Practical Reversing V – Exploit Development Basics

Harsimran Walia/Amit Malik

www.SecurityXploded.com

Disclaimer

The Content, Demonstration, Source Code and Programs presented here is "AS IS" without any warranty or conditions of any kind. Also the views/ideas/knowledge expressed here are solely of the trainer's only and nothing to do with the company or the organization in which the trainer is currently working.

However in no circumstances neither the trainer nor SecurityXploded is responsible for any damage or loss caused due to use or misuse of the information presented here.

Acknowledgement

- Special thanks to **null** & **Garage4Hackers** community for their extended support and cooperation.
- Thanks to all the trainers who have devoted their precious time and countless hours to make it happen.

Reversing & Malware Analysis Training

This presentation is part of our **Reverse Engineering & Malware Analysis** Training program. Currently it is delivered only during our local meet for FREE of cost.

For complete details of this course, visit our Security Training page.

Who am I #1

Harsimran Walia

- Research Scientist @ McAfee
- Expertise: Malware Analysis, Exploit development and Vulnerability Analysis
- Twitter: b44nz0r
- Email: walia.harsimran@gmail.com

Who am I #2

Amit Malik (sometimes DouBle_Zer0,DZZ)

- Member SecurityXploded
- Security Researcher @ McAfee Labs
- RE, Exploit Analysis/Development, Malware Analysis
- Email: m.amit30@gmail.com

Course Q&A

- Keep yourself up to date with latest security news
  - http://www.securityphresh.com

- For Q&A, join our mailing list.
  - http://groups.google.com/group/securityxploded

Contents

- What is an Exploit?
- Classification of exploits
- Exploitation Techniques
  - Direct EIP overwrite
  - SEH overwrite

Vulnerability

- In computer security, a vulnerability is a weakness which allows an attacker to reduce a system's information assurance.
- Vulnerability is the intersection of three elements: a system susceptibility or flaw, attacker access to the flaw, and attacker capability to exploit the flaw.

- Source: Wikipedia

Exploit

- Piece of software/code that takes advantage of a vulnerability in order to cause unintended or unanticipated behaviour to occur on computer software, hardware [Wiki]
- This frequently includes
  - gaining control of a computer system or
  - privilege escalation or
  - a denial-of-service attack.

Exploit (contd)

- Exploits can be in any form based on the software it exploits:
- Software : exploit
  - Adobe reader : pdf file
  - Microsoft word : doc file
  - Microsoft excel : xls file
  - Internet Explorer : Attacker hosted website or html file
  - and so on..

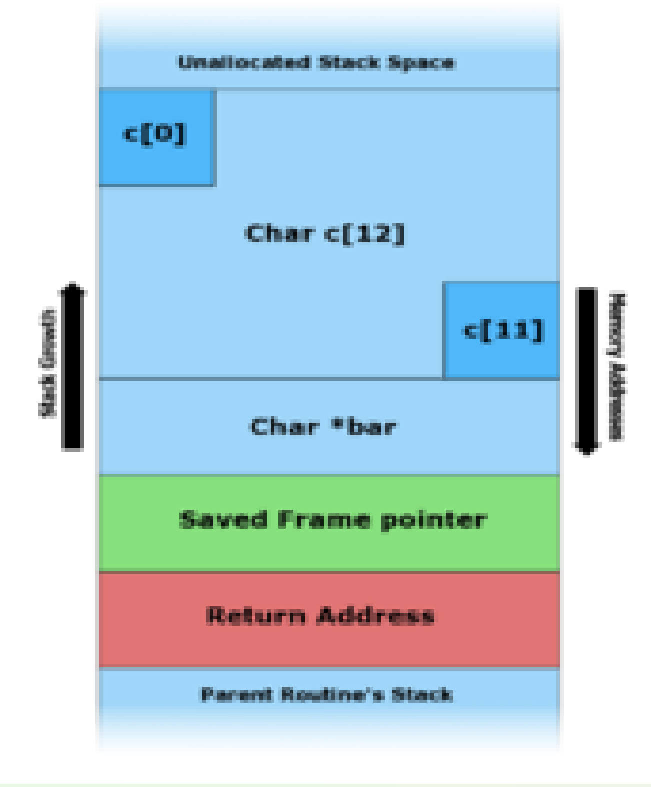
Classification

- Based on the vulnerability they exploit
  - Buffer Overflow, Memory Corruption, Use-After-Free
- Local or Remote
  - Local Privilege Escalation, Remote code execution
- Result of running the exploit
  - DoS, EoP etc

