New issue

Jump to bottom

libsolv “prune_to_recommended” function two heap overflow vulnerabilities #420

Closed **yangjiageng** opened this issue on Dec 13, 2020 · 1 comment

yangjiageng commented on Dec 13, 2020 • edited

Description:
There are two heap-buffer overflow vulnerabilities in
static void prune_to_recommended(Solver *solv, Queue *plist) at src/policy.c: line 403 & line 514
FOR_PROVIDES(p, pp, rec)
MAPSET(&solv->recommendsmap, p); // line 403
The first case, it involves variable “solv->recommendsmap”.
The libsolv defines MAPSET as following:
#define MAPSET(m, n) ((m)->map[(n) >> 3] |= 1 << ((n) & 7))
Therefore, MAPSET(&solv->recommendsmap, p) involves the variable “solv->recommendsmap->map[p >> 3]”.
The type of the variable “solv->recommendmap” is the structure Map.
The Map structure defines as following:
typedef struct s_Map {
unsigned char *map;
int size;
} Map;
If the value of the index variable “p>>3” is bigger than “solv->recommendmap->size”, there will be a heap-buffer overflow bug.

if (!MAPTST(&solv->recommendsmap, p)) // line 514
The libsolv defines MAPTST as following:
#define MAPTST(m, n) ((m)->map[(n) >> 3] & (1 << ((n) & 7))).
Therefore, the variable “MAPTST(&solv->recommendsmap, p)” is same with “solv->recommendmap->map[p >> 3] & (1 << (p & 7))”.
It also causes a heap overflow bug as the first bug.
Our PoC files can trigger these two heap overflow bugs.

Please reproduce this issue through the following PoC: /libsolvBuildDir/tools/testsolv PoC-policy_update_recommendsmap-403
If you configure CC with flag -fsanitize=address, you will get the following outputs:

```
testcase_read: system: unknown repo 'system'
str2job: bad line 'in'
testcase_read: system: unknown repo 'system'
  O6Đ龘SjK  x¡_rkD$,gc"r°©I='<§溯9  ±ⱼ§E>}'DHA  ehÉ‹ς
testcase_read: cannot parse command 'zM''ⱷr  ‘BĐtŠ¾°Y0bOyjˆ®ojRŽ0by#¯7a`N¯¯ˆ
c␣x鼻¥0¯c&¤@¯¯Ÿv^ч蜘G¯¸=zm¯?|¼¯#5³GÐ鴦(“t□r°66¢¯A)µ煊¼Q
testcase_read: could not open 'FuzzDir/out/<inlitest'
testcase_read: cannot parse command '| ⁅ µ@gþ °°¸E3DnoV(  弓
testcase_read: cannot parse command 'dy  1
*
栖b  F¯
Ч$Ů‌a
  1'
testcase_read: cannot parse command '1焰¼x‚Ş‌ćbÚhnK#>alternative'
=================================================================
==107444==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6020000000d1 at pc 0x7f2ba097fb3c bp 0x7ffc4dab2950 sp 0x7ffc4dab2948
READ of size 1 at 0x6020000000d1 thread T0
XshellXshellXshell #0 0x7f2ba097fb3b in policy_update_recommendsmap /root/Experiments/real-world/libsolv/src/policy.c:403:10
#1 0x7f2ba097fb3b in prune_to_recommended /root/Experiments/real-world/libsolv/src/policy.c:499:5
#2 0x7f2ba097fb3b in policy_filter_unwanted /root/Experiments/real-world/libsolv/src/policy.c:1365:7
#3 0x7f2ba081531d in resolve_dependencies /root/Experiments/real-world/libsolv/src/solver.c:2039:4
#4 0x7f2ba07faba4 in solver_run_sat /root/Experiments/real-world/libsolv/src/solver.c:2722:15
#5 0x7f2ba083065a in solver_solve /root/Experiments/real-world/libsolv/src/solver.c:4137:3
#6 0x4f1eea in main /root/Experiments/real-world/libsolv/tools/testsolv.c:241:8
#7 0x7f2b9f80cbf6 in __libc_start_main /build/glibc-S7xCS9/glibc-2.27/csu/../csu/libc-start.c:310
#8 0x41e6f9 in _start (/root/Experiments/real-world/libsolv/build/tools/testsolv+0x41e6f9)

0x6020000000d1 is located 0 bytes to the right of 1-byte region [0x6020000000d0,0x6020000000d1)
allocated by thread T0 here:
#0 0x4abe48 in calloc /root/Downloads/llvm-build/llvm/projects/compiler-rt/lib/asan/asan_malloc_linux.cpp:154
#1 0x7f2ba0969f10 in solv_calloc /root/Experiments/real-world/libsolv/src/util.c:79:9
#2 0x7f2ba07e3dba in map_init /root/Experiments/real-world/libsolv/src/bitmap.c:24:22
#3 0x7f2ba07f1abe in solver_create /root/Experiments/real-world/libsolv/src/solver.c:1322:3
#4 0x7f2ba9da92d4 in testcase_read /root/Experiments/real-world/libsolv/ext/testcase.c:2268:15
#5 0x4f144b in main /root/Experiments/real-world/libsolv/tools/testsolv.c:159:11
#6 0x7f2b9f80cbf6 in __libc_start_main /build/glibc-S7xCS9/glibc-2.27/csu/../csu/libc-start.c:310
```

