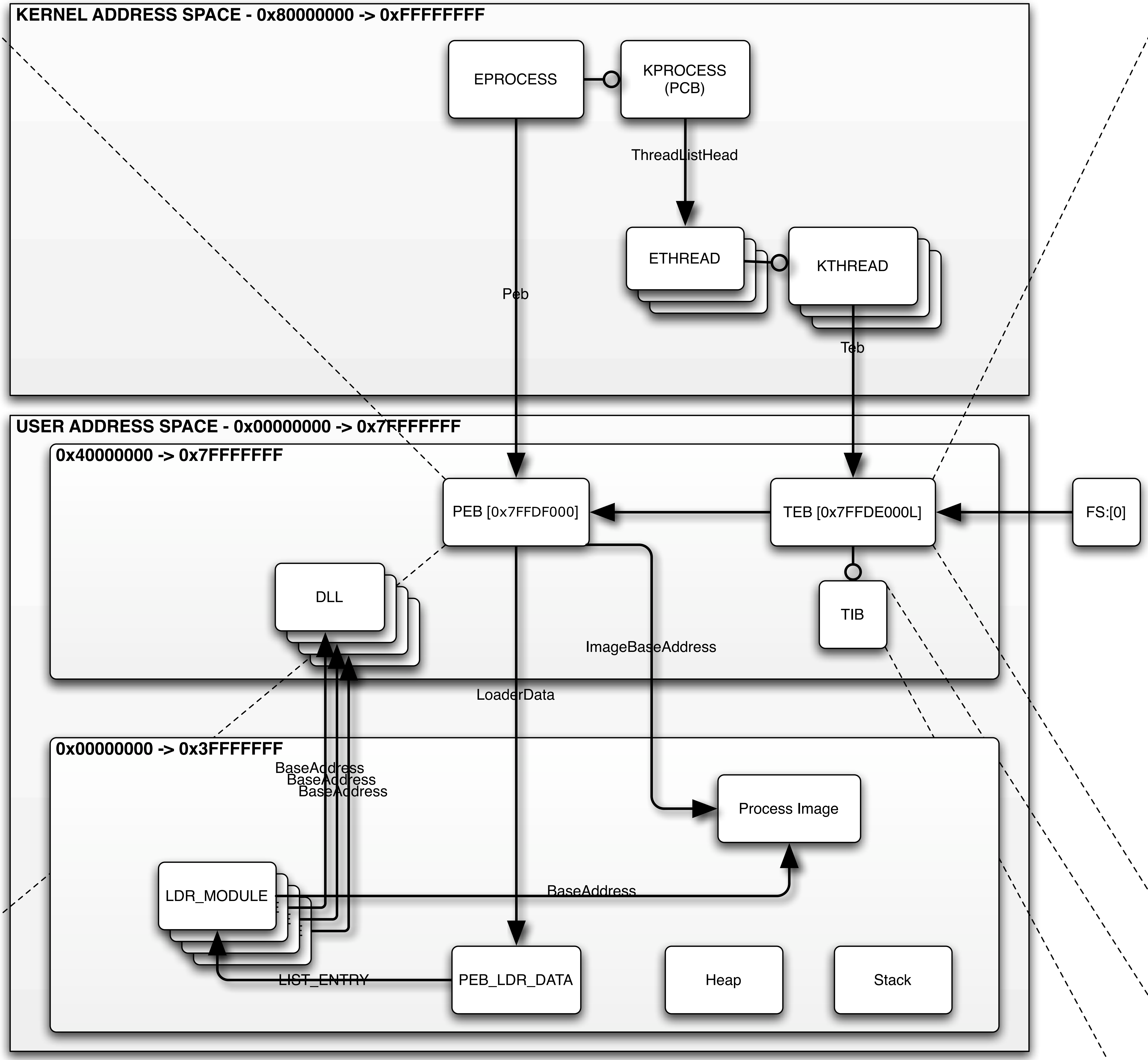
Memory Layout for Windows XP

```
struct _PEB {
0x000 BYTE InheritedAddressSpace;
0x001 BYTE ReadImageFileExecOptions;
0x002 BYTE BeingDebugged;
0x003 BYTE SpareBool;
0x004 void* Mutant;
0x008 void* ImageBaseAddress;
0x00c _PEB_LDR_DATA* Ldr;
0x010 _RTL_USER_PROCESS_PARAMETERS* ProcessParameters;
0x014 void* SubSystemData;
0x018 void* ProcessHeap;
0x01c _RTL_CRITICAL_SECTION* FastPebLock;
0x020 void* FastPebLockRoutine;
0x024 void* FastPebUnlockRoutine;
0x028 DWORD EnvironmentUpdateCount;
0x02c void* KernelCallbackTable;
0x030 DWORD SystemReserved[1];
0x034 DWORD ExecuteOptions:2;           // bit offset: 34, len=2
0x034 DWORD SpareBits:30;        // bit offset: 34, len=30
0x038 _PEB_FREE_BLOCK* FreeList;
0x03c DWORD TlsExpansionCounter;
0x040 void* TlsBitmap;
0x044 DWORD TlsBitmapBits[2];
0x04c void* ReadOnlySharedMemoryBase;
0x050 void* ReadOnlySharedMemoryHeap;
0x054 void** ReadOnlyStaticServerData;
0x058 void* AnsiCodePageData;
0x05c void* OemCodePageData;
0x060 void* UnicodeCaseTableData;
0x064 DWORD NumberOfProcessors;
0x068 DWORD NtGlobalFlag;
0x070 _LARGE_INTEGER CriticalSectionTimeout;
0x078 DWORD HeapSegmentReserve;
0x07c DWORD HeapSegmentCommit;
0x080 DWORD HeapDeCommitTotalFreeThreshold;
0x084 DWORD HeapDeCommitFreeBlockThreshold;
0x088 DWORD NumberOfHeaps;
0x08c DWORD MaximumNumberOfHeaps;
0x090 void** ProcessHeaps;
0x094 void* GdiSharedHandleTable;
0x098 void* ProcessStarterHelper;
0x09c DWORD GdiDCAttributeList;
0x0a0 void* LoaderLock;
0x0a4 DWORD OSMajorVersion;
0x0a8 DWORD OSMinorVersion;
0x0ac WORD OSBuildNumber;
0x0ae WORD OSCSDVersion;
0x0b0 DWORD OSPlatformId;
0x0b4 DWORD ImageSubsystem;
0x0b8 DWORD ImageSubsystemMajorVersion;
0x0bc DWORD ImageSubsystemMinorVersion;
0x0c0 DWORD ImageProcessAffinityMask;
0x0c4 DWORD GdiHandleBuffer[34];
0x14c void (*PostProcessInitRoutine)();
0x150 void* TlsExpansionBitmap;
0x154 DWORD TlsExpansionBitmapBits[32];
0x1d4 DWORD SessionId;
0x1d8 _ULARGE_INTEGER AppCompatFlags;
0x1e0 _ULARGE_INTEGER AppCompatFlagsUser;
0x1e8 void* pShimData;
0x1ec void* AppCompatInfo;
0x1f0  _UNICODE_STRING CSDVersion;
0x1f8  void* ActivationContextData;
0x1fc  void* ProcessAssemblyStorageMap;
0x200 void* SystemDefaultActivationContextData;
0x204 void* SystemAssemblyStorageMap;
0x208 DWORD MinimumStackCommit;
};
```



```
struct _TEB {
0x000 _NT_TIB NtTib;
0x01c void* EnvironmentPointer;
0x020 _CLIENT_ID ClientId;
0x028 void* ActiveRpcHandle;
0x02c void* ThreadLocalStoragePointer;
