ida-x86emu
x86 Emulator Plugin for IDA Pro

Chris Eagle

cseagle@nps.edu

CON 2004

Outline

- Introduction
- Operation
- Demos
- Summary

Background

- IDA Pro
  - Interactive Disassembler Professional
  - http://www.datarescue.com/idabase
- Premier disassembly tool for reverse engineers
  - Handles many families of assembly language
- Runs on Windows
  - Linux in the works!

What?

- ida-x86emu is a plugin for IDA Pro that allows emulated execution of x86 instructions
- Written in C++
  - Currently packaged as VC++ 6.0 project
- Available here:
  - http://sourceforge.net/projects/ida-x86emu

CON 2004

Why?

- Hand tracing assembly language is a pain in the ass
- Anti-reverse engineering techniques attempt to obfuscate code paths
- Allows automated unpacking/decrypting of "protected" binaries
  - UPX, burneye, shiva, tElock, ASPack, …

Primary Motivation

- Getting at protected executables
  - Most viruses/worms are protected in some way
  - Often tweaked UPX
- Challenge for static reverse engineering is getting past the protection
  - ida-x86emu allows you to "run" through the decryption routine within IDA Pro

Outline

- Introduction
- Operation
- Demos
- Summary

IDA Pro

- Load the binary of interest
- IDA builds a database to characterize each byte of the binary
- Performs detailed analysis of code
  - Recognizes functions boundaries and library calls
  - Recognizes data types for library calls

