## SEH all-at-once attack

## a New technique to bypass SafeSEH+SEHOP protections

## Table of Contents

- 1. What is SEH ?
- 2. E\_R struct
- 3. E\_R (windbg)
- 4. classical SEH overwrite
- 5. SafeSEH, SEHOP
- 6. History of SEH Protections bypass
- 7. SEH All-at-Once attack method
- 8. Case by case exploit
- 9. End.

## 1. What is SEH?

MS Windows system's StackShield.
SEH(Structured Exception Handling)
fs:[0] saves nt!\_TEB and first member field is nt!\_TIB (not a pointer). nt!\_TIB's first field is latest installed SEH E\_R struct address.

# 2. E\_R struct(Exception\_Registration)

- Next | <- [ebp-10h]</p>
- I\*Exception Handler | <- [ebp-Ch]</p>
- Ian address of Image <- [ebp-8]</p>
- Index to User-defined exception handler | <- [ebp-4]</p>
- SFP |
- RET |

- ARGS |
- \*Next: 4-byte next E\_R struct pointer. Next E\_R's handler is older than current E\_R. (think about queue)
- \*Exception Handler: 4-byte Exception Handler function address.
- \*an address of Image: It used for calculating user-defined exception handler's address with next field(index) on \_except\_handler3 or \_except\_handler4 compiler generated exception handler.
- Index: it is the count of '\_\_try{ ... \_\_try{' coding from count 0FFFFFFFh.
  - ex) \_\_try{ } \_\_except(...){} then this count is zero(0)
    - \_\_try{ ... \_\_try{ then this count is 1.

# 3. E\_R (windbg)

- 0:000> dt \_EXCEPTION\_REGISTRATION\_RECORD
- ntdll!\_EXCEPTION\_REGISTRATION\_RECORD
- +0x000 Next : Ptr32 \_EXCEPTION\_REGISTRATION\_RECORD
- +0x004 Handler : Ptr32 \_EXCEPTION\_DISPOSITION
- Above two fields are E\_R struct's member.
- Note. I says E\_R struct on this presentation is

```
(Next | Handler | [ebp-8] | [ebp-4] ) == 16 bytes.
```

Note II. 

- Under applications built by 'Debug' mode, Handler gots kernel32!\_except\_handler3 address and [ebp-8], [ebp-4] will used to calculate user-defined exception handler.
- Under 'Release' mode, Handler gots the address of compiler generated Image!\_except\_handler3 and also works as 'Debug' mode.

## 4. classical SEH overwrite

| vuln buffer | E\_R struct |SFP|RET|SHELLCODE|
 ^

- Step 1. overflow the stack vuln buffer to E\_R struct.
   (E\_R.Next = '₩xeb₩x06₩x90₩x90')
  - (E\_R.Handler = address of 'pop; pop; ret;' sequence bytecode.)
  - Looking for the address of 'pop esi; pop esi; ret
  - ( '₩x5e₩x5e₩xc3') or 'add esp, 8;ret' to ROP(Return-Oriented-Programming).
  - And overwrite the address onto E\_R.Handler.

## When Exception Ocurred:

- E\_R.handler will called when after exception occurred.
- (Exception Dispatcher -> Exception Handler caller routines -> E\_R.handler called!)
- when before calling E\_R.handler, [esp+8] gots the address &E\_R struct.
- 'pop pop ret' pops 8 bytes and returned to &E\_R. then the overflowed short jmp will executed. '₩xeb₩x06 ( short jmp \$+6 )'. This jmp short jumping into shellcode

## 5. SafeSEH, SEHOP

#### SafeSEH protection:

- E\_R.handler must not pointers image area and windows DLL module address ranges(ntdll.dll, kernel32.dll, msvcrt.dll, ...)
- E\_R.handler doesn't pointers stack area.
- E\_R.handler only can pointers one of registered exception handler addresses or unloaded module address range.
- SEHOP protection:
- SEH chain(E\_R.Next -> E\_R.Next -> ...) must be never corrupted.
- E\_R struct address are must 4byte aligned.
- new final handler after default handler(kernel32!\_except\_handler\*) added on SEHOP applied platforms also must be never currupted.
- win server 2008, win server 2008 R2 (default enabled)
- win vista sp1 also supported but disabled by default.

