

Beyond EIP

spoonm & skape

BlackHat, 2005

Part I

Introduction

Who are we?

- ▶ spoonm
 - ▶ Full-time student at a Canadian university
 - ▶ Metasploit developer since late 2003
- ▶ skape
 - ▶ Lead software developer by day
 - ▶ Independent security researcher by night
 - ▶ Joined the Metasploit project in 2004

What will we discuss?

- ▶ Payload stagers
 - ▶ Windows Ordinal Stagers
 - ▶ PassiveX
 - ▶ Egghunt

What will we discuss?

- ▶ Payload stagers
 - ▶ Windows Ordinal Stagers
 - ▶ PassiveX
 - ▶ Egghunt
- ▶ Payload stages
 - ▶ Library Injection
 - ▶ The Meterpreter
 - ▶ DispatchNinja

What will we discuss?

- ▶ Payload stagers
 - ▶ Windows Ordinal Stagers
 - ▶ PassiveX
 - ▶ Egghunt
- ▶ Payload stages
 - ▶ Library Injection
 - ▶ The Meterpreter
 - ▶ DispatchNinja
- ▶ Post-exploitation suites
 - ▶ Very hot area of research for the Metasploit team
 - ▶ Suites built off of advanced payload research
 - ▶ Client-side APIs create uniform automation interfaces
 - ▶ Primary focus of Metasploit 3.0

Background: the exploitation cycle

- ▶ **Pre-exploitation** - Before the attack
 - ▶ Find a bug and isolate it
 - ▶ Write the exploit, payloads, and tools

Background: the exploitation cycle

- ▶ **Pre-exploitation** - Before the attack
 - ▶ Find a bug and isolate it
 - ▶ Write the exploit, payloads, and tools
- ▶ **Exploitation** - Leveraging the vulnerability
 - ▶ Find a vulnerable target
 - ▶ Gather information
 - ▶ Initialize tools and post-exploitation handlers
 - ▶ Launch the exploit

Background: the exploitation cycle

- ▶ **Pre-exploitation** - Before the attack
 - ▶ Find a bug and isolate it
 - ▶ Write the exploit, payloads, and tools
- ▶ **Exploitation** - Leveraging the vulnerability
 - ▶ Find a vulnerable target
 - ▶ Gather information
 - ▶ Initialize tools and post-exploitation handlers
 - ▶ Launch the exploit
- ▶ **Post-exploitation** - Manipulating the target
 - ▶ Command shell redirection
 - ▶ Arbitrary command execution
 - ▶ Pivoting
 - ▶ Advanced payload interaction

Part II

Exploitation Technology's State of Affairs

Payload encoders

- ▶ Robust and elegant encoders do exist
 - ▶ SkyLined's Alpha2 x86 alphanumeric encoder
 - ▶ Spoonm's high-permutation Shikata Ga Nai

Payload encoders

- ▶ Robust and elegant encoders do exist
 - ▶ SkyLined's Alpha2 x86 alphanumeric encoder
 - ▶ Spoonm's high-permutation Shikata Ga Nai
- ▶ Payload encoders generally taken for granted
 - ▶ Most encoders use a static decoder stub
 - ▶ Makes NIDS signatures easy to write

NOP generators

- ▶ NOP generation hasn't publicly changed much
 - ▶ Most PoC exploits use predictable single-byte NOPs (0x90), if any
 - ▶ ADMmutate's NOP generator easily signed by NIDS (Snort, Fnord)
 - ▶ Not considered an important research topic to most

NOP generators

- ▶ NOP generation hasn't publicly changed much
 - ▶ Most PoC exploits use predictable single-byte NOPs (0x90), if any
 - ▶ ADMmutate's NOP generator easily signed by NIDS (Snort, Fnord)
 - ▶ Not considered an important research topic to most
- ▶ Still, NIDS continues to play chase the tail
 - ▶ The mouse always has the advantage; NIDS is reactive
 - ▶ Advanced NOP generators and encoders push NIDS to its limits
 - ▶ Many protocols can be complex to signature (DCERPC fragmentation)

NOP generators

- ▶ NOP generation hasn't publicly changed much
 - ▶ Most PoC exploits use predictable single-byte NOPs (0x90), if any
 - ▶ ADMmutate's NOP generator easily signatored by NIDS (Snort, Fnord)
 - ▶ Not considered an important research topic to most
- ▶ Still, NIDS continues to play chase the tail
 - ▶ The mouse always has the advantage; NIDS is reactive
 - ▶ Advanced NOP generators and encoders push NIDS to its limits
 - ▶ Many protocols can be complex to signature (DCERPC fragmentation)
- ▶ Metasploit 2.4 released with a wide-distribution multi-byte x86 NOP generator (Opty2)

Exploitation techniques

- ▶ Exploitation techniques have become very mature
 - ▶ Linux/BSD/Solaris techniques are largely unchanged
 - ▶ Windows heap overflows can be made more reliable (Oded/Shok)
 - ▶ Windows SEH overwrites make exploitation easy, even on XPSP2

Exploitation techniques

- ▶ Exploitation techniques have become very mature
 - ▶ Linux/BSD/Solaris techniques are largely unchanged
 - ▶ Windows heap overflows can be made more reliable (Oded/Shok)
 - ▶ Windows SEH overwrites make exploitation easy, even on XPSP2
- ▶ Exploitation vectors have been beaten to death

Exploitation techniques

- ▶ Exploitation techniques have become very mature
 - ▶ Linux/BSD/Solaris techniques are largely unchanged
 - ▶ Windows heap overflows can be made more reliable (Oded/Shok)
 - ▶ Windows SEH overwrites make exploitation easy, even on XPSP2
- ▶ Exploitation vectors have been beaten to death
- ▶ ...so we wont be talking about them

Standard payloads

- ▶ Standard payloads provide the most basic manipulation of a target
 - ▶ Port-bind command shell
 - ▶ Reverse (connectback) command shell
 - ▶ Arbitrary command execution

Standard payloads

- ▶ Standard payloads provide the most basic manipulation of a target
 - ▶ Port-bind command shell
 - ▶ Reverse (connectback) command shell
 - ▶ Arbitrary command execution
- ▶ Nearly all PoC exploits use standard payloads

Standard payloads

- ▶ Standard payloads provide the most basic manipulation of a target
 - ▶ Port-bind command shell
 - ▶ Reverse (connectback) command shell
 - ▶ Arbitrary command execution
- ▶ Nearly all PoC exploits use standard payloads
- ▶ Command shells have poor automation support
 - ▶ Platform dependent intrinsic commands and scripting
 - ▶ Reliant on the set of applications installed on the machine
 - ▶ Hindered by chroot jails and host-based ACLs

“Advantage” payloads

- ▶ Advantage payloads provide enhanced manipulation of hosts, commonly through the native API
- ▶ Help to reduce the tediousness of writing payloads
- ▶ Core ST's InlineEgg

Part III

Payload Stagers

What are payload stagers?

- ▶ Typically small stubs that load and execute another payload
- ▶ Useful in conditions where size is limited

Introduction

Implementation: reverse stager

Overview

Implementation

Practical use: HTTP tunneling

Pros & cons

Overview

Hunting for eggs with SEH

Hunting for eggs with system calls

Part IV

Payload Stages

What are post-exploitation stages?

Overview

Types of library injection

In-memory library injection on Windows

In-memory library injection on UNIX

Library injection in action: VNC

Overview

Design goals

Communication protocol specification

Client/Server architecture

Extension flexibilities

Meterpreter extensions in action: Stdapi

Cool dN stuff here

Part V

Post-Exploitation Suites