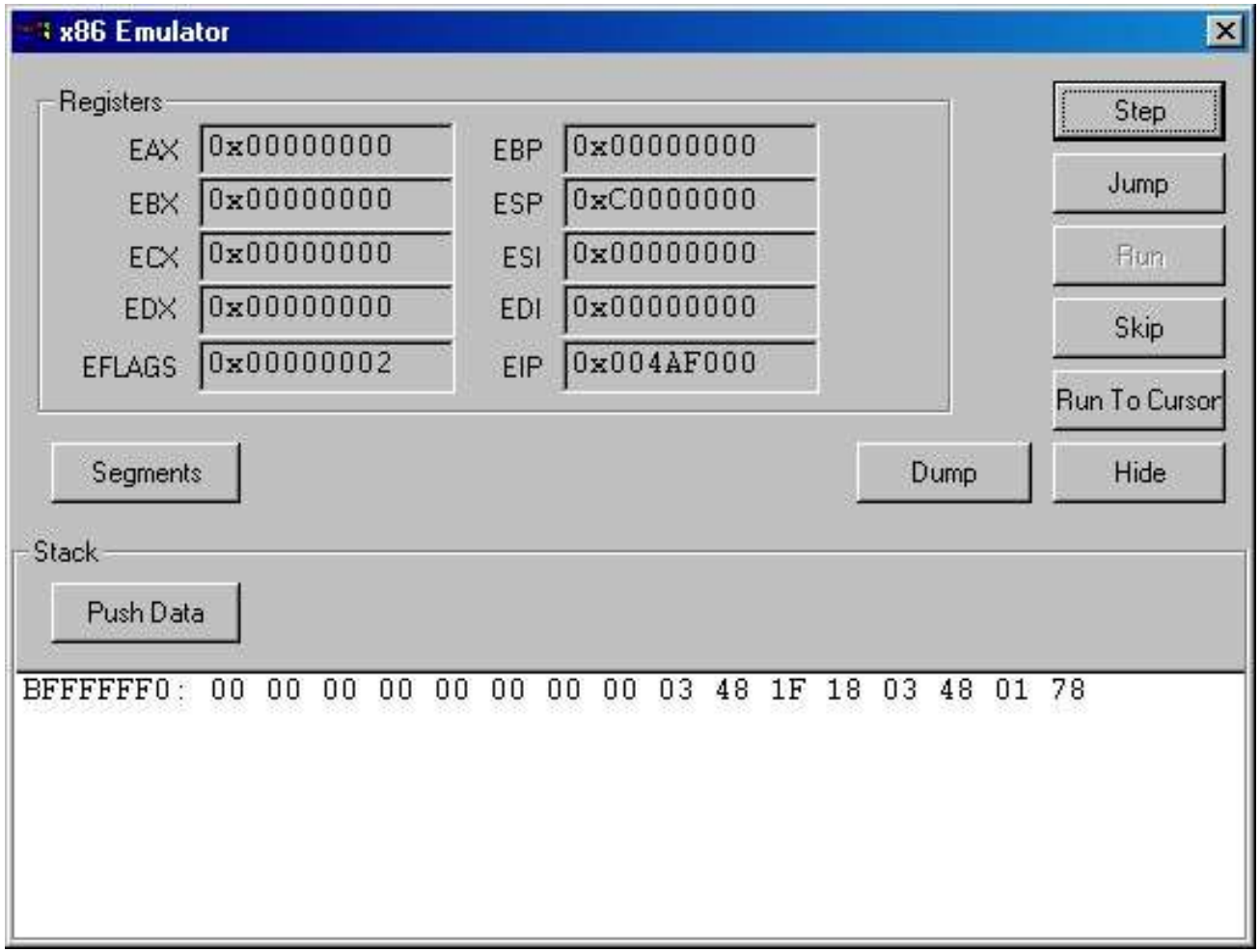
Obfuscated Code

- Challenging for IDA
- Usually only get sensible output for entry function
- Protected program appears as data rather than code because it is obfuscated/encrypted
- Jumps into middle of instructions confuse flow analysis

The Plugin

- Two pieces
  - User interface
    - Windows-specific gui code
    - Handles dialog boxes
  - x86 emulator
    - Platform independent
    - Executes a single instruction at a time
      - Reads from IDA database or user-supplied memory block

Console



CON 2004

Outline

- Introduction
- Operation
- Demos
- Summary

Using It

- Alt-F8 brings it up
- eip initialized to cursor
- Step and go
  - The plugin tells IDA to reorganize its code display based on ACTUAL code paths
  - Defeats jump into the middle of an instruction type obfuscation

Features

- Run to Cursor
  - No breakpoints yet
- Plugin supplies its own stack
  - Stack push places arguments on the stack
  - Useful if you want to setup a function call
- No dynamic memory at this point
  - Can fake small heap operations using the stack

UPX Demo

- One of the most common obfuscators
- Reversible using UPX itself
- UPX corruptors exist that break UPX's reversing capability
- No problem for the plugin

Burneye Demo

- Early ELF protector by Team TESO
- Actually embeds the entire protected ELF, including the ELF headers within
- Once decrypted, the protected binary can be dumped out of the IDA database

Shiva Demo

- Shiva is a binary protector
  - Similar goals to Burneye
- Multilevel encryption protects binary
- Polymorphic stage 1 decryptor
- Embedded key recovery functions for last stage decryption