0x030 _PEB* ProcessEnvironmentBlock;
0x034 DWORD LastErrorValue;
0x038 DWORD CountOfOwnedCriticalSections;
0x03c void* CsrClientThread;
0x040 void* Win32ThreadInfo;
0x044 DWORD User32Reserved[26];
0x0ac DWORD UserReserved[5];
0x0c0 void* WOW32Reserved;
0x0c4 DWORD CurrentLocale;
0x0c8 DWORD FpSoftwareStatusRegister;
0x0cc void* SystemReserved1[54];
0x1a4 int ExceptionCode;
0x1a8 _ACTIVATION_CONTEXT_STACK ActivationContextStack;
0x1bc DWORD SpareBytes1[24];
0x1d4 _GDI_TEB_BATCH GdiTebBatch;
0x6b4 _CLIENT_ID RealClientId;
0x6bc void* GdiCachedProcessHandle;
0x6c0 DWORD GdiClientPID;
0x6c4 DWORD GdiClientTID;
0x6c8 void* GdiThreadLocalInfo;
0x6cc DWORD Win32ClientInfo[62];
0x7c4 void* glDispatchTable[233];
0xb68 DWORD glReserved1[29];
0xbdc void* glReserved2;
0xbe0 void* glSectionInfo;
0xbe4 void* glSection;
0xbe8 void* glTable;
0xbec void* glCurrentRC;
0xbf0  void* glContext;
0xbf4  DWORD LastStatusValue;
0xbf8  _UNICODE_STRING StaticUnicodeString;
0xc00 WORD StaticUnicodeBuffer[261];
0xe0c void* DeallocationStack;
0xe10 void* TlsSlots[64];
0xf10  _LIST_ENTRY TlsLinks;
0xf18  void* Vdm;
0xf1c  void* ReservedForNtRpc;
0xf20  void* DbgSsReserved[2];
0xf28  DWORD HardErrorsAreDisabled;
0xf2c  void* Instrumentation[16];
0xf6c  void* WinSockData;
0xf70  DWORD GdiBatchCount;
0xf74  UChar InDbgPrint;
0xf75  UChar FreeStackOnTermination;
0xf76  UChar HasFiberData;
0xf77  UChar IdealProcessor;
0xf78  DWORD Spare3;
0xf7c  void* ReservedForPerf;
0xf80  void* ReservedForOle;
0xf84  DWORD WaitingOnLoaderLock;
0xf88  _Wx86ThreadState Wx86Thread;
0xf94  void** TlsExpansionSlots;
0xf98  DWORD ImpersonationLocale;
0xf9c  DWORD IsImpersonating;
0xfa0  void* NlsCache;
0xfa4  void* pShimData;
0xfa8  DWORD HeapVirtualAffinity;
0xfac  void* CurrentTransactionHandle;
0xfb0  _TEB_ACTIVE_FRAME* ActiveFrame;
};
```

```
struct _NT_TIB {
0x00  _EXCEPTION_REGISTRATION_RECORD* ExceptionList;
0x04  void* StackBase;
0x08  void* StackLimit;
0x0c  void* SubSystemTib;
0x10  void* FiberData;
0x10  DWORD Version;
0x14  void* ArbitraryUserPointer;
0x18  _NT_TIB* Self;
};
```

———○ Structure contained within parent

——▶ Structure pointed to by the parent

Last updated on Fri Dec 16 2005
Created by Ero Carrera Ventura

References:

NTIllusion: A portable Win32 userland rootkit, Kdm; Phrack 62, Volume 0x0b, Issue 0x3e, Phile #0x0c of 0x10

Inside Microsoft® Windows® 2000, Third Edition [Chapter 6: Processes, Threads, and Jobs]
http://www.microsoft.com/mspress/books/sampchap/4354.asp