## 6. History of SEH overflow

#### SafeSEH bypass method:

- 'Defeating the Stack Based Buffer Overflow Prevention Mechanism of
- Microsoft Windows 2003 Server.'
- (<u>http://www.ngssoftware.com/papers/defeating-w2k3-stack-protection.pdf</u>)
- David Litchfield, 8 September 2003.
- kind of approaches of SEH overflows

#### best SEH exploit writing tutorials:

Corelan Team (<u>http://www.corelan.be:8800/index.php/articles/</u>)

#### SEHOP bypass method:

- 'Bypassing SEHOP' (<u>http://www.sysdream.com/articles/sehop\_en.pdf</u>)
- SYSDREAM Cooperation, lastly (2010 ?!)

# 7. SEH All-at-Once attack method

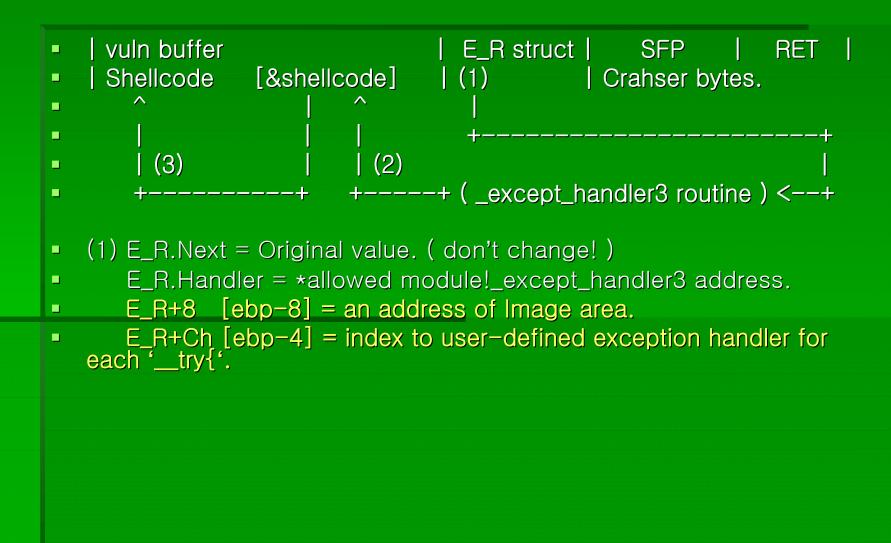
 new attack method to bypass SafeSEH + SEHOP protections all at once.

#### SafeSEH bypass:

 Registered Exception handler(DLL or Images compiler generated \_except\_handler3 or same code of \_except\_handler3) allowed to execute! Then can attack the handler!

#### SEHOP bypass:

 E\_R.Next chains must pointers next valid E\_R struct. (to final handler's E\_R struct) Then If we can change the E\_R.Next on overflowed stack frame. And saves shellcode onto vulnerable buffer than after '|RET|'.



- If you cacluate properly the [ebp-8] and [ebp-4] together to pointer an address of shellcode. Then After called \_except\_handler3 handler, firstly calculated the user-defined exception handler address by using the two values and finally makes an indirect calling using the address.
- When the calculated address(address of &shellcode) is called indirectly... then eip register pointers the shellcode!

- This method similar with one of David Litchfield's 2003 technique. But the method is some different at the way of execute shellcode. And it was applies only for SafeSEH. SEHOP protection was doesn't exists at the time.
- And so This presentation says a new try and new method to exploit SafeSEH+SEHOP.
- (about this, you can reference Case by case exploit's comment)

## 8. Case by case exploit

- AudioTran stackoverflow case by case exploit:
- http://www.x90c.org/All\_at\_Once\_SEH\_attack/audiotran\_safeseh\_seh op\_exploit(SEH\_all-at-once\_attack).c.txt

## 9. End

### Thank you.

- x90c (Kyong Joo, Jung)
- Personal Homepage: <u>http://www.x90c.org</u>
- E-mail: <u>geinblues@gmail.com</u>