使用Pintool进行恶意软件脱壳

(2012-8月)

[ref: http://jbremer.org/malware-unpacking-level-pintool/]

**摘要**

本文主要介绍使用Pintool脱壳RunPE的恶意软件。

**Pin**

Pin介绍参见官网及相关论文；

RunPE

Tan Cheew Keong发明的一种在Windows中启动另一个应用进程的方法。(LoadEXE: Dynamic Forking of Win32 EXE – Tan Chew Keong)

使用拦截系统调用：

*PIN\_AddSyscallEntryFunction*

*PIN\_AddSyscallExitFunction*

The following code snippet represents a simple pintool which dumps the system call number, the first four parameters to the system call, as well as the return value of every system call that’s being performed.

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35 | #include <stdio.h>  #include "pin.H"    void syscall\_entry(THREADID thread\_id, CONTEXT \*ctx,      SYSCALL\_STANDARD std, void \*v)  {      printf("system-call: %d, arguments:",          PIN\_GetSyscallNumber(ctx, std));      for (int i = 0; i < 4; i++) {          ADDRINT value = PIN\_GetSyscallArgument(ctx, std, i);          printf("  %d 0x%08x", value, value);      }  }    void syscall\_exit(THREADID thread\_id, CONTEXT \*ctx,      SYSCALL\_STANDARD std, void \*v)  {      ADDRINT return\_value = PIN\_GetSyscallReturn(ctx, std);      printf(", return-value: %d 0x%08x\n", return\_value,          return\_value);  }    int main(int argc, char \*argv[])  {      if(PIN\_Init(argc, argv)) {          printf("Usage: %s <binary> [arguments]\n");          return 0;      }        PIN\_AddSyscallEntryFunction(&syscall\_entry, NULL);      PIN\_AddSyscallExitFunction(&syscall\_exit, NULL);        PIN\_StartProgram();      return 0;  } |