LOAD:0A04B0D0 ; ---------- CODE XREF: start+B↑j
LOAD:0A04B0D0 loc_A04B0D0:
LOAD:0A04B0D0 sub esp, 4
LOAD:0A04B0D6 mov [esp], esi
LOAD:0A04B0D9 push ecx
LOAD:0A04B0DA push edi
LOAD:0A04B0DB push eax
LOAD:0A04B0DC jz short near ptr loc_A04B0E1+2
LOAD:0A04B0DE push eax
LOAD:0A04B0DF jnz short near ptr loc_A04B0E1+1
LOAD:0A04B0E1 loc_A04B0E1:
LOAD:0A04B0E1 ; CODE XREF: LOAD:0A04B0DF↑j
LOAD:0A04B0E1 ; LOAD:0A04B0DC↑j
LOAD:0A04B0E1 mov eax, 0BE535258h
LOAD:0A04B0E6 xlat
LOAD:0A04B0E7 sub dl, [edx]
LOAD:0A04B0E9 aad 81h
LOAD:0A04B0EB out dx, al
LOAD:0A04B0EC jz short loc_A04B14D
LOAD:0A04B0EE jmp near ptr 70CDE25Dh
LOAD:0A04B0EE ; ----------
LOAD:0A04B0F3 dd 814B63B9h, 2C0000C1h, 0EBCE3160h, 92B9BE01h, 3107CF97h
LOAD:0A04B0F3 dd 80BF66FFh, 0C78126h, 3110A4C0h, 52E981F9h, 89176B40h
LOAD:0A04B0F3 dd 0F7D0B9CFh, 0F181596Fh, 28D44DEAh, 0B866C031h, 0C0811DDDh
LOAD:0A04B0F3 dd 46E50000h, 0C889C129h, 90C701EBh, 5740775h, 8041404h
LOAD:0A04B143 db 68h
LOAD:0A04B144 ; ----------
LOAD:0A04B144 loc_A04B144:
LOAD:0A04B144 ; CODE XREF: LOAD:0A04B174↓j
LOAD:0A04B144 cmp edi, 4
LOAD:0A04B14A jz short loc_A04B14F
LOAD:0A04B14A ; ----------
LOAD:0A04B14C db 75h ; u
LOAD:0A04B14D ; ----------
LOAD:0A04B14D loc_A04B14D:
LOAD:0A04B14D ; CODE XREF: LOAD:0A04B0EC↑j
LOAD:0A04B14D add bh, bh

Shiva Key Recovery

- Shiva contains 5 different types of encrypted blocks
- Each block gets its own key
  - Blocks of same type share the same key
- In this case we need to recover 5 keys in order to decrypt all of the types of blocks

Key Obfuscation

- Shiva contains a key reconstruction function for each type of crypt block
- Block decryption sequence
  - Identify block type (0-IV)
  - Call appropriate key reconstruction function
  - Decrypt block
  - Clear the key

Key Construction

- Functions are obfuscated
  - Similar to layer 1 decrypt
  - Differ from one binary to the next
  - Resistant to script-based recovery
- But
  - They are easy to locate
  - A table points to the start of each function

Key Extraction

- The plugin can be used to run the functions and collect the keys!
- Demo

Using the Keys

- With 5 keys in hand it is possible to decrypt all of the crypt blocks
- The plugin can be used to invoke Shiva's decryption function
  - Setup the stack
    - Pointer to the block
    - Pointer to the key
  - Step through the decryption function

Outline

- Introduction
- Operation
- Demos
- Summary

To Do

- Breakpoints
- Handle library calls
- Heap functionality
- Windows exception handling

Summary

- Acts as something of a "universal" decryption script for protected binaries
- Dramatically reduces time to reverse protected binaries
- Emulator code can be used independently of gui code to create automated unwrappers
  - Combine with ELF or PE parser
- Suggestions welcome

Questions?

- Thanks for coming
- Contact info:
  - Chris Eagle
  - cseagle@nps.edu

References

- Armouring the ELF: Binary encryption on the UNIX platform, grugq & scut, http://www.phrack.org/phrack/58/p58-0x05
- Shiva: Advances in ELF Runtime Binary Encryption, Clowes & Mehta, Black Hat USA 2003, http://www.blackhat.com/presentations/bh-usa-03/bh-us-03-
- Strike/Counter Strike: Reverse Engineering Shiva, Eagle, Black Hat Federal 2003, http://www.blackhat.com/presentations/bh-federal-03/bh-fed

CON 2004

References

- Shiva-0.96, Clowes & Mehta, http://www.blackhat.com/presentations/bh-usa-03/bh-us-03-
- Burneye-1.0.1, scut, http://teso.scene.at/releases/burneye-1.0.1-src.tar.bz2
- IDA Pro, Data Rescue, http://www.datarescue.com/idabase/
- The Ultimate Packer for eXecutables http://upx.sourceforge.net/