CoreLabs Overview

Advisories

Papers

Freesoft

IS Links

‹ Back

Advisories

03 | 20 | 2003

Vulnerability in Mutt Mail User Agent

Core Security Technologies Advisory
http://www.coresecurity.com

Date Published: 2003-03-20

Last Update: 2003-03-20

Advisory ID: CORE-20030304-02

Bugtraq ID: 7120

CVE Name: CAN-2003-0140

Title: Mutt Controlled IMAP server buffer overflow

Class: Boundary Error Condition (Buffer Overflow)

Remotely Exploitable: Yes

Locally Exploitable: No

Advisory URL: http://www.coresecurity.com/common/showdoc.php?idx=310&idxseccion=10

Vendors notified:
. Core Notification: 2003-03-11
. Notification aknowledged by Mutt: 2003-03-12
. Fix developed by Mutt: 2003-03-17
. Fix incorporated to releases of Mutt stable and unstable branches: 2003-03-19
. Public announcement of fixed packages: 2003-03-19

Release Mode: COORDINATED RELEASE

Vulnerability Description:

Mutt is a very popular small text-based MUA (Mail User Agent) for Unix operating systems.
For more information about Mutt visit http://www.mutt.org

The Mutt Mail User Agent (MUA) has support for accessing remote mailboxes through the IMAP protocol.

By controlling a malicious IMAP server and providing a specially crafted folder, an attacker can crash the mail reader and possibly force execution of arbitrary commands on the vulnerable system with the privileges of the user running Mutt.

Vulnerable Packages:

Versions of Mutt up to, and including, 1.4.0 (stable)
Versions of Mutt up to, and including, 1.5.3 (unstable)

Solution/Vendor Information/Workaround:

Mutt 1.4.1 (stable branch) and 1.5.4 (unstable) have been released with a fix for the vulnerability.

These versions will soon be available from ftp://ftp.mutt.org/mutt/.

Credits:

This vulnerability was found by Diego Kelyacoubian, Javier Kohen, Alberto Solino, and Juan Vera from **Core Security Technologies** during Bugweek 2003 (March 3-7, 2003).

We would like to thank Thomas Roessler, Edmund Grimley Evans and Marco d'Itri for their quick response to our report and the generation of fixed Mutt packages.

Technical Description - Exploit/Concept Code:

According to the RFC2060 (INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1), section 5.1.3: "By convention, international mailbox names are specified using a modified version of the UTF-7 encoding described in [UTF-7]."

When mutt has to convert from its internal representation in UTF-8 to UTF-7-like encoding it calls indirectly the function utf8_to_utf7() in module imap/utf7.c. The aforementioned function miscalculates the maximum output length; therefore provided that one can control the IMAP server, it is possible to craft a folder name that will generate output at least 50% larger than the calculated maximum.

These perl oneliners will generate two different folder names whose length is past the calculated maximum:

```
 perl -e 'print (chr(0x10) x 20)'
 perl -e 'print ((chr(0x10) . chr(0x41)) x 20)'
```

The second produces a longer output after conversion. It might be necessary to increase the multiplier to see Mutt crash.

A post-mortem analysis of the crashed process shows:

```
#0  0x4207434f in _int_realloc () from /lib/i686/libc.so.6
#1  0x42073416 in realloc () from /lib/i686/libc.so.6
#2  0x080aafbd in safe_realloc (p=0xbfffe194, siz=121) at lib.c:96
#3  0x080c58d2 in utf8_to_utf7 (u8=0x80f5708 "", u8len=0, u7=0xbfffe1d4,
    u7len=0x0) at utf7.c:237
#4  0x080c5961 in imap_utf7_encode (s=0xbfffe1d4) at utf7.c:252
#5  0x080c4cf7 in imap_munge_mbox_name (
    dest=0xbfffe720
"imap://abcd@192.168.10.10/\020A\020A\020A\020A\020A\020A\020A\020A
\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A",
    dlen=1024,
    src=0x80f0e90
"\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A
\020A\020A\020A\020A\020A\020A\020A")
at util.c:507
#6  0x080bfe65 in imap_open_mailbox (ctx=0x80f0d78) at imap.c:548
#7  0x08082cca in mx_open_mailbox (
    path=0xbfffedd0
"imap://abcd@192.168.10.10/\020A\020A\020A\020A\020A\020A\020A\020A
\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A\020A",
flags=0,
    pctx=0x0) at mx.c:694
#8  0x0805ff66 in mutt_index_menu () at curs_main.c:1032
#9  0x08079083 in main (argc=3, argv=0xbffffa04) at main.c:841
#10 0x420158d4 in __libc_start_main () from /lib/i686/libc.so.6

gdb) x/10i $pc
0x4207434f <_int_realloc+175>:  testb  $0x1,0x4(%eax,%esi,1)
0x42074354 <_int_realloc+180>:  jne    0x4207440b <_int_realloc+363>
0x4207435a <_int_realloc+186>:  mov    0xffffffe8(%ebp),%edi
0x4207435d <_int_realloc+189>:  add    %eax,%edi
0x4207435f <_int_realloc+191>:  cmp    0xfffffff0(%ebp),%edi
0x42074362 <_int_realloc+194>:  jb     0x4207440b <_int_realloc+363>
0x42074368 <_int_realloc+200>:  mov    0x8(%esi),%edx
0x4207436b <_int_realloc+203>:  mov    0xc(%esi),%eax
0x4207436e <_int_realloc+206>:  mov    %eax,0xc(%edx)
0x42074371 <_int_realloc+209>:  mov    %edx,0x8(%eax)
(gdb) p/x $eax
$22 = 0x41424120
(gdb) p/x $esi
$23 = 0x80f2b70
```

$22 is controlled by the attacker.

Although we believe this vulnerability to be exploitable, further research is required to provide proof of concept code and a reliable exploitation method.

About Core Security Technologies

Core Security Technologies develops strategic security solutions for Fortune 1000 corporations, government agencies and military organizations. The company offers information security software and services designed to assess risk and protect and manage information assets.
Headquartered in Boston, MA, Core Security Technologies can be reached at 617-399-6980 or on the Web at http://www.coresecurity.com.

To learn more about CORE IMPACT, the first comprehensive penetration testing framework, visit http://www.coresecurity.com/products/coreimpact

DISCLAIMER:

The contents of this advisory are copyright (c) 2003 CORE Security Technologies and may be distributed freely provided that no fee is charged for this distribution and proper credit is given.