SUMMARY: AddressSanitizer: heap-buffer-overflow /root/Experiments/real-world/libsolv/src/policy.c:403:10 in policy_update_recommendsmap
Shadow bytes around the buggy address:
0x0c047fff7fc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7fd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff8000: fa fa fd fd fa fa 00 fa fa fa 00 02 fa fa 00 00
=>0x0c047fff8010: fa fa 04 fa fa fa 04 fa fa fa[01]fa fa fa 01 fa
0x0c047fff8020: fa fa 00 02 fa fa 07 fa fa fa 00 00 fa fa 04 fa
0x0c047fff8030: fa fa 04 fa fa fa 00 02 fa fa 00 00 fa fa 04 fa
0x0c047fff8040: fa fa 04 fa fa fa fd fa fa fa fd fa fa fa 04 fa
0x0c047fff8050: fa fa 01 fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8060: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
Addressable: 00
Partially addressable: 01 02 03 04 05 06 07
Heap left redzone: fa
Freed heap region: fd
Stack left redzone: f1
Stack mid redzone: f2
Stack right redzone: f3
Stack after return: f5
Stack use after scope: f8
Global redzone: f9
Global init order: f6
Poisoned by user: f7
Container overflow: fc
Array cookie: ac
Intra object redzone: bb
ASan internal: fe
Left alloca redzone: ca
Right alloca redzone: cb
Shadow gap: cc
==107444==ABORTING

Please reproduce this issue through the following PoC: /libsolvBuildDir/tools/testsolv PoC-policy_update_recommendsmap-514
If you configure CC with flag -fsanitize=address, you will get the following outputs:

testcase_read: system: unknown repo 'system'
str2job: bad line 'insta½½system'
result: unknown flag 'trans#>erase'
testcase_read: could not open 'FuzzDir/out/D-1.0-
1rrr
rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrr225rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrro+rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrules'
testcase_read: cannot parse command 'sysine>'
testcase_read: could not open 'FuzzDir/out/<inl:'
testcase_read: cannot parse command '>=Con:'
testcase_read: cannot parse command 'sysine>'
testcase_read: could not open 'FuzzDir/out/<inl:'
str2job: bad line 'insta½½system'
result: unknown flag 'trans#>erase'
testcase_read: could not open 'FuzzDir/out/D-1.0-1rrrrrrrrrrrrrrrrrrrrrrrrrr@~rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrr햏
rr
rrrrrrrrrrrrrrrrrrrro+rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrules'
testcase_read: cannot parse command 'KX'
str2job: bad line 'provrovides E'
result: unknown flag 'transction'
===
==71812==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x602000000013 at pc 0x7f2e7ddb18c2 bp 0x7ffcf10835f0 sp 0x7ffcf10835e8
READ of size 1 at 0x602000000013 thread T0
#0 0x7f2e7ddb18c1 in prune_to_recommended /root/Experiments/real-world/libsolv/src/policy.c:514:12
#1 0x7f2e7ddb18c1 in policy_filter_unwanted /root/Experiments/real-world/libsolv/src/policy.c:1365:7
#2 0x7f2e7de3b5af in solver_choicerulecheck /root/Experiments/real-world/libsolv/src/rules.c:3180:7
#3 0x7f2e7de3b5af in solver_addchoicerules /root/Experiments/real-world/libsolv/src/rules.c:3430:20
#4 0x7f2e7dc60609 in solver_solve /root/Experiments/real-world/libsolv/src/solver.c:4065:5
#5 0x4f1eea in main /root/Experiments/real-world/libsolv/tools/testsolv.c:241:8
#6 0x7f2e7cc3dbf6 in __libc_start_main /build/glibc-S7xCS9/glibc-2.27/csu/../csu/libc-start.c:310
#7 0x41e6f9 in _start (/root/Experiments/real-world/libsolv/build/tools/testsolv+0x41e6f9)

0x602000000013 is located 2 bytes to the right of 1-byte region [0x602000000010,0x602000000011)
allocated by thread T0 here:
#0 0x4abe48 in calloc /root/Downloads/llvm-build/llvm/projects/compiler-rt/lib/asan/asan_malloc_linux.cpp:154
#1 0x7f2e7dd9af10 in solv_calloc /root/Experiments/real-world/libsolv/src/util.c:79:9
#2 0x7f2e7dc14dba in map_init /root/Experiments/real-world/libsolv/src/bitmap.c:24:22
#3 0x7f2e7dc22abe in solver_create /root/Experiments/real-world/libsolv/src/solver.c:1322:3
#4 0x7f2e871da2d4 in testcase_read /root/Experiments/real-world/libsolv/ext/testcase.c:2268:15
#5 0x4f144b in main /root/Experiments/real-world/libsolv/tools/testsolv.c:159:11
#6 0x7f2e7cc3dbf6 in __libc_start_main /build/glibc-S7xCS9/glibc-2.27/csu/../csu/libc-start.c:310

SUMMARY: AddressSanitizer: heap-buffer-overflow /root/Experiments/real-world/libsolv/src/policy.c:514:12 in prune_to_recommended
Shadow bytes around the buggy address:
0x0c047fff7fb0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7fc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7fd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c047fff8000: fa fa[01]fa fa fa 01 fa fa fa 00 fa fa fa fd fd
0x0c047fff8010: fa fa fd fa fa fa fd fd fa fa 07 fa fa fa 00 00
0x0c047fff8020: fa fa 04 fa fa fa 04 fa fa fa 07 fa fa fa 00 00
0x0c047fff8030: fa fa 04 fa fa fa 04 fa fa fa 07 fa fa fa 00 00
0x0c047fff8040: fa fa 04 fa fa fa 04 fa fa fa 07 fa fa fa 00 00
0x0c047fff8050: fa fa 04 fa fa fa 04 fa fa fa 07 fa fa fa 00 00
Shadow byte legend (one shadow byte represents 8 application bytes):
Addressable: 00
Partially addressable: 01 02 03 04 05 06 07
Heap left redzone: fa
Freed heap region: fd
Stack left redzone: f1
Stack mid redzone: f2
Stack right redzone: f3
Stack after return: f5
Stack use after scope: f8
Global redzone: f9
Global init order: f6
Poisoned by user: f7
Container overflow: fc
Array cookie: ac
Intra object redzone: bb
ASan internal: fe
Left alloca redzone: ca
Right alloca redzone: cb
Shadow gap: cc
==71812==ABORTING

The ASAN outputs information about these overflow bug.
And attacker can use this bug to achieve a DoS attack.
Please reproduce and fix these two bugs.

mlschroe commented on Dec 14, 2020 Member

Made testcase reader more robust.

mlschroe closed this as completed on Dec 14, 2020

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants