

# [UNIX] Linux Kernel do\_mremap Local Privilege Escalation Vulnerability

**Source:** <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-01/0017.html>

---

**From:** SecuriTeam ([support\\_at\\_securiteam.com](mailto:support_at_securiteam.com))

**Date:** 01/05/04

To: [list@securiteam.com](mailto:list@securiteam.com)

Date: 5 Jan 2004 15:49:33 +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\_mremap Local Privilege Escalation Vulnerability

---

## SUMMARY

A critical security vulnerability has been found in the Linux kernel memory management code in `mremap(2)` system call due to incorrect bound checks.

## DETAILS

The `mremap` system call provides functionality of resizing (shrinking or growing) as well as moving across process's addressable space of existing virtual memory areas (VMAs) or any of its parts.

A typical VMA covers at least one memory page (which is exactly 4kB on the i386 architecture). An incorrect bound check discovered inside the `do_mremap()` kernel code performing remapping of a virtual memory area may lead to creation of a virtual memory area of 0 bytes length.

The problem bases on the general `mremap` flaw that remapping of 2 pages from inside a VMA creates a memory hole of only one page in length but an additional VMA of two pages. In the case of a zero sized remapping request no VMA hole is created but an additional VMA descriptor of 0 bytes in

## Securiteam: [UNIX] Linux Kernel do\_mremap Local Privilege Escalation Vulnerability

length is created.

Such a malicious virtual memory area may disrupt the operation of other parts of the kernel memory management subroutines finally leading to unexpected behavior.

A typical process's memory layout showing invalid VMA created with mremap system call:

```
08048000-0804c000 r-xp 00000000 03:05 959142 /tmp/test
0804c000-0804d000 rw-p 00003000 03:05 959142 /tmp/test
0804d000-0804e000 rwxp 00000000 00:00 0
40000000-40014000 r-xp 00000000 03:05 1544523 /lib/ld-2.3.2.so
40014000-40015000 rw-p 00013000 03:05 1544523 /lib/ld-2.3.2.so
40015000-40016000 rw-p 00000000 00:00 0
4002c000-40158000 r-xp 00000000 03:05 1544529 /lib/libc.so.6
40158000-4015d000 rw-p 0012b000 03:05 1544529 /lib/libc.so.6
4015d000-4015f000 rw-p 00000000 00:00 0
[*] 60000000-60000000 rwxp 00000000 00:00 0
bffff000-c0000000 rwxp fffff000 00:00 0
```

The broken VMA in the above example has been marked with a [\*].

### Impact:

Since no special privileges are required to use the mremap(2) system call any process may misuse its unexpected behavior to disrupt the kernel memory management subsystem. Proper exploitation of this vulnerability may lead to local privilege escalation including execution of arbitrary code with kernel level access. Proof-of-concept exploit code has been created and successfully tested giving UID 0 shell on vulnerable systems.

The exploitability of the discovered vulnerability is possible, although not a trivial one. Paul has identified at least two different attack vectors for the 2.4 kernel series. All users are encouraged to patch all vulnerable systems as soon as appropriate vendor patches are released.

### ADDITIONAL INFORMATION

The original advisory is available at:

<http://isec.pl/vulnerabilities/isec-0012-mremap.txt>

<http://isec.pl/vulnerabilities/isec-0012-mremap.txt>.

The information has been provided by <mailto:ihaquer@isec.pl> Paul Starzetz.

=====

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:

list-unsubscribe@securiteam.com

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

Securiteam: [UNIX] Linux Kernel do\_mremap Local Privilege Escalation Vulnerability

=====  
=====

**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.