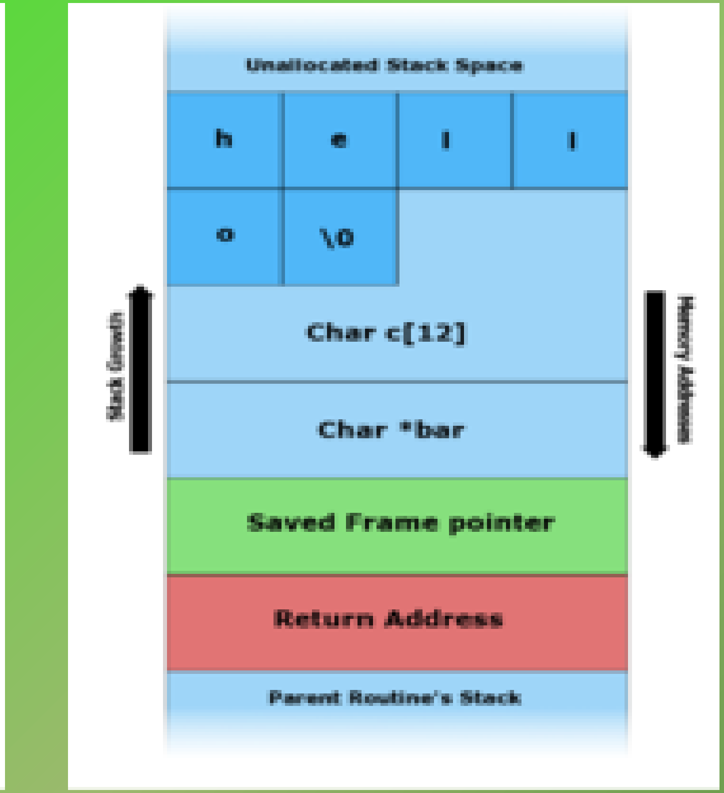
Stack Buffer Overflow

- Occurs when a program writes to memory addresses on the stack outside of the allocated buffer
- For exploiting a stack based buffer overflow is to overwrite the function return address with a pointer to attacker-controlled data (usually on the stack itself)

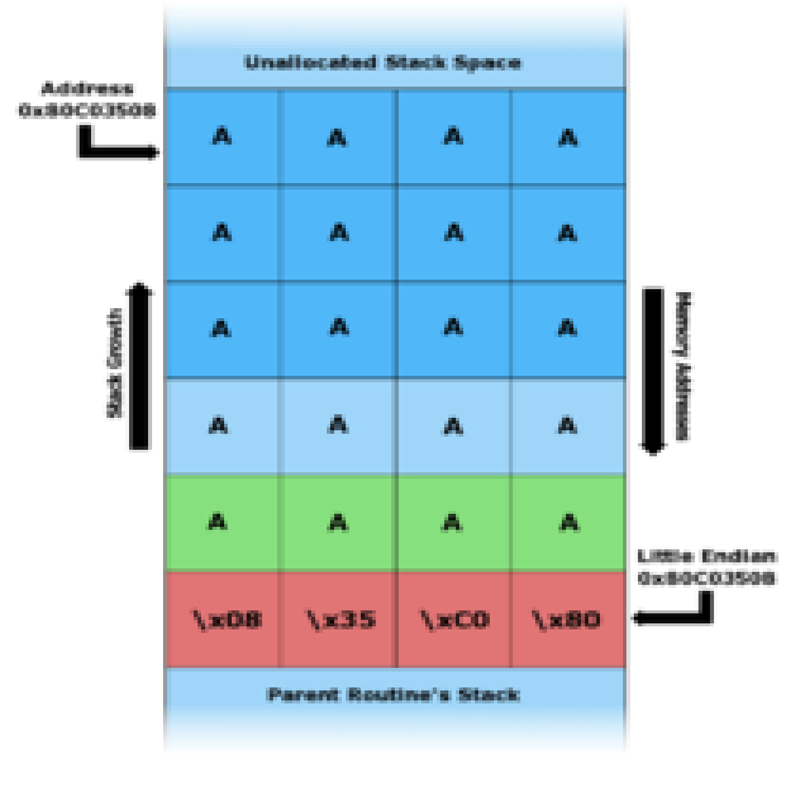
Stack Buffer Overflow in Action



A - Before data is copied.



B - "hello" is the first command line argument.



C - AAAAAAAAAAAAAAAAAAAA\x08\x35\xC0\x80" is the first command line argument.

Direct EIP overwrite (saved ret)

- Every Windows uses process memory that contains 3 major components :
  - code segment (executable instructions). The EIP keeps track of the next instruction
  - data segment (variables, dynamic buffers)
  - stack segment (used to pass data/arguments to functions, and is used as space for variables)
    - The stack starts (= the bottom of the stack) from the very end of the virtual memory of a page and grows upwards (to a lower address).
    - PUSH adds something to the top of the stack,
    - POP will remove one item (4 bytes) from the stack and puts it in a register.

EIP Overwrite Demo

- A vulnerability in
  - “Shadow Stream Recorder version 3.0.1.7
  - Buffer overflow when reading file (.asx)

Step -1 : Create a PoC to generate a crash in the software to verify the vuln

Step -2 : Find the offset to overwrite EIP

Step -3: Find an address of the “jmp esp” instruction

Step -4: Generate a shellcode and append it to the exploit code

Step -5: Putting it all together

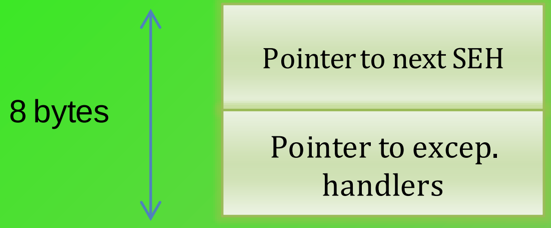
DEMO - EIP

- http://www.youtube.com/watch?v=erl_Aee8oDg

SEH Overwrite

- Exception?
  - An event which disrupts normal execution flow of code and requires execution outside normal flow
  - Software Exception –Generated by program (e.g Invalid file handle)
  - Hardware Exception – Access invalid memory, divide by zero etc
- SEH (structured exception handler)
  - Patented by Borland and licensed to Microsoft
  - Software's method of dispatching and handling exceptions
  - Can handle both software and hardware exceptions
  - For eg try{ } ; except { }; block
  - Whenever an exception happens control is passed on to the OS, which in turn locate and pass the control to the handler chain

SEH Overwrite in Action

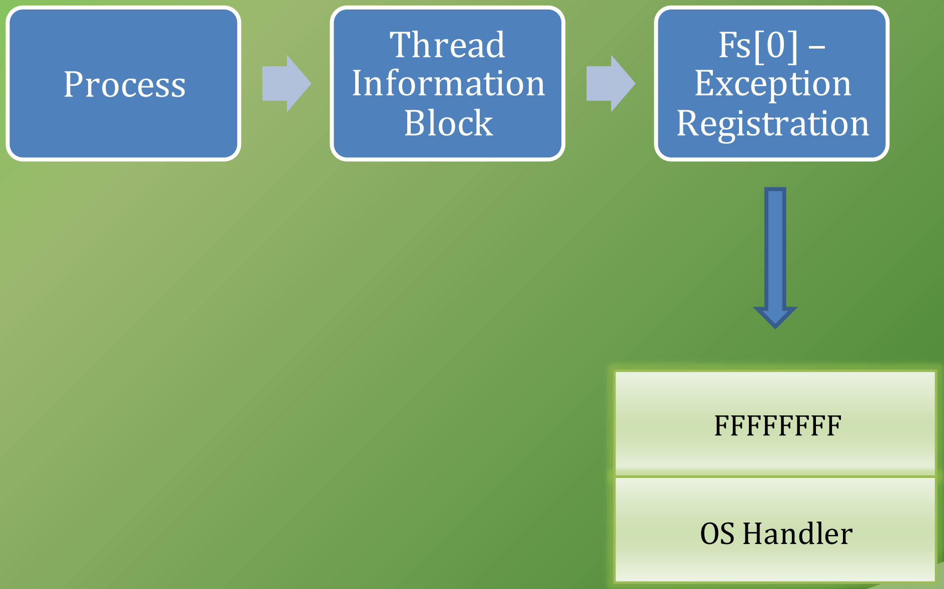


8 bytes

Pointer to next SEH

Pointer to excep. handlers

Exception Registration Record



Process

Thread Information Block

Fs[0] – Exception Registration

FFFFFFFF

OS Handler

Default ERR

SEH Overwrite Demo

- A vulnerability in
  - “MM Player 2.2
  - Buffer overflow when reading file (.ppl)

Step -1 : Create a PoC to generate a crash in the software to verify the vuln

Step -2 : Find the offset to overwrite nSEH + SEHandler

Step -3: Find an address of the command sequence “pop pop ret ”

Step -4: Generate a shellcode and append it to the exploit code

Step -5: Putting it all together

SEH Overwrite Demo

- http://www.youtube.com/watch?v=njQ47H7jO4s&feature=youtu.be

Reference

- Complete Reference Guide for Reversing & Malware Analysis Training

Thank You !

www.SecurityXploded.com